电 子 科 技 大 学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

硕士学位论文

MASTER THESIS

论文题目 基于半监督学习的遥感目标检测算法设计与实现

学科专业 计算机科学与技术

学 号 201821080515

作者姓名 邹芷桐

指导教师 段翰聪 教授

分类号 _____ 密级 _____

UDC [注1] _____

学 位 论 文

基于半监督学习的遥感目标检测算法设计与实现

（题名和副题名）

邹芷桐

（作者姓名）

指导教师 段翰聪 教 授

电子科技大学 成 都

（姓名、职称、单位名称）

申请学位级别 硕士 学科专业 计算机科学与技术

提交论文日期 2021.03.17 论文答辩日期 2021.05.12

学位授予单位和日期 电子科技大学 2021 年 6 月

答辩委员会主席 _____

评阅人 _____

注 1：注明《国际十进分类法 UDC》的类号。

DESIGN AND IMPLEMENTATION OF REMOTE SENSING TARGET DETECTION ALGORITHM BASED ON SEMI-SUPERVISED LEARNING

A Master Thesis Submitted to

University of Electronic Science and Technology of China

Discipline: **Computer Science And Technology**

Author: **Zou Zhitong**

Supervisor: **Prof. DuanHancong**

School: **School of Computer Science & Engineering**

摘 要

随着遥感卫星技术的不断发展，海量遥感图像数据每天都在产生，被广泛应用于海洋研究、气候监测、资源勘探等领域。针对遥感图像的目标检测任务，是遥感图像处理与分析领域备受关注的课题，也是后续复杂的研究任务的重要基础。在目标检测任务中，数据标签需要准确标出目标所在位置，完成所有数据标注往往会耗费极大的人力和财力。因此，引入半监督学习，有效利用无标签数据帮助模型训练，缓解人力依赖，同时避免数据浪费，对于遥感目标检测研究具有很高的实用意义。

本文工作主要分为两大部分：

第一部分，针对遥感目标特性的网络优化设计。由于遥感图像拍摄高度的特殊性，遥感目标存在着多尺度、多方向、小而密集等特点。本文针对其特点对 Faster R-CNN 网络进行优化设计：添加多尺度预测结构；使用 K-Means 生成符合数据集分布的 Anchor 设置；提出空间注意力模块 SAM 和通道注意力模块 CAM，并按不同的顺序应用在不同层级的特征图上；修改回归框损失为更符合评价指标的 GIOU 损失等。经过验证，改进后的 RF R-CNN 网络与原网络相比 mAP 提高了 3.84%。

第二部分，基于半监督的遥感目标检测算法研究。第一，对半监督的自训练方法进行了递进式的研究。首先在目标检测任务上进行了基于伪标签的简单自训练实验和改进，提出慢启动自训练方法。其次，通过对慢启动方法局限性的分析，引入了与主动学习结合的主动半监督自训练方法。在此过程中，本文提出了一种基于委员会的不确定度采样策略，为主动半监督自训练采样出了高不确定度和低不确定度的样本。实验结果证明该方法对模型性能提升帮助很大，且相比于用随机采样策略的 mAP 高出 3.37%。 第二，对无需生成伪标签的一致性正则化方法进行了研究，提出了一种基于 Mean Teacher 的学生-教师半监督训练框架。在此框架中，本文设计了目标检测任务下针对无标签样本的一致性损失，对有标签样本和无标签样本进行了同时训练。实验结果证明该方法在使用相同数量的无标签样本时，mAP 比改进后的慢启动自训练方法高出 2.11%。

关键词：遥感图像，目标检测，半监督学习，自训练，一致性正则化

ABSTRACT

With the continuous development of remote sensing satellite technology, massive amounts of remote sensing image data are generated every day for environmental monitoring, ocean research, climate monitoring, resource exploration, etc. The task of target detection for remote sensing image is a hot topic in the field of remote sensing image processing and analysis, and it is also an important basis for the follow-up complex research tasks. In the task of target detection, data labels need to accurately mark the location of the target, and it often costs a lot of manpower and financial resources to complete all data labeling. Therefore, the introduction of semi supervised learning to effectively use unlabeled data to help model training, alleviate human dependence, and avoid data waste, has high practical significance for remote sensing target detection research.

This thesis is divided into two parts:

The first part is the network optimization design for the characteristics of remote sensing target. Due to the particularity of the shooting height of remote sensing image, remote sensing targets have the characteristics of multi-scale, multi direction, small and dense. According to its characteristics, this thesis optimizes the design of Faster R-CNN network: adding multi-scale prediction structure; using K-Means to generate anchor settings that conform to the dataset's distribution; proposing spatial attention module SAM and channel attention module CAM, which are applied to feature maps of different levels in different order; modifying the regression box loss to GIOU loss that is more consistent with the evaluation index. After verification, the improved RF R-CNN network has a 3.84% increase in mAP compared with the original network.

The second part is the research of remote sensing target detection algorithm based on semi-supervised. First, this thesis conducts a progressive research on the semi-supervised self-training method. Firstly, a simple self-training experiment based on pseudo tag is carried out on the target detection task, and a slow start self-training method is proposed. Secondly, through the analysis of the limitations of slow start method, an active semi-supervised self-training method combined with active learning is introduced. In this process, this thesis proposes a committee-based uncertainty sampling strategy to sample samples with high uncertainty and low uncertainty for

active semi-supervised self-training. The experimental results prove that it is very helpful to improve the performance of the model, and the mAP is 3.37% higher than that of the random sampling strategy. Second, research is conducted on a consistent regularization method that does not need to generate pseudo-labels. This thesis proposes a student-teacher semi-supervised training framework based on Mean Teacher. Under this framework, this thesis designs a consistency loss for target detection tasks, and simultaneously trains both labeled samples and unlabeled samples. Experimental results prove that when the method uses the same number of unlabeled samples, mAP is 2.11% higher than the improved slow-start self-training method.

Keywords: remote sensing image, target detection, semi supervised learning, self-training, consistency regularization

目 录

第一章 绪论

1.1 研究背景及意义

目标检测作为计算机视觉领域的基本任务，一直都是备受关注的热门。近年来深度学习技术迅速发展，将目标检测研究推向了新的高潮，从传统的利用手工特征的VJ检测器[1]到各种层出不穷的神经网络模型（如RCNN[2]、Fast RCNN[3]、YOLO[4]、RetinaNet[5]等），将目标检测精度一再刷新。随着目标检测技术的不断发展，其在社会和科学研究中应用的领域也愈加广泛，除了常见的人脸检测、行人检测、生活物品检测外，在智能家居、智慧交通、医疗影像、遥感图像等领域也展现出可观的应用前景，可以说目标检测技术已经深入各行各业。

经过多年的遥感卫星产业发展建设，我国已经逐渐形成了全天候，全球覆盖的对地观测能力。按照使用用途的角度，遥感卫星可以分为军用、民用和商用三种类型，军用遥感卫星主要服务于重点目标侦察等军事目的，民用遥感卫星包括政府、院校、科研机构及民间爱好者发射的用于环境监测、海洋研究、气候监测、资源勘探等用途的遥感卫星，商业遥感卫星则是以实现盈利为目的。总而言之，遥感图像技术已经在国防军事领域、自然科学、民生研究等各个领域中体现了重要的应用价值。而针对遥感图像的目标检测任务，是遥感图像处理与分析领域备受关注的课题，是后续复杂的研究任务的重要基础。如何充分利用图像中的色彩、纹理、边缘、形状等信息对目标进行检测、分类和精确定位，是遥感图像目标检测任务的关键所在，而遥感数据的特点与自然场景图像存在较大差异，在传统的目标检测数据集上应用良好的神经网络模型，直接迁移到遥感数据上并不能获得最好的结果，需要我们针对遥感目标的特性设计专门的网络，而近两年越来越多的公开遥感数据集的出现，为遥感目标检测方向的发展打下了一个良好的基础。

遥感卫星发射数量逐年递增，而随着遥感技术的不断发展，遥感图像愈加凸显的一大优势就是容易获取，数据量大，仅是一颗绕地卫星在一天内不间断全天候地对地面进行扫描监测，就会产生海量的遥感图像数据。对于广阔的遥感图像，仅靠人工对所有图像中感兴趣目标进行标注，十分费时费力，并且容易出现误差，对于数据的利用效率低下，所以本文提出引入半监督学习技术提高对大量的无标签样本的利用效率。半监督学习介于监督学习和无监督学习之间，对于训练数据的要求只要一部分样本存在标签即可，剩下的大部分样本都可以没有标签，半监督学习能够通过利用无标签样本中的信息来帮助模型精度提升。半监督学习缓解了对人工标注的依赖，降低标注成本，因此针对遥感图像目标检测任务引入半监督学习技术

具有现实的研究意义。

1.2 国内外研究现状

目标检测一直以来都是计算机视觉领域内的热门研究方向，在 2012 年之前，深度学习还未兴起，研究人员通常是使用人们根据经验设计的特征和传统的机器学习算法，如 2001 年提出的 VJ 人脸检测器，设计 Haar 特征来描述局部窗口内的特征，主要反映了局部窗口内的灰度变化情况，比如皮肤比头发的颜色更浅，脸颊比鼻翼的颜色更浅等。计算得出 Haar 特征后将其输入一个高效的分类算法 AdaBoost，最后将分类器认为是人脸的局部窗口输出，VJ 检测器在当时将人脸检测的速度推向历史新的高度。除了 VJ 检测器外，还有使用 HOG 特征与 SVM 算法进行行人检测[6]也是经典的检测策略。但以 2012 年作为分界线，深度学习神经网络的兴起对图像应用领域产生了极大的影响，2012 年 Alex Krizhevsky 提出的 AlexNet[7]，赢得了当年的 ILSVRC 竞赛，将机器识别的错误率从降低了接近 10%（26%→ 16%），并且远超第二名，引起了极大的轰动，也让研究人员们看到了神经网络的无限潜力，自此之后，各种神经网络层出不穷，如 VGG[8]、GoogleNet[9]、ResNet[10]等，横扫各大比赛冠军，深度学习一派欣欣向荣的景象，而目标检测领域也出现了很多优秀的网络模型，按照训练阶段主要分为两阶段模型和单阶段模型，两阶段模型以 RCNN 系列为代表，单阶段模型以 YOLO 系列为代表，直到 2020 年还在不断更新迭代。近年来，国内的科研机构和科技公司也在各大国际顶会产出了非常多优秀的论文和网络模型，如旷视提出的 ShuffleNet[11]、DetNet[12]，华为提出的 GhostNet[13]，商汤提出的 SiamRPN[14]等。随着深度学习在国内及国际繁荣发展，将遥感行业与深度学习相结合是非常自然的尝试，目前已有一些论文产出，如 Adam Van Etten 提出的 YOLT[15]，是基于 YOLOv2 模型改进，能非常快速地对大尺寸卫星图像进行扫描处理，还有王彦情、马雷等人提出的综述[16]。

20 世纪 90 年代就有研究者开始尝试在训练分类器时利用无标记样本来提高分类器性能，但 2000 年后半监督学习才逐步形成相对独立的理论和算法体系[17]。近年来，随着大数据时代的到来，对样本标记的依赖愈加严重，研究如何使用未标记的样本进行训练成了自然的需求，半监督学习技术才开始逐渐崭露头角，被研究者们关注。2013 年，Dong-Hyun Lee 提出了 Pseudo Label[18]，继承 self-training 的思想，先使用已有的标签样本训练出一个具有一定精度的初始分类器，再用该分类器对无标签样本进行预测，生成伪标签，再将有标签样本与带有伪标签的无标签样本一同进行训练，将拥有伪标签的无标签样本同样视为有标签样本，计算交叉熵来评估误差大小。2016 年起，关于使用一致性正则（Consistency Regularization）的

论文相继出现，如英伟达的 Laine 等人提出的 π Model 和 Temporal ensembling Model[19]，日本研究者 Takeru Miyato 等人提出的 VAT 算法[20]，Antti Tarvainen 等人提出的 Mean Teacher 算法[21]，谷歌团队提出的 MixMatch 等[22]，都遵循着一致性正则的原理，即对于一个输入，即使受到了一定的干扰，系统对其的预测都应该是一致的。另外还有一些基于机器学习的半监督技术如半监督支持向量机（Semi-supervised support Vector Machine），协同训练（co-training），图论半监督学习等，但由于很难与目标检测算法相结合，所以不再赘述。上述理论方法都是基于分类任务提出的，据本文作者调查的结果，在半监督学习目标检测领域，研究者们还没有提出专门的理论方法，基本都是将已有的半监督学习分类技术迁移到目标检测任务，并且经过本文作者的考察，将半监督学习方法应用在遥感目标检测相结合的研究屈指可数，杜泽星等人提出了一种基于 GAN 的半监督学习方法[23]，使用无标签样本预训练 GAN 网络，再使用 GAN 网络级联为目标检测网络，也并非传统的半监督学习方法。

1.3 面临的问题

综上所述，虽然基于半监督学习的遥感目标检测任务具有极高的现实意义，但是目前研究人员对于遥感目标检测的半监督学习的理论研究还处在一个起步阶段，绝大部分的半监督方法都是针对图像分类任务提出的，而分类任务与检测任务之间存在着不小的差距。分类是对图像的类别进行判断，输出值为某个类别，检测是更细粒度地对图像中的物体进行定位、分类，输出值为一个甚至多个包围框和框中物体的类别。所以针对图像分类提出的半监督方法不一定适用于目标检测任务，本文将尽绵薄之力，尝试将各种半监督学习方法与遥感目标检测任务相结合，利用半监督学习使用无标签数据来提高遥感目标检测模型的性能。

另一方面，遥感图像与生活常见图像存在着一定的不同，常用的目标检测数据集，如 VOC 数据集[24]，多为平视角，且拍摄距离相对较近，而由于遥感卫星对地观测的角度为鸟瞰角度，且距离通常大于 30 千米，所以遥感图像与传统目标检测图像相比存在较大的差异，而在 VOC 数据集上效果很好的模型不一定在遥感数据集上效果很好，所以本文将针对遥感目标的特性进行网络的定制优化。

1.4 本文的主要研究内容和整体架构

本文一共分为五个章节，基本架构安排如下：

第一章，介绍基于半监督学习的遥感目标检测的研究背景和意义，确定了将半监督技术引入遥感目标检测领域的必要性，同时介绍了国内外研究现状，回顾了目

标检测领域、半监督学习领域的发展和现状，提出了研究所面临的问题，即半监督学习目标检测研究尚处于起步阶段，遥感图像与传统图像之间也存在一定的差异。

第二章，介绍本文技术涉及领域的一些基础理论以及相关方法，包括目标检测和半监督学习领域。

第三章，对遥感数据集的特点进行分析，针对遥感目标的特性进行网络优化设计，提出不同的改进模块并对其优化效果进行分析，最后进行实验对优化效果进行验证。

第四章，对基于半监督学习方法的遥感目标检测算法进行研究。主要分为简单自训练方法、主动半监督自训练方法和一致性正则化方法三个方法，对原有的基于分类任务的方法改造为针对目标检测任务的方法，并进行实验和分析。

第五章，总结全文工作内容，并对本文工作未竟之事进行展望。

第二章 基础理论研究及相关方法介绍

本章对本文涉及的模型或方法进行原理性介绍，从本质出发分析其原理，主要分为两大模块，其中一个模块是对目标检测算法的介绍，以训练的阶段的不同划分为两阶段和单阶段模型，另一个模块就是对半监督学习算法的介绍。

2.1 目标检测算法概述

目标检测任务就是对于一张输入图像，返回其中感兴趣目标的包围框和目标类别。传统的机器视觉将目标检测的整体流程概括如图 2-1 所示：

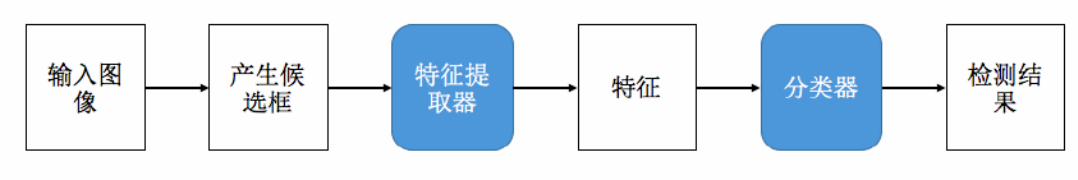


图 2-1 目标检测流程图

在产生候选框步骤中，早期的目标检测模型大多数是使用滑动窗口法选取候选框，即选用不同大小的窗口，从图像的最左上位置开始，从左到右，从上到下，依次进行遍历，每次遍历就会产生一个候选窗口，遍历结束后产生的所有窗口集合就是目标的候选集合，很显然这个方法将会产生大量无效的候选框（即框中根本没有目标或只有部分目标），后续还要对所有候选框进行特征提取，付出的计算资源与收益差距太大。为了解决这个问题，J.R.R. Uijlings 等人提出了选择性搜索（Selective Search，SS）算法，该算法的核心思想就是利用图像中的物体具有局部相似性（颜色，纹理），根据某种算法计算子区域间的相似性，并不断进行区域合并，并对合并后的区域计算外接矩形，作为候选框。Selective Search 算法相比于滑窗法极大地提高了候选框的生成效率和质量，节省了许多计算资源，在 R-CNN 和 Fast R-CNN 中都在被使用。后来随着深度学习的发展，目标检测的整个流程都交给了神经网络来完成，如 RPN 网络代替 SS 算法，此部分在 2.2.2 节中将详细阐述。

产生候选框集合后下一步就是将其送入特征提取器中，对图像进行特征提取。在传统的机器学习方法中提取的都是研究人员们根据研究经验设计的特征，如 HAAR 特征，HOG 特征，SIFT 特征等，这些特征的一个特点就是具有可解释性，如 HAAR 特征反映了局部的灰度变化，HOG 和 SIFT 特征反映了梯度变化，但手

工设计的特征存在较大的局限性，对图像的描述层次较为低级，而深度卷积神经网络虽然解释性不强，但是能够提取出鲁棒性强的高层次特征，这是因为神经网络的本质就是用非常复杂的模型（上百万的参数）来对数据的模式进行拟合记忆，而手工设计提取的特征只是简单匹配一部分数据模式而已，很多隐秘的关联不能被我们所发现和描述，所以神经网络的特征提取能力更强。

提取出候选框图像的特征后下一步就是将其输入分类器中进行类别判断。传统的机器学习方法使用了 SVM、AdaBoost 等分类器，而现在经过技术的不断发展演化，不仅包括最后的分类步骤，上述所有步骤都由神经网络一力承担，具体哪层网络承担的角色也变得模糊，中间过程被神经网络“封装”起来，成为一个“黑盒”。根据是否生成感兴趣区域（proposal region），目标检测算法根据检测步骤的不同，大致分为两类：两阶段（two-stage）模型和单阶段（one-stage）模型，下面将对卷积神经网络的基本组成和目标检测两大类模型进行介绍。

2.1.1 卷积神经网络概述

关于图像的任务，使用其他领域非常常见的全连接网络并不是很好的选择，因为图像是由像素点组成的，每个像素点有 RGB 三个通道，因此一张 256x256 大小的图像，如果使用全连接方式，即本层与相邻层的所有神经元之间都互相连接，那么即便假设第一层隐藏层仅 15 个神经元且没有偏置项（bias），输入层至第一层间也需要 256*256*3*15=2949120 个权重（weight），参数量过于巨大，在反向传播时需要大量的计算，从计算资源角度上并不建议使用全连接网络。受生物领域的感受野机制的启发，研究者提出了卷积神经网络（Convolutional Neural Networks, CNN），引入“图像卷积”概念，相对于传统的全连接神经网络，大大减少了所需参数量。卷积神经网络有三个十分重要的特性：局部连接、权重共享及空间上下采样，他们是由卷积操作的特殊性引入的，权重共享使得 CNN 具有了平移不变性，空间上下采样使得 CNN 拥有了缩放不变性，使得其能够更好地抽取出图像高层语义特征。卷积神经网络的基本组件包含卷积层、池化层、全连接层，下面将依次对它们进行介绍：

（1）卷积层

CNN 中最关键，最具有特色的部件就是卷积层，正是因为卷积层的存在给卷积神经网络带来了局部连接、权重共享的特性。卷积层由一组卷积核组成，卷积层的输入有两种情况，分别为原始图像或上一层网络输出的特征图，一次卷积操作示意如图 2-2 所示（仅展示通道数为 1 的情况）：

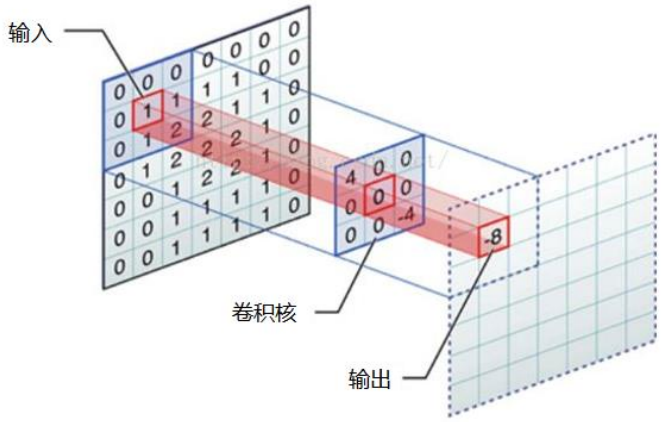


图 2-2 卷积操作示意图

卷积核对输入的图像进行依次扫描，每次卷积操作将对应的位置乘积之和作为本层输出特征图的特征值，最终获得一张输出特征图（Feature Map）。卷积核的维度为[N，W，H，K]，其中 N 为输入特征图通道数，W、H 分别为卷积核的宽和高、K 为卷积核的数量，该值决定了本层输出的特征图的通道数量。另外卷积核还有两个超参数，即步长 (Stride)、零填充 (Padding)，步长是指卷积核滑动的间隔，零填充是在输入图像的边缘添加值为零的像素点的个数。每次卷积操作仅是输入图像上某 W*H 区域内的像素点与卷积核进行数值运算，且卷积核的参数仅在后向传播时更新，前向推理的过程中并不发生改变，因此卷积层具有局部连接、参数共享的特性，并且卷积核将在输入图像上进行滑动扫描，不同的卷积核负责提取不同特征，因此卷积层具有一定的平移不变性。

（2）池化层

池化层又称为下采样层，如果在卷积网络前向推理过程中图像大小一直不发生变化，最后一层卷积输出特征图与原图像大小一样，那么整个网络在运行过程中将会由于中间存在高维度的卷积核，产生极高的计算量，并且卷积核对应的感受野将会受限。所以为了降低运行时的计算量和扩大感受野，我们一般都会在增加卷积核维度的同时降低特征图的尺寸。根据采样方式的不同，池化层又分为最大池化、平均池化、随机池化等。一般来说使用平均和最大池化较多，如图 2-3 为平均和最大池化示意图：

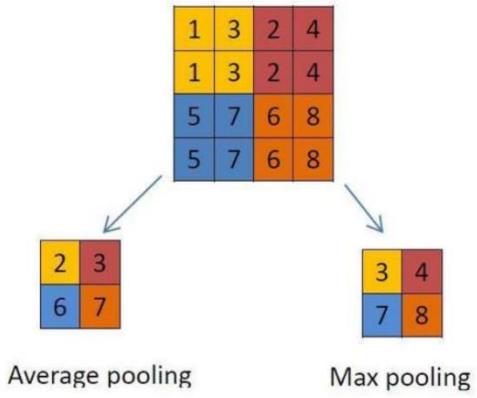


图 2-3 平均池化和最大池化示意图

池化层能使特征图变小，使得同样大小的卷积核在原图上对应的感受野变大，有利于提取全局抽象特征，并且带来一定的缩放不变性。

（3）全连接层

全连接层在本节开始已经进行了介绍，由于全连接层的参数量巨大，所以在图像相关的网络中一般只用全连接层进行高层次的全局语义信息的交互，所以一般只在最后使用一层或几层的全连接层进行交互和输出。

（4）激活函数

激活函数是卷积神经网络中不可或缺的一环。由于卷积层和全连接层都可以抽象为数学公式（2-1）：

$$Y_j^l = \sum_{i \in C_j} w_{ij}^l * X_i^{l-1} + b_j^l \qquad (2-1)$$

这里C_j表示输入本层的特征图的通道集合，X_i^{l-1}表示上一层中第 i 个通道的特征图，w_{ij}^l表示对应的卷积核权重，b_j^l表示对应的偏置项，Y_j^l表示本层经过卷积或全连接操作后的结果。可以看出卷积层和全连接层对于输入的改变都是线性的，而我们需要拟合的数据空间肯定不是线性分布的，因此需要引入非线性因素，也就是激活函数。常见的激活函数有 Sigmoid、ReLU 等。Sigmoid 函数的表达式见公式（2-2）：

$$\mathrm{f(x)} = \frac{1}{1+e^{-x}} \qquad (2-2)$$

所对应的函数图像如图 2-4 所示：

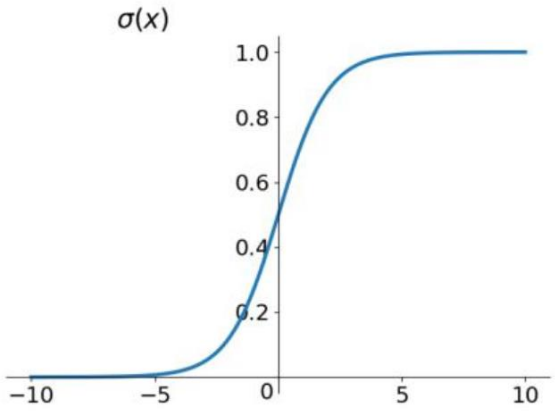


图 2-4 Sigmoid 函数图

从图像中可以直观看出，Sigmoid 函数对实数范围内的数字都可以映射到(0,1)之间，优点是输出有限，优化稳定，并且(0,1)之间可以作为输出层表示分类的概率，可解释性强。缺点就是在输入极大或极小时会出现饱和现象，函数导数几乎为零，在反向传播时会出现梯度消失的现象（即梯度基本为零），导致无法完成参数更新。并且由于 sigmoid 是指数形式，使用时计算复杂度较高。后来研究者们提出的 ReLU（Rectified Linear Unit），又称线性整流函数，在一定程度上解决了 Sigmoid 中梯度消失的问题。ReLU 的表达式如公式（2-3）所示：

$$f(x) = max(0, x) \tag{2-3}$$

其函数图像如图 2-5 所示：

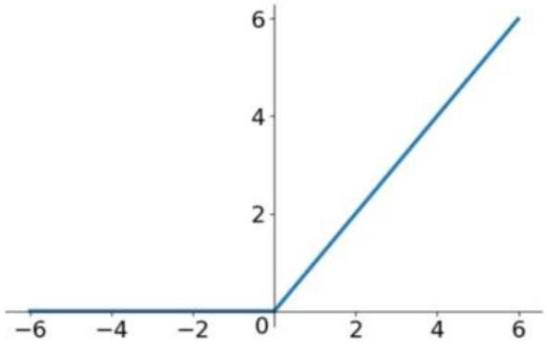


图 2-5 ReLU 函数图

在 x 大于 0 时，ReLU 的导数恒为 1，不会导致梯度消失或梯度爆炸的情况出现，并且 ReLU 的不需要指数运算，只要一个求最大值的操作就可以得到激活值，计算简单高效。但 ReLU 也存在一定缺点，倘若某次梯度更新幅度过大，某些节点权重调整后输出皆为负，导致 ReLU 激活值为 0，反向传播时梯度也为 0，该节点

权重在后续的更新中无法被更新，输出一直为负，就会陷入僵局，这样的情况被称为 Dead ReLU，研究者们也提出了 leaky ReLU 这样的激活函数来解决这一问题。

以 1998 年出现的 LeNet-5[25]为例，该网络就是一个简单而典型的卷积神经网络，后续演化出的各种卷积神经网络都离不开图 2-6 这样的基本结构。

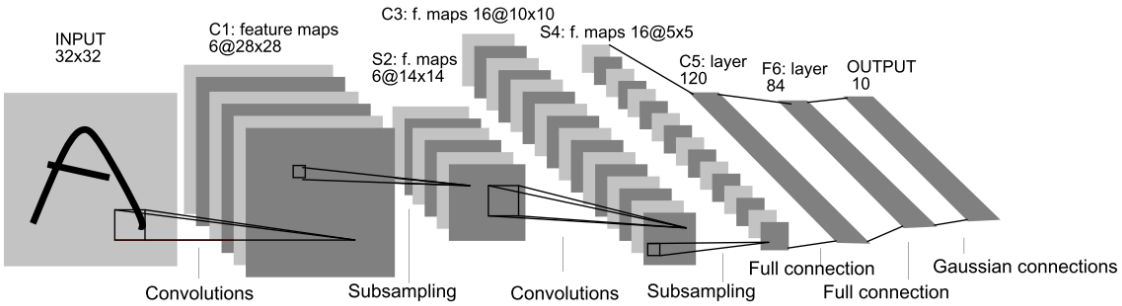


图 2-6 LeNet-5 网络结构

2.1.2 两阶段模型

目标检测模型的一大分支就是两阶段（two-stage）模型，其特点就是模型存在显式地生成区域建议（region proposal，也可称为候选区域）的步骤。R-CNN 是两阶段目标检测器的起源，R-CNN 算法流程与图 2-1 的流程一致，首先通过选择性搜索方法生成候选框，然后，将候选框从图像中裁取出来输入 CNN 中提取特征，最后，使用一组 SVM 分类器来判断候选框内的物体的类别，每个类别对应一个分类器，与此同时将特征输入线性回归器中，对候选框的坐标进行细微调整。如图 2-7 为 R-CNN 的流程示意图：

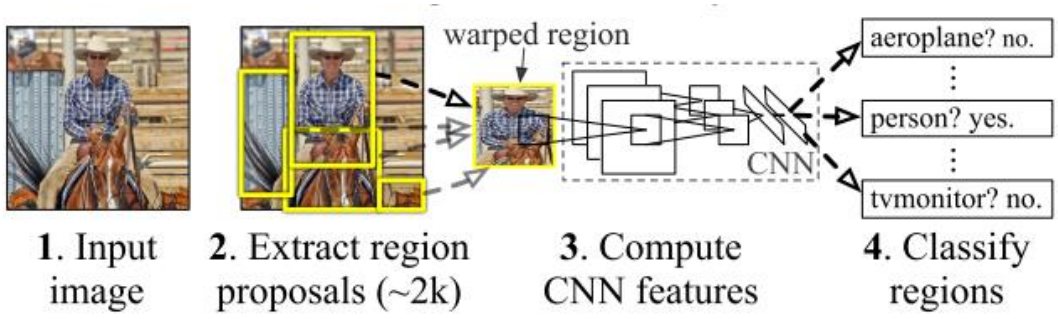


图 2-7 R-CNN 流程示意图

在最后获取了检测框的坐标与打分之后，R-CNN 使用了一个巧妙的算法来对那些重叠度高的预测框进行筛选，也就是非极大抑制（Non-Maximum Suppression，NMS）算法。该算法的策略是首先对所有检测框根据置信度进行排序，然后将置信度排序最高的框与其他的检测框计算交并比（即两个框交集与并集的比值，这是一

个对是否准确定位框很重要的衡量指标，在 3.4.2 节中有详细解释），如果其值大于研究人员提前设定的某阈值，说明两个框的重叠度很高，此时我们选择只留下置信度最高的那个框，将置信度较小的那个框剔除，因为同一个目标只留下一个最好的检测结果就可以了。接下来，对剩下的框重复这样的比较-剔除操作，如果剩余的框的交并比都小于该阈值了，说明关于这个目标的预测已经被剔除完毕，于是又选择下一个置信度最高的框，重复上述的操作，直到处理完所有的框。这样做保留了置信度最高的框，去除掉了那些虽然同样命中目标，但框的坐标可能存在一定偏移所以置信度较低的框。如图 2-8 为检测及 NSM 算法的效果图，可以看出经过 NMS 抑制后那些质量较低的检测框被筛除，检出效果大大优化：

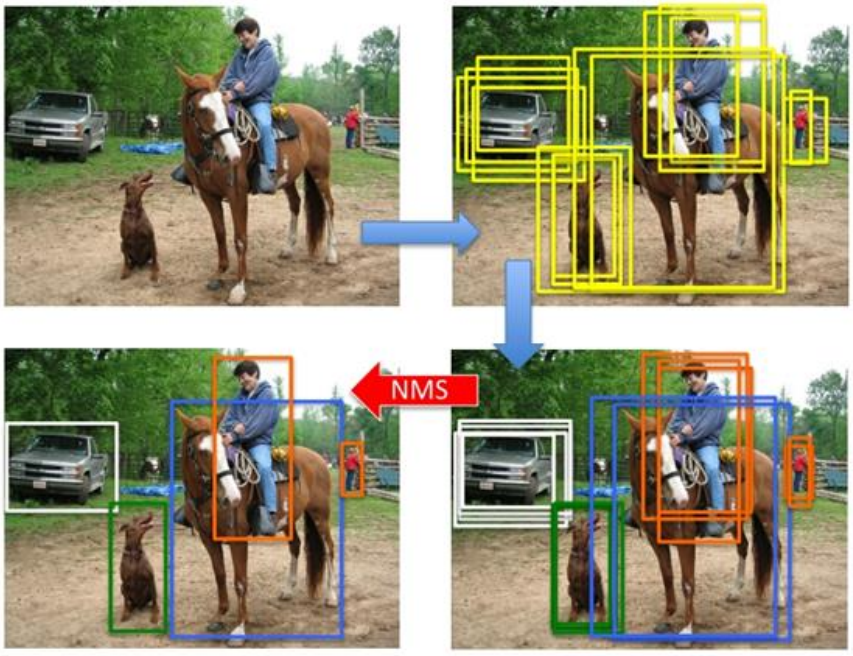


图 2-8 NMS 算法效果图

相较于传统的机器学习目标检测方法，R-CNN 有着更好的性能，但是也存在不足与局限：在卷积神经网络提取候选框特征的步骤中，由于全连接层的存在，要求 CNN 输入图像具有统一的尺寸，因此需要将输入图像缩放或裁剪到固定的尺寸，这样的操作会造成信息的丢失或者图像的扭曲，将会影响到检测器的精度。并且所有框都要进行特征提取操作，但实际上这些候选框中重叠的部分非常多，这样导致 CNN 进行了非常多的重复卷积计算。

针对 R-CNN 的问题，研究人员又提出了 Fast R-CNN，如图 2-9 为 Fast R-CNN 的架构图：

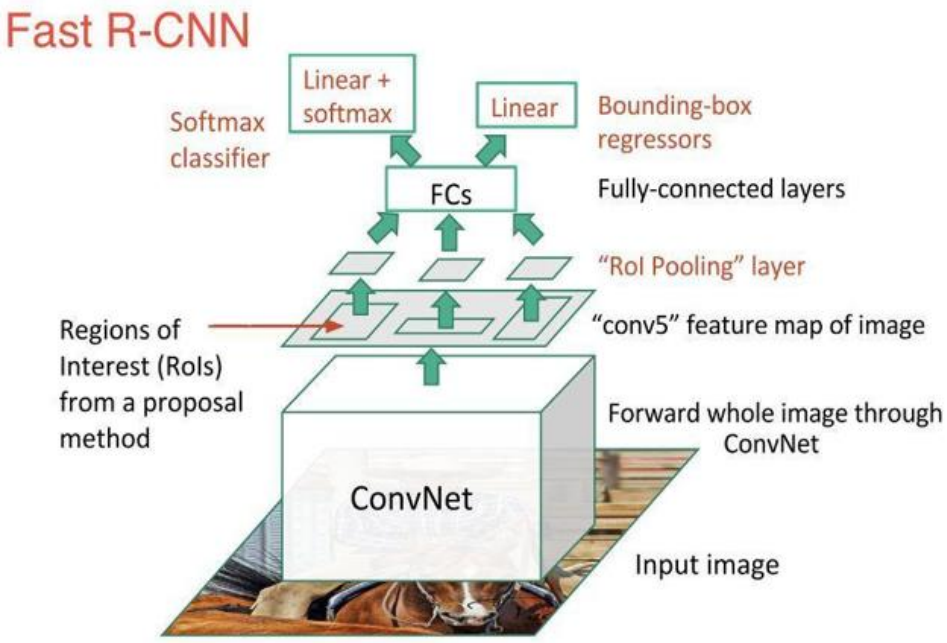


图 2-9 Fast R-CNN 架构图

Fast R-CNN 在 R-CNN 的基础上又做出了许多改进。首先，Fast R-CNN 不再将候选框分别地通过 CNN 提取特征（因为这样会导致很多重复的提取计算），而是直接将图像送入 CNN 进行特征提取，获得对应的的特征图（Feature Map），然后找到根据候选框在原始图像上的位置以及网络的 Stride，我们可以计算出该框在特征图上对应的位置，将对应位置的特征图裁剪出来，作为每个候选框的卷积特征输入到之后的层。其次，针对特征提取网络输入必须固定为统一尺寸的局限，Fast R-CNN 在最后的卷积层和全连接层之间加入了用于解除对特征提取网络的输入尺寸限制的空间金字塔池化（Spatial Pyramid Pooling，SPP）层， 又称为 ROI Pooling（Region of Interest Pooling）层，该层的核心思想是将通过选择性搜索获得的候选框分为 RxC 的网格，将每一个网格压缩输出一个值（一般都是使用最大池化），将输出值按固定方式组合起来，就将任意大小的候选框压缩为了 RxC 大小的特征向量，然后输入后续层进行特征计算。使用 SPP 层将解除对特征提取网络的输入尺寸限制，使之不论多大的输入，输出到全连接层的特征向量尺寸都是固定的。SPP 层的思想如图 2-10 所示：

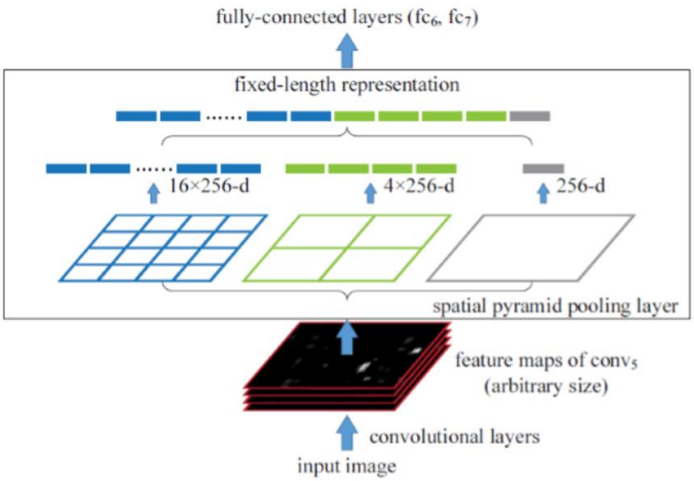


图 2-10 SPP 层流程图

除此之外，Fast R-CNN 使用卷积神经网络替代了连接在特征提取网络的全连接层之后的 SVM 分类器和线性回归器，分别用于预测候选框的类别和坐标值，还设计了多任务损失函数（Multi-task Loss），损失函数分支示意如图 2-11 所示：

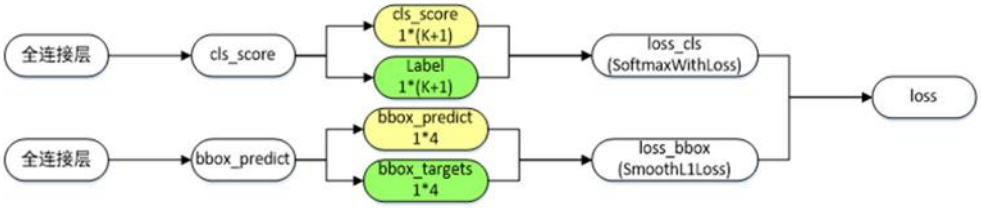


图 2-11 损失函数分支示意图

其函数表达式如公式（2-4）所示：

$$\mathrm{L}(\mathrm{p},\mathrm{u},t^{u},\mathrm{v}) = L_{cls}(p,u) + \lambda[u \geq 1]L_{loc}(t^{u},v) \qquad (2-4)$$

其中 p 为类别的网络预测值，u 为类别真实值，$[u \geq 1]$为指示函数，当 u 大于等于 1 时值为 1，小于 1 时值为 0，t^{u}为检测框的坐标预测值，v 为真实值。由公式可以看出，总的损失 L 是由两部分损失（L_{cls}，L_{loc}）得到的，超参数 λ 控制了这两个损失之间的平衡。其中类别损失L_{cls}使用的损失函数为常用的 Softmax，检测框损失L_{loc}使用的损失函数为新设计的 SmoothL_1函数，该函数是针对L_1、L_2损失函数提出的优化方案，三种损失函数的数学定义如公式（2-5）（2-6）（2-7）所示：

$$L_1(x) = |x| \qquad (2-5)$$

$$L_2(x) = x^2 \qquad (2-6)$$

$$smoothL1(x) = \begin{cases} 0.5x^2 & if|x| < 1 \\ |x| - 0.5 & otherwise \end{cases} \qquad (2-7)$$

画出L_1、L_2、SmoothL_1损失函数图像如图 2-12 所示：

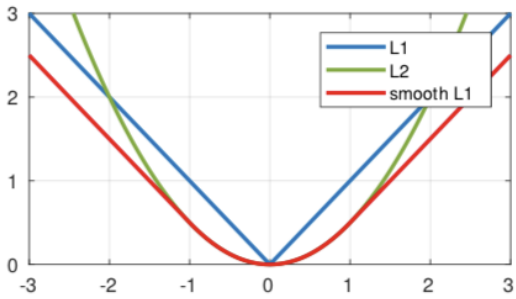


图 2-12 三种损失函数图像

由图可见，在训练初期，网络回归的框与真实标签中的框差距较大时，损失函数的输入 x 会很大，此时梯度保持为 1，使得训练平稳收敛，不至于波动过大。而训练后期，网络的回归框已经达到了一定的精度，损失函数的输入变得很小，此时梯度越来越小，趋于平缓，使得训练能够收敛至极小。综上所述，SmoothL_1综合了L_1和L_2损失的优点，为检测框坐标的回归精度带来一定的提升。

Fast R-CNN 较于 R-CNN 有了速度上质的飞跃，使得基于 CNN 的目标检测达到了准实时的地步，但是也还存在一定的局限，Selective Search 算法占据了检测过程的大部分时间（生成 Proposal Region 大约两三秒，而特征提取+分类所需时间不到一秒），这无法满足工程应用中的实时需求，并且 Selective Search 算法独立在网络训练和推理之外，并没有实现端到端训练，之后的 Faster RCNN[26]的改进之一便是此点。Faster RCNN 设计了 Region Proposal Networks（RPN），首次利用 CNN 代替了 Selective Search 算法来生成候选区域，整体架构如图 2-13 所示：

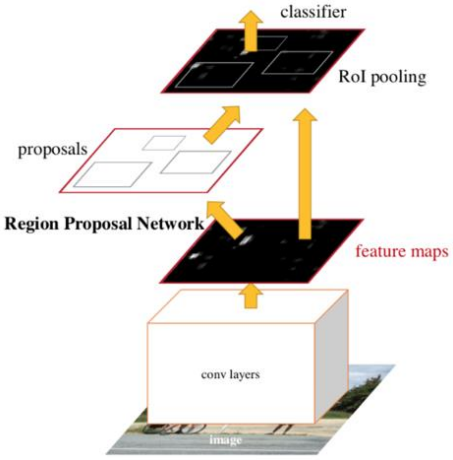


图 2-13 Faster R-CNN 架构图

可以看出与 Fast R-CNN 不同，Faster R-CNN 是完全的端到端训练，输入一张图片，网络的输出便是目标检测框的类别及坐标。其中最大的改进就是提出 RPN 网络，RPN 和 ROI Pooling 共用同一个 CNN 卷积得到的特征图，接下来使用多个不同的模板框在特征图上滑动然后送入 cls 分支和 reg 分支，这里的模板被称为锚点（Anchor），不同长宽、不同比例的锚点以一定的步长在原图上进行一次滑动，就对应 RPN 中的一个待回归的区域，在 RPN 中 cls layer 分支只用于预测候选区域中是否包含物体，reg layer 分支对坐标偏移进行回归，proposal layer 综合两个分支的结果输出最终的区域建议（proposals）。如图 2-14 为 RPN 网络的流程示意图：

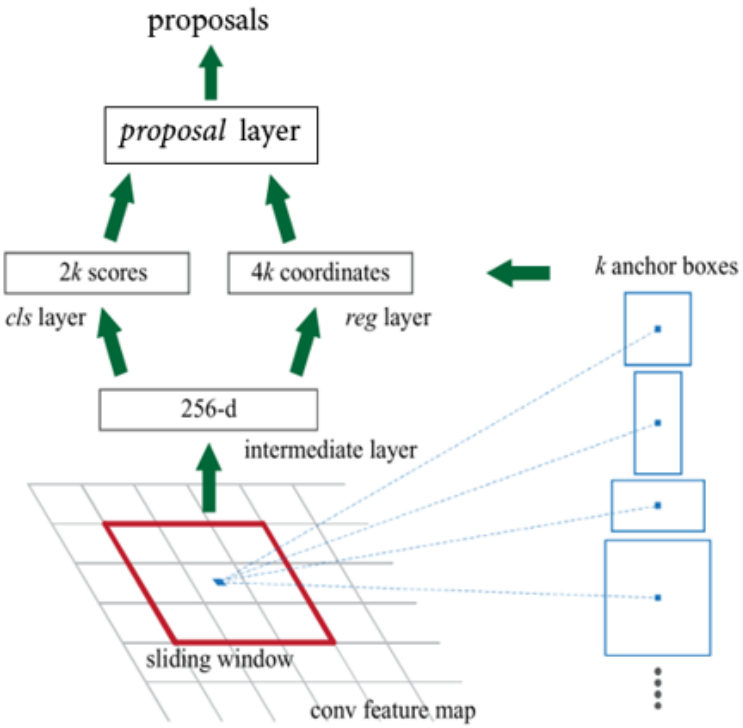


图 2-14 RPN 流程示意图

Faster R-CNN 中的 ROI 层的作用没有改变，将不同大小的输入转换为长度统一的输出，然后送入后续的全连接层和分类和回归分支网络中，输出感兴趣区域所属的类和位置坐标。R-CNN 处理一张图片需要 50 秒，Fast R-CNN 需要 2 秒，而 Faster R-CNN 只需要 0.2 秒！达到了实时处理的要求，并且精度并没有丢失，因此在实际工程使用的中对速度和精度要求都很高的情况下常常选择 Faster R-CNN 作为检测模型。

2.1.3 单阶段模型

目标检测的另一大分支就是单阶段（one-stage）模型，其特点就是模型端到端训练，不存在显式地生成区域建议的步骤。2016 年，Joseph Redmon 等人首次提出了单阶段训练网络 YOLO，相较于两阶段模型，YOLO 不使用先生成目标候选区域，再对候选区域进行分类回归的方法，而是直接将图像均匀划分为多个网格，每个网格预测 n 个包围框和 m 个类别的置信度（n 和 m 为超参数，根据实际任务而定），置信度反映了目标是否为对应类别的概率。如图 2-15 为 YOLO 的预测策略：

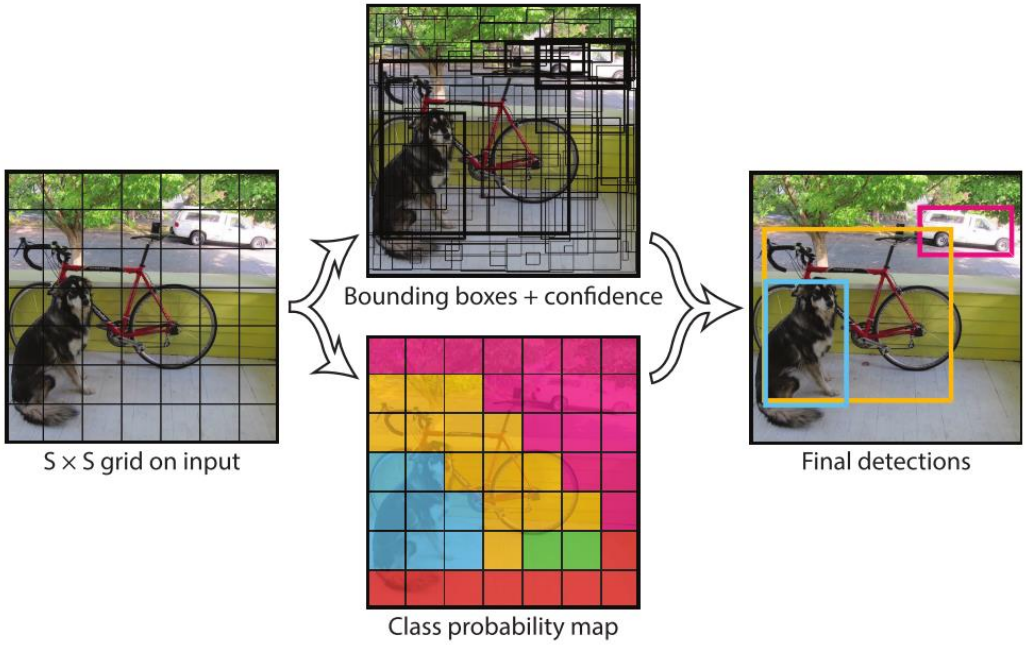


图 2-15 YOLO 预测流程图

YOLO 缺点是单个网格只负责单个目标，若网格内存在有多个目标，则无法全部检出，所以对于密集的目标检测效果不好。

SSD[27]相对于 YOLOv1 采用了更新的设计理念，首先采用了多尺度的特征图对目标进行检测，由于池化层的存在，网络深度越深，小目标的信息丢失越多，所以使用小的特征图对于小目标的检测十分不利。SSD 采用多尺度的特征图，在低层检测小目标，高层检测大目标，提高了网络的检出性能。其次，SSD 放弃了参数量很大的全连接层，使用参数量更少的卷积层来进行特征信息融合和提取，并且 SSD 继承了锚点的思想，设置了不同长宽比的先验框，在训练过程中，每个先验框与标签真实框计算 IOU，若大于人为设定的阈值，则两个框互相匹配。如图 2-16 为 SSD 与 YOLO 的网络架构对比图：

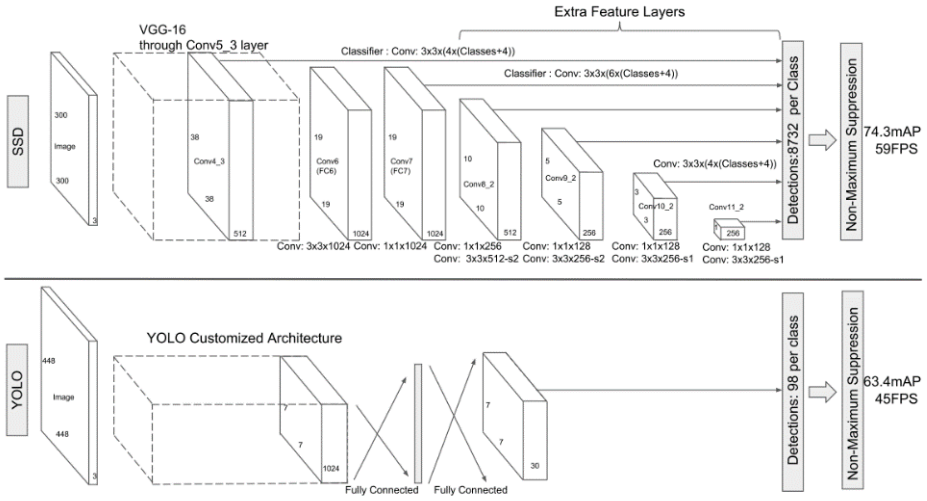


图 2-16 SSD 与 YOLO 架构对比图

Joseph 等人对 YOLOv1 进行了改进，提出了更快更好的 YOLOv2 检测框架，其中引入了 K-Means 算法对 Anchor 进行聚类，本文第三章中也沿用了该技巧对遥感目标网络进行优化设计，因此本文在此先对 K-Means 算法的原理进行介绍。K-Means 模型基于一个距离假设条件，即对于给定的样本集，假设其中存在多个可划分的类别，且相同类别的数据之间距离更近，即数据之间的相似度与他们之间的距离成反比，距离越近越相似，距离越远越不相似。因此其基本思想就是随机初始化 K 个聚类中心，这里的 K 为人工给定的数值，表示最后想要得到的分类个数，接下来，计算样本与 K 个聚类中心的距离，将样本分配给最近的聚类中心所在的类别，处理完所有样本后，计算每个类别的中心点，将其作为新的聚类中心，重复上述操作，直到聚类中心不再移动或移动距离小于给定的阈值。表 2-1 为 K-Means 算法流程：

表 2-1 K-Means 聚类算法

算法 2-1 K-Means 聚类算法
Input：样本集 D=$\{x_1, x_2, \ldots, x_m\}$; 聚类簇数 K 最小更新阈值 T **Algorithm**： 1. 从 D 中随机选择 K 个样本作为初始聚类中心$\{\mu_1, \mu_2, \ldots, \mu_k\}$ 2. **repeat** 3. 令$C_i = \emptyset$ $(1 \leq i \leq K)$

4. **for** j=1, 2, ⋯, m **do**
5. 计算样本x_j与各聚类中心μ_i $(1 \le i \le K)$的距离$d_{ji} = ||x_j - \mu_i||_2$
6. 根据距离最近的聚类中心确定x_j的簇标记：$\lambda_j = \underset{i\in(1,2,\dots,K)}{\operatorname{argmin}} d_{ji}$
7. 将样本x_j划入相应的簇：$C_{\lambda_j} = C_{\lambda_j} \cup \{x_j\}$
8. **end for**
9. **for** i=1,2,…,K **do**
10. 计算新聚类中心：$\mu'_i = \frac{1}{|C_i|}\sum_{x\in C_i} x$
11. **if** $\mu'_i \neq \mu_i$ **then**
12. 将当前聚类中心μ_i更新为μ'_i
13. **else**
14. 保持当前聚类中心不变
15. **end if**
16. **end for**
17. **until** 当前均值向量均未更新或更新值小于阈值 T

Output：簇划分 C=$\{C_1, C_2, \dots, C_K\}$

2.2 半监督学习方法概述

按照训练数据的组成不同，可以将机器学习划分为三大类，分别为监督学习、半监督学习和无监督学习，如图 2-17 为三种学习方式的训练数据划分：

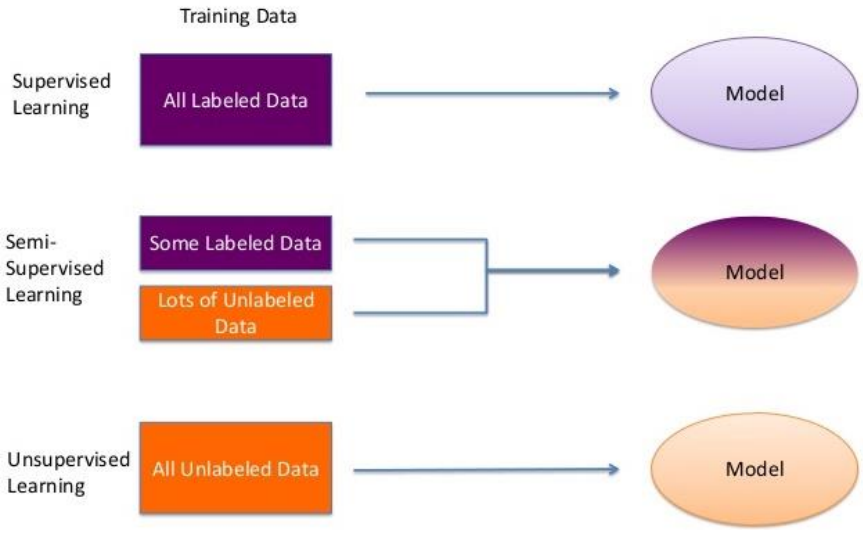


图 2-17 三种学习方式的训练数据划分

监督学习的所有训练数据全部由标签数据组成，半监督学习的训练数据由少部分标签数据和大量的无标签数据组成，而无监督学习的训练数据全部由无标签数据组成。所谓“监督”就是指利用标签中携带的信息对模型的学习方向进行引导，而半监督学习的目标就是，除了利用真实标签携带的信息外，还要对无标签样本中隐含的信息进行挖掘，在样本数据量很大的情况下，我们可以学习到隐藏在数据中的数据空间分布，利用这部分信息来帮助提升模型的性能。

半监督学习理论能够成立依赖于三大假设——独立同分布假设、聚类假设和流形假设。独立同分布假设是指训练数据中的无标签样本与有标签样本都应该是从总体中独立同分布采样得到。聚类假设是指在同一簇中的样本应当属于同一个类，对于簇的定义就是在空间分布中明显存在样本聚集的区域，这个假设可以用另一种等价方式表达，就是网络最终学习到的决策边界应该尽量穿过样本点较少的空间分布区域，这样会尽可能避开样本点聚集的区域，减少误分类的情况。流形假设是指当特征的维度非常高时会出现维度诅咒的问题，此时在高维空间的密度和距离概念趋于无效，则上述两个假设都难以成立。流形假设则假设高维空间的样本可以映射到一个低维空间的流形结构上，在局部邻域内的样本拥有相似的输出值。在满足这三大假设的前提下，半监督学习算法就可以通过无标签样本带来的辅助信息来探明样本空间中的空间分布情况，使得原本密度区别并不明显的区域变得清晰可分，数据量越大的情况下，某些区域的样本会变得明显稠密，而稀疏区域的增量并不明显，从而指引算法对决策边界进行调整，使其尽量通过样本数据分布比较稀疏的区域。图 2-18 为无标签样本帮助学习算法调整决策边界的例子。

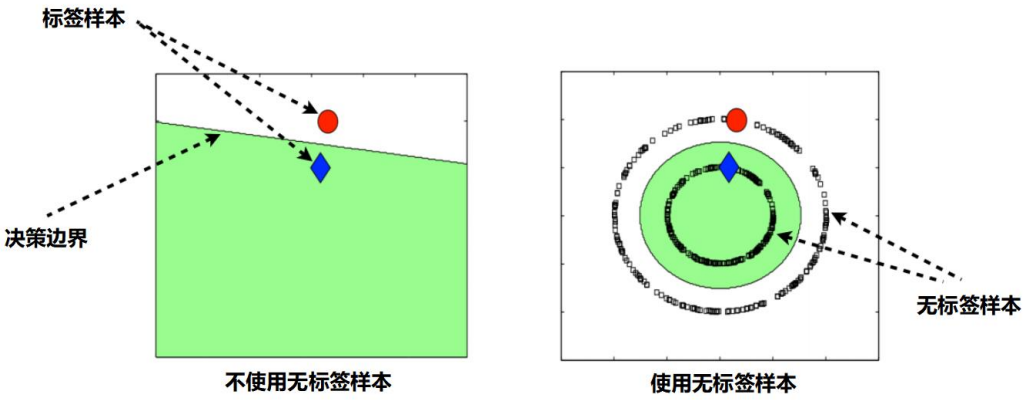


图 2-18 无标签样本指导决策边界调整

在左图只有两个标签样本的情况下，只使用两个点之间的一条直线就可以将两个类别很好地区分，而加入无标签样本后，两个类的数据空间分布变得明确，决策边界也从直线调整为曲线。因此，使用无标签样本可以提高模型的分类准确性。

下面将介绍两种适用于深度学习网络的半监督学习方法。

2.2.1 自训练方法

自训练（self-training）方法是一种简单实用的启发式方法，其基本思想是先在有标签样本上训练得到一个基本模型，然后使用基本模型对无标签样本进行预测，根据某种算法将预测结果中准确度较高的结果保留，作为伪标签（pseudo label），再将伪标签样本与标签样本组合起来，使用标准的监督学习方法对模型进行再训练。自训练方法流程如图 2-19 所示：

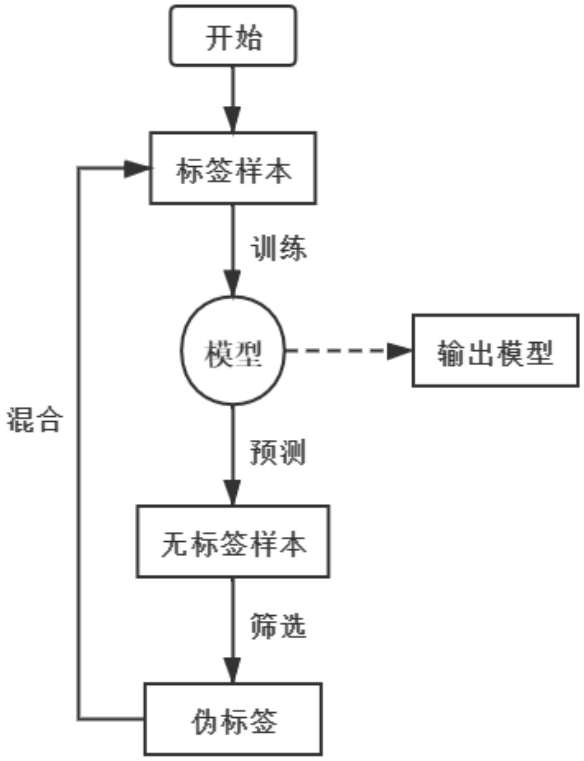


图 2-19 自训练方法流程图

2.2.2 一致性正则化方法

一致性正则化方法是聚类假设的具象化体现，其核心思想就是对于一个输入样本，即使受到微小的噪声干扰，其输出都应该是一致的，因为样本受到微小干扰相当于在数据分布空间内的微小摆动，而分布空间中紧密相邻的点，应当拥有同样的标签。

2.2.2.1 π 模型方法

π 模型是 2017 年 Laine 等人提出的一个半监督学习算法[19]，其算法流程如图 2-20 所示：

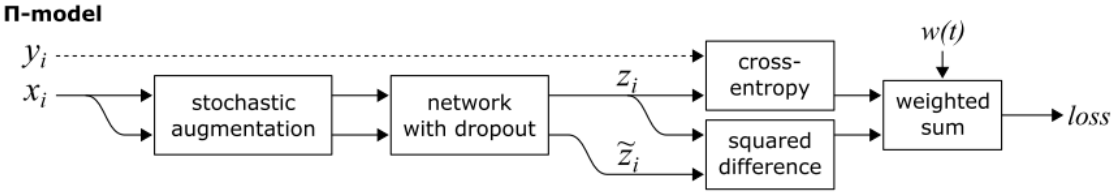


图 2-20 π 模型算法流程图

对于同一个输入x_i，在训练阶段进行两次前向运算，前向运算中包含随机的数据增强操作和 dropout 操作，这两个操作分别带来数据的扰动和模型的扰动，因此获得两次不同的前向运算结果Z_i和$\tilde{Z}_i$，π 模型的损失函数除了Z_i与标签y_i的有监督损失外，还添加了Z_i与$\tilde{Z}_i$的一致性损失，损失函数如公式（2-8）所示：

$$\text{loss} = -\frac{1}{|B|}\sum_{i\in(B\cap L)} log z_i[y_i] + w(t)\frac{1}{C|B|}\sum_{i\in B}||z_i - \tilde{z}_i||^2 \qquad (2-8)$$

其中前半部分为有监督训练损失，采用交叉熵损失函数，B 为训练时的 batch 集合，L 为有标签样本集合，当 i 属于 B 集合与 L 集合的交集时才进行计算；后半部分为半监督的一致性损失，采用均方差损失函数，用于评估所有的数据，既包含有标签样本，也包含无标签样本。w(t)为时变系数，是迭代次数的函数，由于网络初始训练阶段的预测十分不准，此时的一致性损失权重应该设置得非常小，到后期再慢慢增大。

2.2.2.2 Mean Teacher

为了解决 π 模型不稳定的问题，Laine 等人提出 Temporal Ensembling 的方法，主要思想就是对同一个无标签样本输入，在训练阶段只进行一次前向运算，另一次通过之前所有 epoch 对该样本的输出进行指数滑动平均（Exponentially Moving Average，EMA）来获得，这样前向运算的次数减少一半，速度提升两倍，是典型的空间换时间方法。通过 EMA 来平均之前 epoch 的输出隐式地利用了集成学习的思想，epoch 越大时模型输出越稳定。但是 Temporal Ensembling 要记录下当前 epoch 中所有测试图片对应的伪标签，在数据集大时所需存储空间巨大，并且 Temporal Ensembling 每个 epoch 才进行一次滑动平均，更新的间隔过长，模型的权重已有了较大的变化，这样计算出来的一致性损失误差较大，于是 Tarvainen 等人提出了 Mean Teacher 的方法[21]对其进行改进。Mean Teacher 使用了知识蒸馏中常用的教

师-学生双网络模型，其中的学生模型就是正常训练的模型，而教师模型通过比 epoch 粒度更细的 step 粒度上对学生模型权重的指数滑动平均来获得（每个 epoch 中有多个 step，step 是训练的最小单位，就是模型一次前向运算和后向传播的过程）。算法的流程图如图 2-21 所示：

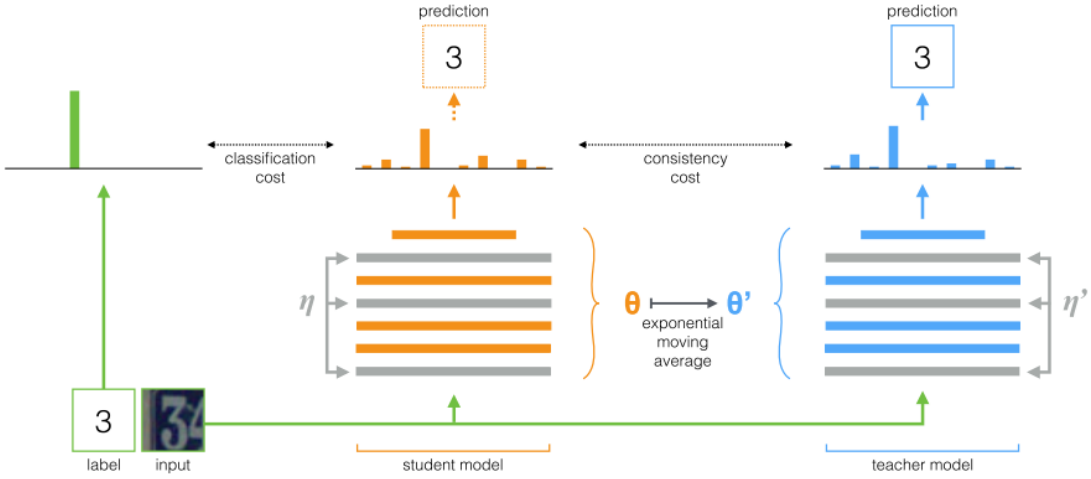


图 2-21 Mean Teacher 算法模型

定义训练步骤中教师模型更新的公式如公式（2-9）所示，其中α是平滑系数：

$$\theta'_t = \alpha\theta'_{t-1} + (1-\alpha)\theta_t \qquad (2-9)$$

2.3 本章小结

本章介绍了本文研究所依赖的基础理论，主要分为两部分，第一部分首先对卷积神经网络常用层的功能和特性做了概述，然后详细介绍了目标检测领域的两大类模型及其经典的算法模型。第二部分介绍了适用于神经网络的两类半监督学习方法，自训练方法和一致性正则化方法。有了本章的基础理论铺垫，后续章节将基于此开展相关实验。

第三章 针对遥感目标特性的网络优化设计

本章将会根据遥感目标的特性对 Faster R-CNN 网络进行优化设计。首先，本章将对本文所用的遥感数据集的构建和分布进行介绍，并对遥感图像目标的特点进行分析，然后针对遥感目标的特点，分析 Faster R-CNN 网络本身的不足，提出四点网络优化策略，最后通过实验对提出的优化模块效果进行验证与分析。

3.1 数据集构建与分析

3.1.1 数据集构建

本文研究的是半监督学习方法对遥感目标检测的模型提升效果，因此数据集首先应该是航空拍摄的遥感图像，其次应该有少量的标签样本和大量的无标签样本。通常从卫星传回地面的遥感图像都是以景为单位，一景卫星遥感图像所包含的面积在数千-数万平方公里的范围内，如北京揽宇方圆的 SPOT5 商业卫星一次拍摄的范围保持在 60 公里*60 公里，一景图像则是 3600 平方公里。商业卫星或军事卫星拍摄的遥感图像都难以获得，并且原始图像传回地面后还要经过一系列的图像处理、切片加工等步骤，才能投入使用。好在随着遥感行业的不断发展，越来越多的公开遥感数据集可供研究选择，如 LEVIR 数据集[28]，HRRSD 数据集[29]，UCAS-AOD 数据集[30]，RSOD 数据集[31]等，不同的数据集中有不同的遥感目标，如飞机、轮船、立交桥、城市建筑、油罐、港口、棒球场、网球场等，数量也都各不一致，有的目标只有上百张，为了方便下一章节中划分出大量的无标签样本，并控制半监督学习的学习难度，因此本文数据集只挑选整合了遥感目标中具有代表性的、数据资源最丰富的飞机和轮船目标。表 3-1 为本文所用数据集的整体分布情况：

表 3-1 数据集整体分布

类别	训练集	测试集
飞机	4286 张	451 张
轮船	3162 张	324 张
总计	7448 张	775 张

3.1.2 数据集特点分析

遥感数据集通常都是通过离地至少上千米的航空飞机或卫星进行拍摄，由于拍摄位置和视角的特殊性，遥感数据与 VOC 和 COCO 等常见目标数据集有着很

大区别，分析其特点有：

1. 尺度差异大。由于拍摄高度和镜头分辨率的不同，且地面上同类型目标的规格也有较大差异，所以遥感数据集中目标尺度差距较大，目标十几像素到几百像素都有可能。
2. 方向角度多。由于视角为鸟瞰视角，目标的方向受当时摆放角度决定，（不同于人脸、行人数据集中的目标都是竖着的），因此遥感数据集中存在着多种多样的方向，检测器应该对方向具有一定鲁棒性。
3. 目标小而密。在某些场景下，遥感图像会出现小而密集的目标，比如轮船目标，在靠岸停放的场景下船只都是紧紧依靠的。
4. 背景复杂性。航空遥感图像的视野较大，图像中除了目标还有大量干扰物体。

图 3-1 为本遥感数据集样例展示，图 3-2 为常见目标检测数据集 VOC 的数据样例展示，可以看出相比于常见目标数据集，遥感数据集目标具有上述的多尺寸、多方向、目标密集、背景复杂的特点。

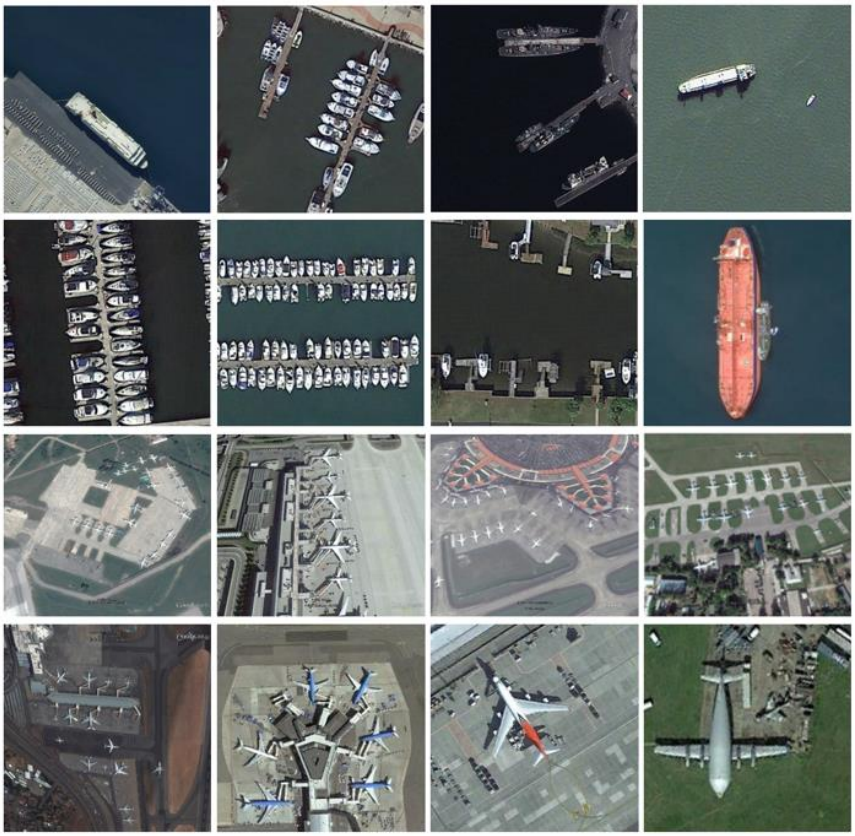


图 3-1 遥感数据集样例展示

图 3-2 VOC 数据样例展示

3.2 针对遥感目标的网络优化

由 3.2 节分析可得，遥感数据集与常见目标数据集存在较大差别，因此本文将针对遥感目标的特点对 Faster R-CNN 进行优化设计。

3.2.1 基于 FPN 的多尺度预测结构

Faster R-CNN 的网络结构如图 3-3 所示：

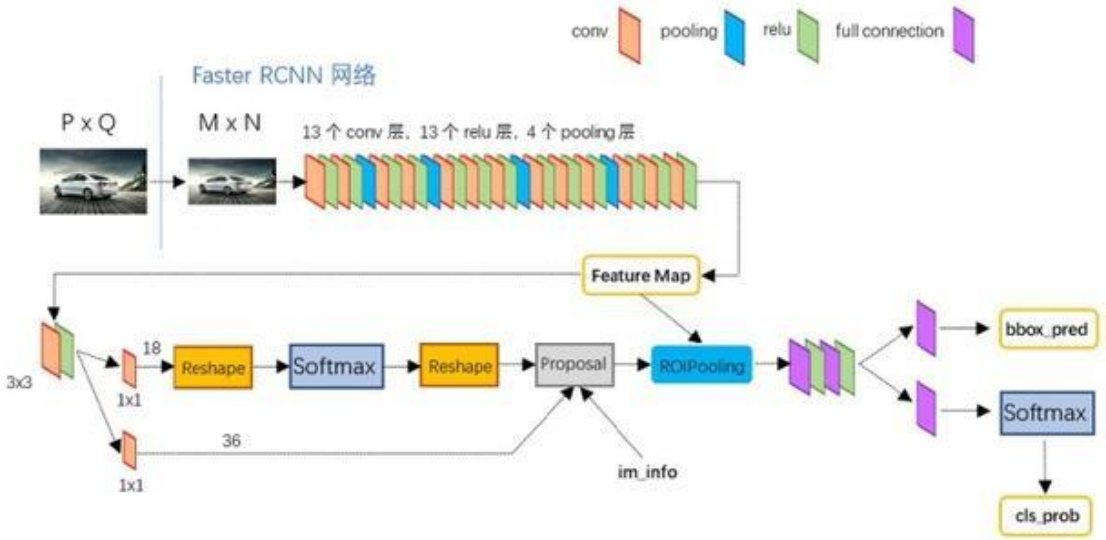


图 3-3 Faster R-CNN 网络结构图

从结构图中可以看出，Faster R-CNN 的特征提取网络使用了 13 个 conv 层，13 个 relu 层，4 个 pooling 层，然后将最后一层的特征图输入 RPN 网络进行分类预测和候选框回归。这样的结构是不利于小目标检测的，因为每经过一次 pooling

层，图像会缩小二分之一，经过 4 层 pooling 层之后图像大小已经下采样到原始输入图像大小的十六分之一，如果原图中目标规模低于 16 像素，经过池化后在特征图中已经几乎没有了对应的特征信息，模型很难成功检出。除此之外，随着卷积神经网络的深度增加，学习到的特征更加抽象且全局化，Zeiler 等人[32]对多层卷积神经网络的特征图进行了详细的可视化，发现低层的特征都是边缘、颜色、纹理等简单的局部特征，随着网络深度的增加，感受野越来越大，高层特征也更加抽象化，语义信息更加丰富。总的来说，底层特征因为包含丰富的几何轮廓和位置信息，对小目标的检测更有利，高层特征因为感受野足够大，语义信息丰富，对大目标的检测更有利。对于目标尺度差异较大的数据，我们应该使用不同层次的特征图进行预测，而不是只使用单一的特征图。Tsung-Yi Lin 等人创造的特征金字塔网络 FPN（Feature Pyramid Network[33]）网络结构提出了一种高效的多尺度特征层融合策略，如图 3-4 所示：

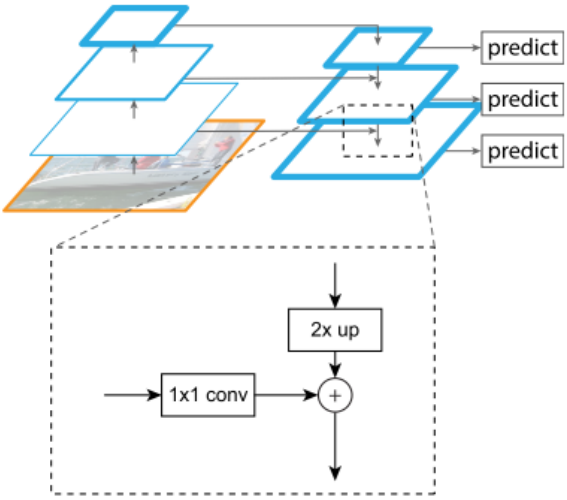


图 3-4 FPN 结构示意图

FPN 包括自下而上，自上而下和横向连接三个部分，自下而上即是网络的正向推理过程，经过卷积层的累积，特征逐渐从具象特征开始变得抽象。自上而下即是通过 2 倍最近邻上采样的方式将上一层特征放大到与横向连接过来的下一层特征一样的大小，横向连接时低层特征要先进行 1x1 卷积，主要是为了保证进行 add 操作时特征通道维度相同，自上而下的特征图与横向连接的特征图相加后还要进行一个 3x3 卷积，特征通道数保持不变，主要是为了消除上采样的混叠效应（Aliasing Effect）。本文将 Faster R-CNN 的主干网络从 VGG 改为分类精度更高的 ResNet50，并从 ResNet50 与 RPN 网络之间加入 FPN 结构，ResNet50 的网络结构如图 3-5 所示：

layer name	output size	config
conv1	1/2	7x7,64, stride=2
conv2_x	1/4	3x3 max pooling, stride=2 $\begin{bmatrix} 1\times1,64 \\ 3\times3,64 \\ 1\times1,256 \end{bmatrix} \times 3$
conv3_x	1/8	$\begin{bmatrix} 1\times1,128 \\ 3\times3,128 \\ 1\times1,512 \end{bmatrix} \times 4$
conv4_x	1/16	$\begin{bmatrix} 1\times1,256 \\ 3\times3,256 \\ 1\times1,1024 \end{bmatrix} \times 6$
conv5_x	1/32	$\begin{bmatrix} 1\times1,512 \\ 3\times3,512 \\ 1\times1,2048 \end{bmatrix} \times 3$
classifier	1x1	average pooling, 1000-d fc, softmax

图 3-5 ResNet50 网络结构

在 ResNet50 和 RPN 之间加入 FPN 结构后的网络结构如图 3-6 所示：

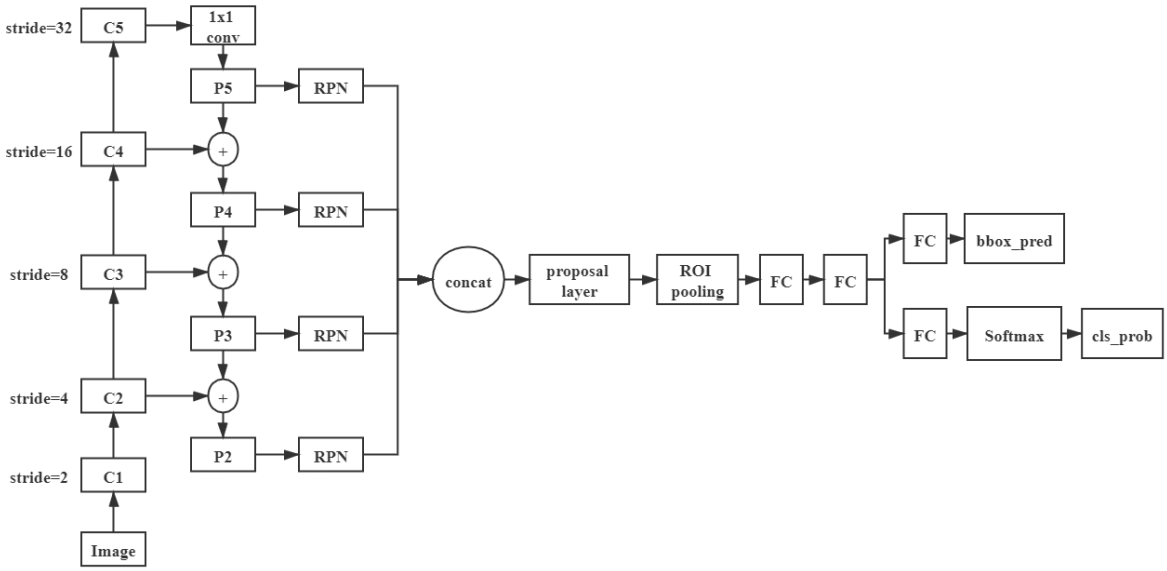


图 3-6 加入 FPN 后的网络结构

其中 C1 为 conv1 模块的缩写,C2、C3、C4、C5 以此类推。考虑到 C1 尺寸太大且特征过于低级对目标分类不利所以没有使用，最终产生 P2、P3、P4、P5 四个尺度的特征金字塔进行检测,其中 C5 到 P5 只进行了一个 1x1 卷积进行通道降维，没有改变特征图大小。值得注意的是不同于 Faster R-CNN 在同一尺度上使用全部 Anchor 进行预测，本文加入 FPN 后的网络将在不同尺度的特征图使用不同尺寸的 Anchor 预测，小 Anchor 由分辨率较高的特征图负责，大 Anchor 则由分辨率较低的特征图负责，RPN 网络单独针对每个特征图进行 proposal 的预测，然后将获得的区域建议拼接（concat）起来，再一同通过 proposal layer 获得最终的 proposal。在原本的 Faster R-CNN 中输出的特征图只有一个，ROI pooling 层直接将 proposal 映射到该特征图上将其裁取出来即可，而加入 FPN 架构后特征图为集合{P2，P3，P4，P5}，将原图的 proposal 映射到哪个尺度的特征图成为问题，本文采用公式（3-1）决定：

$$\mathrm{k} = \lfloor k_0 + \log_2\left(\sqrt{wh}/300\right) \rfloor \qquad 2 \le \mathrm{k} \le 5 \qquad (3-1)$$

其中k_0是面积为 300x300 的 proposal 应该映射到的特征图层级,本文中k_0=5。通过上述公式可以计算出每个层级的特征图复杂预测的最大和最小 proposal 面积，如表 3-2 所示：

表 3-2 特征图对应 proposal 面积

特征图层级	最小 proposal 面积	最大 proposal 面积
P5 stride=32	300x300	输入图像面积
P4 stride=16	150x150	300x300
P3 stride=8	75x75	150x150
P2 stride=4	1x1	75x75

将 proposal 除以每个特征图层级对应的步长，就可以算出映射到对应的特征图上的大小，基本上每个特征图对应的面积边长都是 10~20 像素左右，且最高层 P5 和最底层 P2 负责预测过大和过小的目标，说明分配是较为均匀、合理的。

3.2.2 基于 K-Means 的 Anchor 生成

在 Faster R-CNN 中设计了 9 个不同大小和宽高比的 Anchor，具体设置如表 3-3 所示：

表 3-3 Faster R-CNN Anchor 设置

Base Anchor	Ratios	[w, h, x_centre, y_centre]
16x16	23x12 (2:1)	[184, 96, 7.5, 7.5] scale=8
		[368, 192, 7.5, 7.5] scale=16
		[736, 384, 7.5, 7.5] scale=32
	16x16 (1:1)	[128, 128, 7.5, 7.5] scale=8
		[256, 256, 7.5, 7.5] scale=16
		[512,512, 7.5, 7.5] scale=32
	11x22 (1:2)	[88, 176, 7.5, 7.5] scale=8
		[176, 352, 7.5, 7.5] scale=16
		[352, 704, 7.5, 7.5] scale=32

从表中可以看出，Faster R-CNN 的 Anchor 生成方式是基于一个 Base Anchor，按照{1:1, 1:2, 2:1}的比例进行长宽比的变换，然后按照{8,16,32}的尺度进行放大，从表中第三列可以看出 Faster R-CNN 中最小的 Anchor 为 128x128 大小，这样的设置明显不适用于我们的遥感数据集，尽管网络在回归过程中会自动对 Anchor 进行偏移缩放调整，但是如果 Anchor 的大小和真实目标的大小差距过大，仍然会对模型的回归框效果产生影响，因此本文将采用 K-Means 聚类的方法对数据集 Ground Truth 框的大小进行搜索，找出符合本数据集的 Anchor 设置。这样聚类出来的 Anchor 设置排除了人工指定带来的主观因素影响，更加符合数据集的样本空间分布情况。

首先要确定参与聚类的元素。数据集标签中每个框由五个元素定义，即{cls、x、y、w、h}，cls 是目标所属类别，x、y 为真实框左上角的横、纵坐标，w、h 为真实框的宽度、高度。由于 anchor 本身没有类别的区分，且 anchor 的位置是在网络下采样倍数确定之后就固定了的，所以对{cls、x、y}元素进行聚类是没有意义的，只有 anchor 的宽度和高度会影响初始的 anchor 分布情况，因此我们只对真实框的宽高进行聚类。

确定好聚类对象后，如何描述聚类对象之间的“距离”也是一个问题，在 K-Means 算法中是使用欧式距离进行度量，数学公式如公式（3-2）下：

$$d_{ji} = ||x_j - \mu_i||_2 \quad (i = 1,2,\dots,K; j = 1,2,\dots,N) \qquad (3-2)$$

其中 K 为聚类中心数量，N 为待聚类样本数量，μ_i表示第 i 个聚类中心，x_j表示第 j 个待聚类样本， x和μ都是维度相同的向量。我们也可以将真实框的宽高抽象化，将其视为二维空间中的坐标点，然后使用欧式距离刻画两点之间的“距离”，即两个框之间的相似度，求得的距离越小，说明两个框之间越相似。但是这样的距离度量方式存在着一定问题，就是大框类别会比小框类别产生更大的误差，图 3-7 将对这个问题举例进行说明：

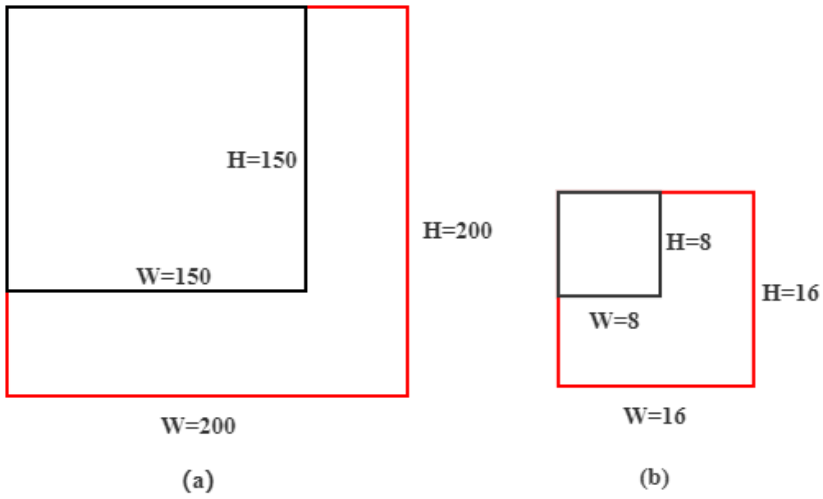


图 3-7 欧氏距离对框进行度量示意图

图 3-7 中红色框为算法的聚类中心，黑色框为某个样本，我们将两个框的左上角对齐便于对两个框进行直观对比。根据前文提出的抽象方式，我们将图（a）中的聚类中心定义为μ=(200,200)，样本定义为 x=(150,150)，根据公式（3-2）对两个框的距离进行计算，得到式子（3-3）：

$$d_{(a)} = ||x - \mu||_2 = \sqrt{(200-150)^2 + (200-150)^2} \approx 71 \qquad (3-3)$$

同样地，我们将图（b）中的聚类中心定义为μ'=(16,16)，样本定义为x'=(8,8)，根据公式计算得到式子（3-4）：

$$d_{(b)} = ||x' - \mu'||_2 = \sqrt{(16-8)^2 + (16-8)^2} \approx 11.3 \qquad (3-4)$$

得到的结果表示，图（a）中的两个框间的“距离”远远大于图（b）中的两个框，即图（b）中的框比图（a）中的框更加相似，但是肉眼观察就可以发现事实并非如此，明显图（a）中黑色框的面积占比相比于图（b）更大。因此，我们不能使

用欧式距离作为两个框相似度的度量，必须采用其他方法来刻画。在 3.2.2 节中，我们介绍了一种衡量两个框重叠度的计算公式，即 IOU 公式，其值与框的宽高没有直接联系，只与两个框之间的交并比紧密相关，并且 IOU 本来就是模型输出的检测框与真实框的相似度的衡量标准，因此很适合作为 K-Means 对各个真实框间“距离”的度量方式。但是，IOU 的数值含义是当 IOU 值越大时，两个框越相似，为了满足 K-Means 中“距离越远越不相似”的假设，将距离公式定义为式（3-5）：

$$d_{ji} = 1 - IOU = 1 - \frac{B_{x_j} \cap B_{\mu_i}}{B_{x_j} \cup B_{\mu_i}} \qquad (i = 1,2,\dots,K; j = 1,2,\dots,N) \qquad (3-5)$$

其中 K 为聚类中心个数，N 为待聚类样本数量，B_{μ_i}表示第 i 个作为聚类中心的框，B_{x_j}表示第 j 个待聚类的框。利用公式对图 3-7 中的框进行计算得到式（3-6）（3-7）：

$$d_{(a)} = 1 - \frac{B_x \cap B_\mu}{B_x \cup B_\mu} = 1 - 0.5625 = 0.4375 \qquad (3-6)$$

$$d_{(b)} = 1 - \frac{B_{x'} \cap B_{\mu'}}{B_{x'} \cup B_{\mu'}} = 1 - 0.25 = 0.75 \qquad (3-7)$$

基于 IOU 的距离计算方式得到了与基于欧式距离相反的结果，但是更加地符合实际，因此我们将基于 IOU 的距离替换欧式距离，总结出针对 Ground Truth 框聚类的算法流程如表 3-4 所示：

表 3-4 针对 Ground Truth 框的 K-Means 聚类算法

算法 3-1 针对 Ground Truth 框的 K-Means 聚类算法
Input：数据集 Ground Truth 框集合 G=$\{x_1, x_2, \dots, x_m\}$； 聚类簇数 K 最小更新阈值 T
Algorithm：
1. 从 G 中随机选择 K 个样本作为初始聚类中心$\{\mu_1, \mu_2, \dots, \mu_k\}$
2. **repeat**
3. 令$C_i = \emptyset$ $(1 \le i \le K)$
4. **for** j=1,2,…,m **do**
5. 计算样本x_j与各聚类中心μ_i $(1 \le i \le K)$的距离$d_{ji} = 1 - \frac{B_{x_j} \cap B_{\mu_i}}{B_{x_j} \cup B_{\mu_i}}$

6. 根据距离最近的聚类中心确定x_j的簇标记：$\lambda_j = \underset{i\in(1,2,\ldots,K)}{\operatorname{argmin}} d_{ji}$
7. 将样本x_j划入相应的簇：$C_{\lambda_j} = C_{\lambda_j} \cup \{x_j\}$
8. **end for**
9. **for** i=1,2,…,K **do**
10. 计算新聚类中心：$w'_i = \frac{1}{|C_i|}\sum_{w\in x, x\in C_i} w$，$h'_i = \frac{1}{|C_i|}\sum_{h\in x, x\in C_i} h$
11. $\mu'_i = (w'_i,\ h'_i)$
12. **if** $\mu'_i \neq \mu_i$ **then**
13. 将当前聚类中心μ_i更新为μ'_i
14. **else**
15. 保持当前聚类中心不变
16. **end if**
17. **end for**
18. **until** 当前聚类中心均未更新或更新值小于阈值 T

Output：簇划分 C=$\{C_1, C_2, \ldots, C_K\}$

同时，由于我们的网络中加入了 FPN 结构，不同的特征层负责检测不同大小的目标，因此不需要在每个尺度的特征层上都使用所有的 Anchor，本文设计的是每层预设 3 个 Anchor，共计设置 12 个 Anchor，因此本文将 K-Means 算法中的参数 K 设为 12，最小更新阈值为 2，然后对遥感数据集进行聚类。获得聚类结果后，将较小的 Anchor 分配给低层特征层，将较大的 Anchor 分配给高层特征层，分配结果如表 3-5 所示：

表 3-5 聚类获得的 Anchor 分配表

特征图层级	聚类获得的 Anchor [w, h]
P2	[15, 17]
	[18, 31]
	[25, 16]
P3	[30, 27]
	[40, 40]
	[54, 54]
P4	[57, 92]
	[76, 67]
	[96, 103]
P5	[114, 296]
	[292, 118]
	[293, 299]

可以看出聚类出来的 Anchor 与 Faster R-CNN 手工设计的 Anchor 有着较大区别，聚类获得的 Anchor 在一定程度上反映了数据集中真实目标的大小分布。

3.2.3 基于 MAM 的混合域注意力机制

近两年注意力机制研究方向受到了计算机视觉领域研究人员的广泛关注，其本质思想借鉴了人类的视觉注意力机制，当人类看到一张图片时，会对整张图像进行扫描，将注意力集中在感兴趣的区域，降低对其他不重要区域的关注，调配有限的认知资源对感兴趣区域进行深度分析，以更快更好地获取目标信息。许多动物都存在着这样的注意力机制，因为资源是宝贵的，如何通过分配资源的使用获取最大的效益，是动物的生存之道。图 3-8 展示了人类在看到一张报纸封面时是如何分配注意力资源的，其中颜色越红的区域表示获得的注意力越多，很明显人们会把注意力放在图像中的婴儿脸部和报纸标题以及文章首句，符合人类天生对人脸的敏感和通过标题快速了解文章信息的阅读习惯。

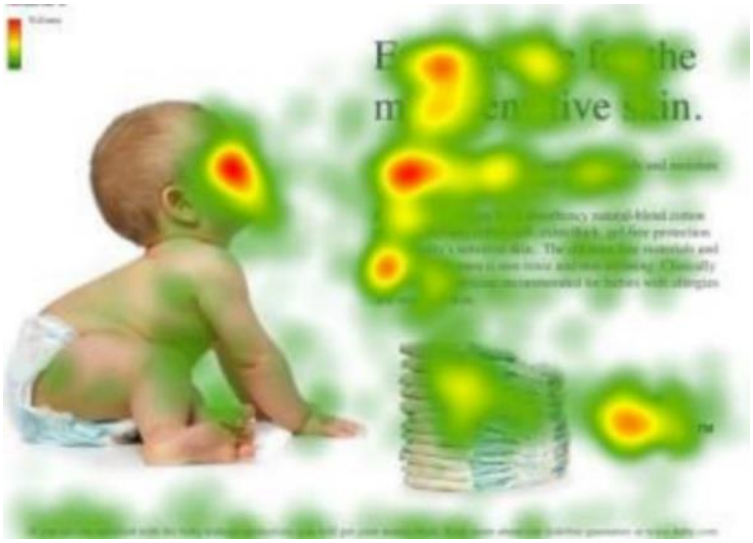


图 3-8 人类注意力分配示例

而深度学习的注意力机制就是为了模拟人类的注意力机制，主要包含两个方面：（1）判断图像中哪些部分的信息是更重要的。（2）将注意力资源分配给那些更重要的部分。

3.2.3.1 空间域和通道域注意力

从注意力域（attention domain）的角度来说注意力机制主要分为三种注意力域：空间域(spatial domain)、通道域(channel domain)和混合域(mixed domain)。

空间域注意力的设计思路和人类视觉注意力机制一脉相承，首先我们知道卷积神经网络中的特征图有着 N*W*H*C 四个维度，N 为训练过程中一个 batch 的图片数量，W 和 H 分别代表特征图的宽度和高度，C 为特征图的通道数。空间域注意力就是在 W*H 的维度上将图像的信息，使用某种变换方式从一个空间映射到另一个空间，并保留感兴趣的目标的信息。Google DeepMind 提出的 STN（Spatial Transformer Network[34]）网络就是具有代表性的空间域注意力网络，论文中提出一种空间变换（spatial transformer）模块（如图 3-9 所示），能够提取出图片中需要关注的区域，同时能够对该区域进行自动的空间旋转、缩放变换，使得输入样本更加容易学习。

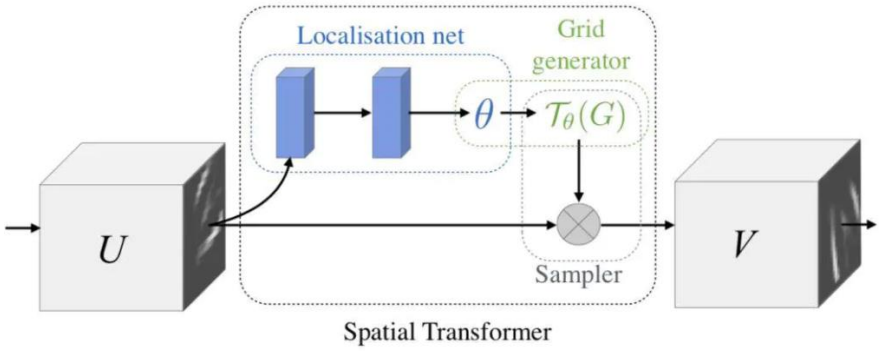


图 3-9 Spatial Transformer 模块

而通道域注意力的设计思路是从特征图的通道（Channel）维度出发的，在输入阶段原始图像通常有 RGB 三种颜色通道，经过卷积层之后，每个卷积核都会生成不同的通道，每个通道其实就表示该图像在不同卷积核上的信号分量。不同的信号分量与关键信息的相关度是有差异的，通道注意力就是针对不同的通道，学习出不同的权重，对重要的信号进行强化，对不重要的信号进行抑制。SENet（Squeeze-and-Excitation Networks[35]）是通道域注意力中非常重要的模型，它提出的 SE 模块，通过挤压（Squeeze）操作对各个特征通道的信息进行全局压缩，然后通过膨胀（Excitation）操作显式地建模特征通道间的相关性，学习出一个 1x1xC 的权重向量，然后在与原始特征图进行 scale 相乘（scale 相乘就是将该向量与特征图在通道维度上一一对齐，然后将学习到的权重与对应的 WxH 大小的特征图上的所有值相乘），增强与目标强相关的特征并抑制相关性不大的特征。如图 3-10 为 SE 模块的

详细结构：

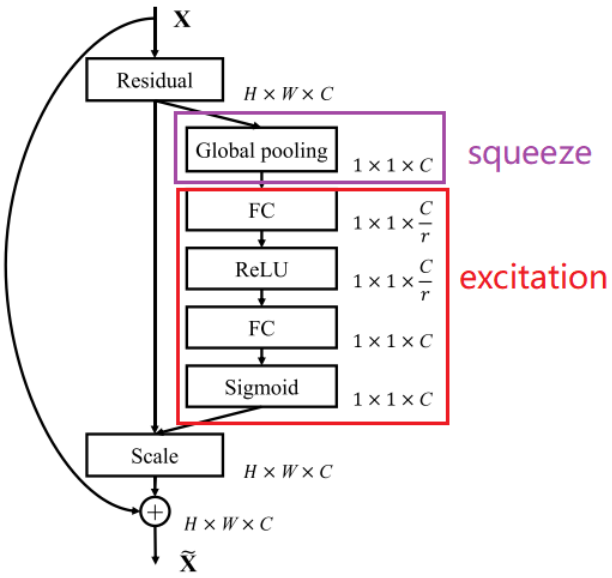


图 3-10 SE 模块

在了解空间域和通道域注意力的设计思路后，对其进行分析，发现两种注意力机制都存在一定的局限性。首先，空间注意力只是在空间层面上对注意力区域进行提取和几何变化，没有利用到通道维度的信息，并且只做几何变化对关键信息的增强有限。而通道注意力在挤压操作中直接对特征图进行全局压缩得到一个实数，将会丢失掉图像中的很多局部细节信息，这样也是不那么可取的。所以综合两种注意力机制思路，就设计出了混合域的注意力模型。

3.2.3.2 MAM 混合域注意力模块

混合域注意力的设计思路就是同时在空间域和通道域上使用注意力机制，17 年 Fei Wang 等人提出的网络[36]通过给残差网络添加掩码(mask)分支来学习权重，同时论文实验证明在空间域和通道域上同时为每个特征学习权重比针对单独的某个域学习权重效果更好，因此本文在 SE 模块的基础上进行修改，分别提出了一种针对通道维度的注意力模块(CAM，Channel Attention Module)，用于学习出 1x1xC 的通道注意力权重，以及针对空间维度的注意力模块（SAM，Spatial Attention Module），用于学习出 HxWx1 的空间注意力权重，两者组合在一起成为混合域注意力模块（MAM，Mixed Attention Module），下面将对上述模块进行详细介绍。

（1） CAM 模块

如图 3-11 为 CAM 模块的具体设计：

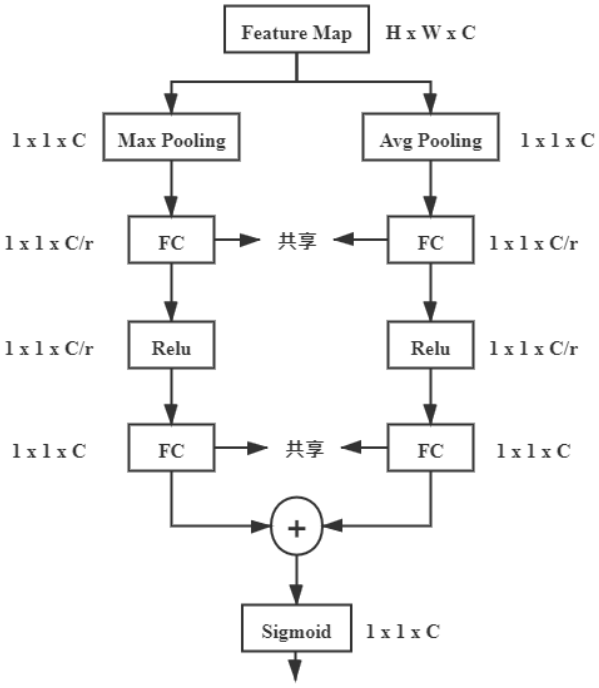


图 3-11 CAM 模块具体设计

CAM 模块的基本组件和 SE 模块保持一致，依然是采用挤压-膨胀的操作学习通道权重，但是 CAM 模块中除了全局平均池化还添加了一个全局最大池化分支，因为全局平均池化获取的是特征图中所有点的平均值，而最大池化得到的是特征图所有点的最大值，是从另一个角度来对特征进行了刻画，相较于只使用平均池化会增加一些更具有区分度的信息。值得注意的是，本文设计的 CAM 模块中两个分支的全连接层是共享权重的，因为从一方面来说权重的本质就是进行特征提取，我们从最大池化输出和平均池化输出中提取出来的特征信息应该是保持一致的，因此应该使用相同的权重，并且共享权重的全连接层在反向传播会同时接收两个分支回传的梯度进行更新，达到了一定的信息融合，也增加了网络的泛化性能。另一方面来说共享权重可以减少训练参数，降低了模型的收敛难度，减少了计算量。出于以上两方面的考虑，我们将两个分支的全连接层设计为共享权重。

（2） SAM 模块

如图 3-12 为 SAM 模块的具体设计：

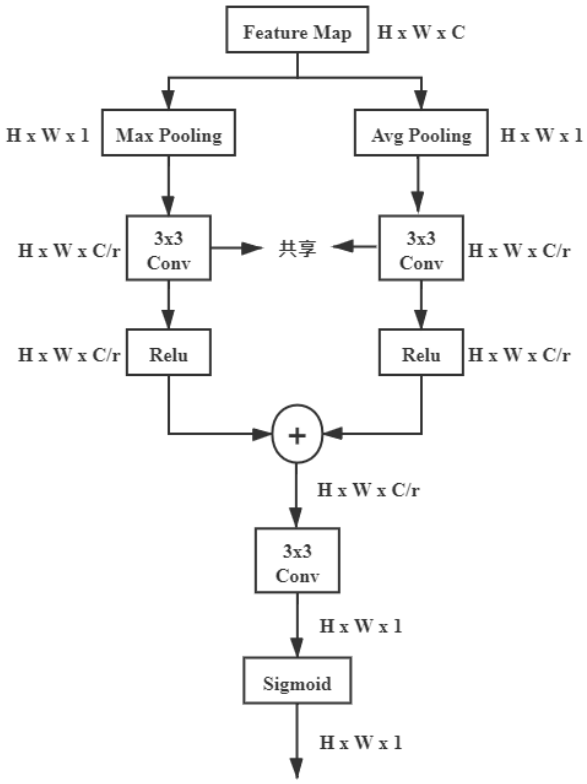


图 3-12 SAM 模块具体设计

SAM 模块借鉴了 CAM 模块的思想，首先在通道维度上分别对特征图进行了全局最大池化和全局平均池化，获得了 HxWx1 的特征图，然后在特征图上使用 3x3 的卷积进行局部特征提取，增强局部信息交流，需要注意的是卷积操作也是共享权重的，目的和 CAM 模块相同。接下来对两个分支的特征图通过 Relu 激活后再相加，获得 HxWxC/r大小的特征图，进行相加而不进行拼接的原因是特征相加，原本响应高的特征与另一个响应高的特征相加会获得更高的响应，而原本响应低的特征与另一个响应低的特征相加获得的增幅并不会很高，一定程度上对高响应特征进行了增强，而这正好符合我们的注意力思想，因此对特征图的整合采取相加的方式。接下来再进行 3x3 卷积，进行通道间的特征融合，将特征通道数压缩至 1，最终经过 sigmoid 后输出 HxWx1 的空间特征图“权重”。

（3） MAM 模块

将 CAM 模块与 SAM 模块组合在一起，就形成了混合域注意力模块 MAM，本文将 MAM 模块按照先后顺序不同再细化为 S_C_MAM 和 C_S_MAM，从命名中就可以看出 S_C_MAM 是 SAM 模块在前，CAM 模块在后，C_S_MAM 是 CAM

模块在前 SAM 模块在后，如图 3-13、图 3-14 为两个模块的具体设计：

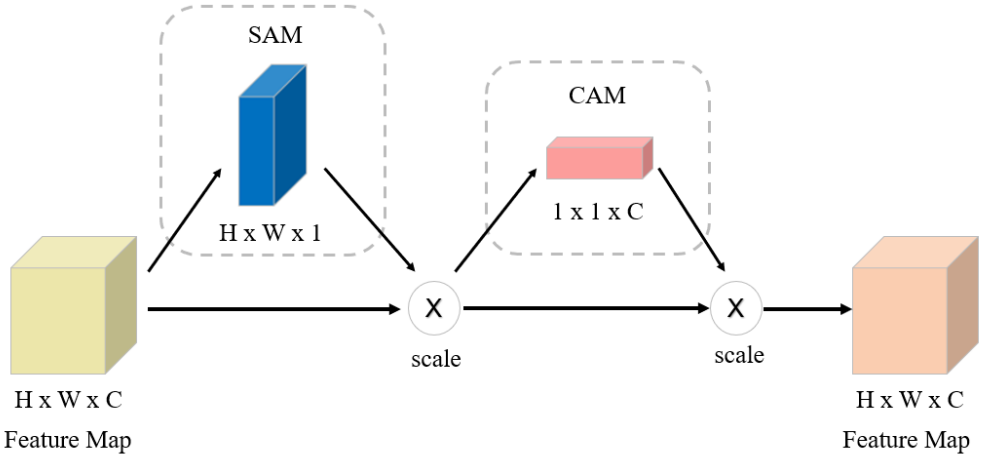


图 3-13 S_C_MAM 模块设计

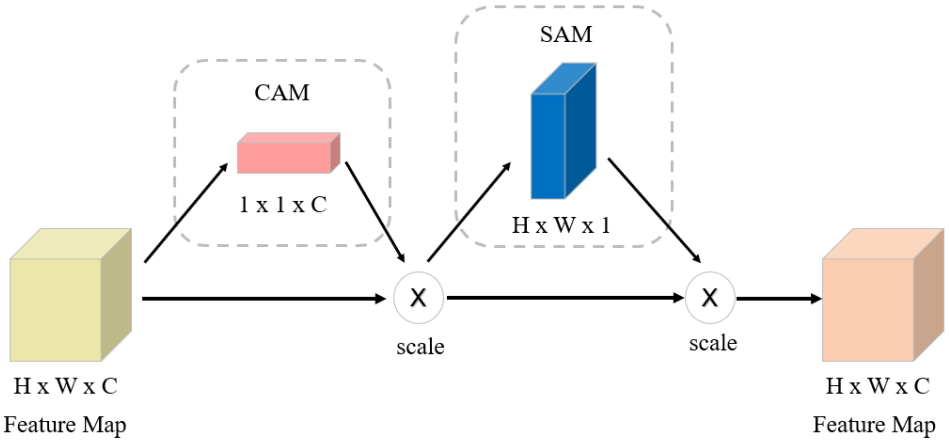


图 3-14 C_S_MAM 模块设计

这样设计的原因是本文设计了基于 FPN 的多尺度预测网络，使用 P2、P3、P4、P5 四个不同尺度的特征图对大小不同的目标分别进行预测，而 P2、P3 特征图相对来说属于低层特征，拥有更多的位置信息，语义信息较为缺失，P3、P4 属于高层特征，拥有更多的语义信息，位置信息较为缺失，因此对低层特征使用先进行通道注意力增强语义信息再进行空间注意力的方式，对高层特征使用先进行空间注意力增强位置信息再进行通道注意力的方式。如图 3-15 为加入 MAM 注意力模块后的网络结构：

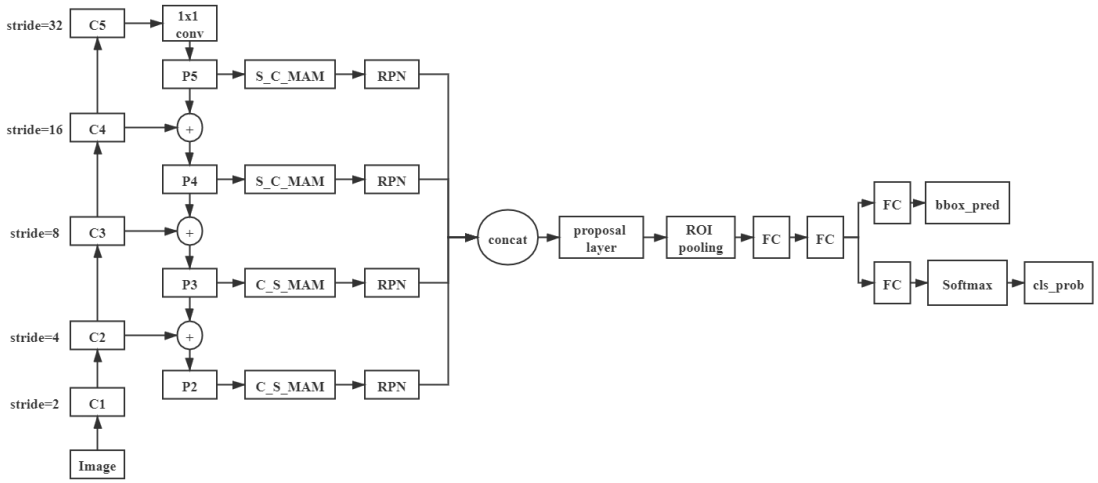


图 3-15 加入 MAM 模块后的网络结构

3.2.4 损失函数设计

Faster R-CNN 的回归框损失函数为 SmoothL1，论文已在 2.2.1 节中对其函数本身进行了详细介绍，在实际使用中的损失公式如公式(3-8)所示：

$$L_{loc}(t^{\mu}, t^{v}) = \sum_{i\in\{x,y,w,h\}} smoothL_1\left(t_i^{\mu} - t_i^{v}\right) \tag{3-8}$$

其中$t^{v} = (t_x^{v}, t_y^{v}, t_w^{v}, t_h^{v})$表示 Ground Truth 框和匹配的 Anchor 计算出的偏移量，$t^{\mu} = (t_x^{\mu}, t_y^{\mu}, t_w^{\mu}, t_h^{\mu})$表示模型对该 Anchor 预测的偏移量，即分别求 4 个偏移量的的 SmoothL1 损失，然后相加作为框的回归损失。虽然 SmoothL1 已在 L1 和 L2 函数的基础上进行了改进，综合了两者优点，使得梯度在输入太大和太小的时候都保持相对稳定，但分析实际使用的损失公式，发现使用 SmoothL1 函数是对框的四个元素分开独立求出损失，但实际上四个元素整合起来才是对框的描述，它们之间是有内部联系的，并且衡量回归框检出效果的评价指标是 IOU，多个检测框可能会有相同的 SmoothL1 损失，但是 IOU 差异很大。所以，为了解决损失函数和评价指标不等价的问题，Jiahui Yu 等人提出了 IOU 损失[37]，将 IOU 评价指标直接作为损失进行梯度下降，具体公式如下公式（3-9）所示：

$$L_{IOU} = 1 - IOU(B, B^{gt}) = 1 - \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \tag{3-9}$$

其中B^{gt}为真实框，B为检测框，但是 IOU 损失存在两个问题，首先，当两个框没有重叠部分时，IOU 的取值为 0，此时梯度也为 0，无法给出训练的优化方向。其次，IOU 无法分辨不同的重叠方式，如图 3-16 中三种不同方向的重叠方式，其 IOU 值都是相等的。

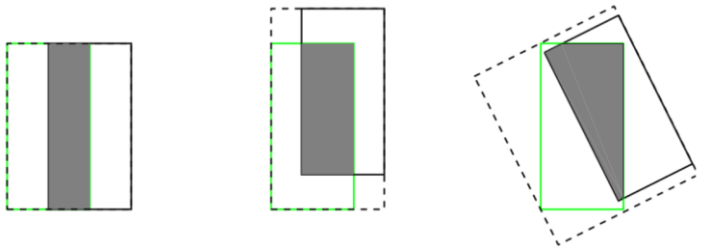


图 3-16 相同 IOU 值的不同重叠方式

为了解决 IOU 损失存在的问题，Hamid Rezatofighi 等人提出了 GIOU（Generalized IoU）损失[38]，在 IOU 损失的基础上添加一个惩罚项，如公式（3-10）（3-11）所示：

$$GIOU = IOU(B, B^{gt}) - \frac{|C - B \cup B^{gt}|}{|C|} \tag{3-10}$$

$$L_{GIOU} = 1 - GIOU(B, B^{gt}) = 1 - IOU(B, B^{gt}) + \frac{|C - B \cup B^{gt}|}{|C|} \tag{3-11}$$

其中 C 为B和B^{gt}的最小外接矩形，其具体关系如图 3-17 所示，惩罚项实际上就是描述了除了两个框以外的部分（图中黄色部分）在最小外接矩形中的占比，可以直观看出，两个框的非重叠部分越小，黄色部分越小。

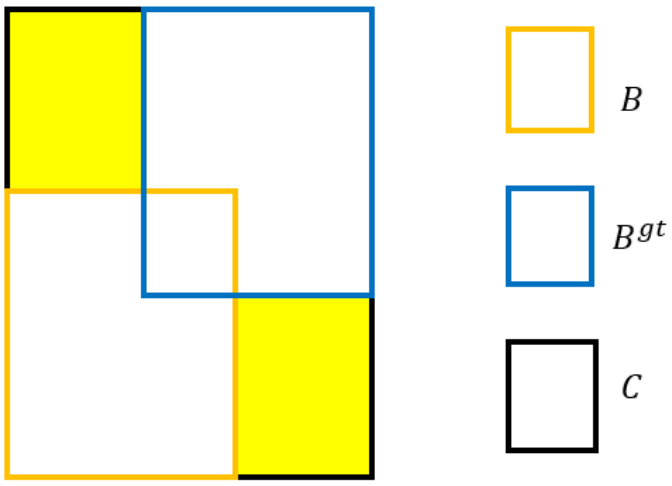


图 3-17 惩罚项组件关系

GIOU 添加的惩罚项考虑到了 IOU 没有考虑到的非重叠区域，能够反映出两个框的重叠方式，且当两个框不重叠时，损失梯度不再是 0，仍然能够引导神经网络的优化方向，因此本文使用 GIOU 损失作为检测的回归框损失，而分类损失仍然使用 Softmax 损失。

3.3 实验与分析

3.3.1 实验环境

实验的软硬件环境如表 3-6 所示：

表 3-6 实验软硬件环境

服务器 IP	192.168.1.89
操作系统	Ubuntu 16.04.5 LTS
CPU	Intel(R) Xeon(R) CPU E5-2683 v3 @ 2.00GHz 56 核心
内存大小	64G
网卡速度	1000Mb/s
硬盘	8.2TB
GPU	Nvidia GeForce GTX 1080 Ti
显卡驱动版本号	390.87
显卡个数	4
单张显卡容量	12GB
CUDA	10.0
cuDNN	7
Python 版本	3.6
Tensorflow 版本	1.13

3.3.2 模型评价指标

对于模型最终性能如何，我们应该有客观的衡量标准，也就是模型的评价指标。首先要对是否成功检测到目标进行定义，这里需要用到第二章中已经提及的非常重要的衡量指标——IOU，其计算公式如公式（3-12）所示:

$$IOU(B, B^{gt}) = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \qquad (3-12)$$

B和B^{gt}分别代表目标的预测框和真实框，|.|表示获取几何面积，IOU 值描述了两个框交集的部分占并集的比例，从一定程度上刻画了两个框的重叠度，一般认为 IOU 值大于 0.5 就是成功检测到了目标。

精确度（precision）和召回率（recall）是目标检测任务中常用到的评价指标。在介绍这两个指标的计算方法之前，首先要介绍一些基本概念：TP（True Positive）为模型成功检测到的正确的目标数量，FP（False Positive）为模型检测为目标但实际不是目标的数量，又称为误报，FN（False Negative）为模型没有检出但实际为目标的数量，又称为漏报。精确度和召回率的计算公式如公式（3-13）（3-14）所示：

$$Precision = \frac{TP}{TP + FP} \quad (3-13)$$

$$Recall = \frac{TP}{TP + FN} \quad (3-14)$$

精确度描述了模型检出为正样本的目标中真实正样本的比例，也就是查准率，召回率描述了所有真实正样本中被模型成功检出的比例，也就是查全率，从两个不同的角度评价模型的性能，一般来说精确度与召回率成反比，精确度越高召回率越低。因此为了从综合的角度描述模型性能，提出了 AP（Average Precision）的概念，AP 使用一句话概括就是召回率在[0,1]范围内的平均精确度，求解该值就是求解横坐标为召回率，纵坐标为精确度的 P-R 曲线下的面积，通常情况下都采用插值求解的方式求出，具体的计算方式分为 VOC07 标准和 VOC10 标准，本文采用的是 VOC10 标准。而 mAP（mean Average Precision）就是多类别情况下各类别 AP 的平均值，本文实验报告的指标都默认是 mAP 指标。

3.3.3 网络训练技巧

（1）数据增强

遥感目标存在着目标方向角度多的问题，而一般来说要学习到目标的方向角度，需要如 RBOX[39]这样设计专门的网络结构，并且数据集本身要带有旋转角度的标签（如 DOTA 数据集[40]）。一方面，由于我们的数据标签中没有带旋转角度的标签，并且神经网络本身就具有一定的旋转不变性，所以我们并没有对目标方向进行针对性优化，另一方面，从神经网络本质上来说，只要各个方向的数据量足够，网络就能检测出各个方向的目标，所以我们使用数据增强操作，对原有的数据进行 90 度、180 度、270 度旋转，就可以在一定程度上增加各种方向的样本，提升网络的性能。

（2）预处理和批归一化层

通常我们在将数据输入网络之前都会进行预处理操作，先计算出数据集的均值和标准差，然后对每个数据进行减去均值除以标准差的操作，使数据变为均值为 0，方差为 1 的分布，这样的操作又称为归一化操作，归一化可以使训练数据拥有相同的分布，加快模型收敛，如果每次数据分布不同，模型更新过程中会付出一定代价用来调整这种分布的差异。而批归一化层（Batch Normalization）的出现[41]，就是继承了预处理归一化的思想，可以在网络的任意一层对特征进行归一化处理。要知道虽然我们对输入数据进行了归一化处理，但是特征经过神经网络的矩阵权重运算和非线性激活后，分布极有可能发生改变，所以在网络中添加 BN 层就可以

重新调整特征的分布，解决梯度在激活函数饱和区产生梯度消失的问题，并且一定程度上了提高网络的泛化性能。

（3）添加正则项

添加正则项是一种简单实用的防止网络过拟合的方式，如最常用的L2正则化，通过将网络参数的平方项之和的均值乘上一个正则化系数添加到网络的损失函数中，这样在梯度反向传播的时候会约束模型参数的数值向 0 靠近，否则模型参数的数值越大，正则化项的值就会很高，违背了网络朝着损失函数梯度下降的方向收敛的原则。一般来说，我们认为参数越多，模型复杂度越高，而模型参数的值趋近于 0，会在一定程度上降低网络的复杂度，增加了网络的泛化性能，防止过拟合。

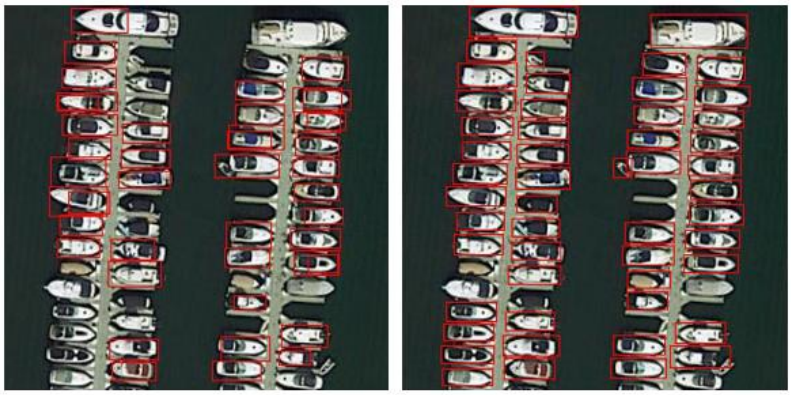
3.3.4 实验与分析

3.3.4.1 定性分析

首先本文对模型优化的效果进行定性分析，我们在数据集中挑选了部分具有代表性的样本，分别使用 Faster R-CNN 和改进后的网络（本文中命名为 Remote Faster R-CNN，后文简称 RF R-CNN）进行预测，轮船目标使用红色检测框，飞机目标使用蓝色检测框，结果如图 3-18 所示：

Faster R-CNN　　RF R-CNN

图（a）



Faster R-CNN　　RF R-CNN

图（b）

图 3-18 网络优化效果对比图

观察两个模型对相同样本的预测效果，图（a）Faster R-CNN 对于第一行的图像占比极小的小船没有检测出来，对第二行的图像占比极大的飞机也没有检测出来，而 RF R-CNN 都成功检出，通过对比我们可以直观看出改进后的 RF R-CNN 网络模型对于尺度差异大的特大和特小目标的检测效果优于原来的 Faster R-CNN。另外，从图（b）中我们可以看出对于目标小而密集的场景，Faster R-CNN 存在一定的漏检和错检情况，而 RF R-CNN 虽然也有一些错误，但是整体效果优于 Faster R-CNN。

3.3.4.2 定量分析

定性分析直观地展示了 RF R-CNN 的优化效果，但我们仍然需要从客观的角度衡量模型提升效果，我们使用相同的设置对两个网络在测试集上进行预测，然后将预测结果按照 3.4.2 节中的模型评价指标进行了计算（IOU 阈值为 0.5），结果如表 3-7 所示：

表 3-7 模型评价指标对比

	plane			ship			
网络模型	precision	recall	AP	precision	recall	AP	mAP
Faster R-CNN	93.56%	89.42%	91.67%	90.63%	83.75%	85.47%	88.57%
RF R-CNN	95.34%	92.85%	94.05%	92.28%	89.32%	90.77%	92.41%

从表中结果可以看出，不管是飞机类还是轮船类，其精度和召回率都得到了提升。对于两个网络来说，精度指标都高于召回率指标，说明对于本数据集，目标查

漏的情况多于目标查错的情况，即检测出的结果大部分都是正确的，但是存在一些目标没有检测出来。观察综合指标 AP，飞机类的 AP 提高了 2.38%，轮船类的 AP 提高了 5.3%，轮船类的 AP 提升幅度更大，观察轮船类的指标提升主要在于召回率，从 83.75%提升到 89.32%，提高了 5.57%，分析原因应该是轮船相比于飞机数据，存在更多的小目标和密集场景，所以经过了针对小目标和密集数据特点的优化之后，轮船的召回率得到明显提高。总体的 mAP 作为网络对本数据集的综合检测性能指标，从 88.57%提高到 92.41%，提高了 3.84%。

另外，为了验证针对遥感目标的所提出的改进策略对网络模型的单独提升效果，我们还对其进行了单个策略的优化效果验证实验，具体方法就是在 Base 网络的基础上，对优化策略进行逐步添加，观察每种优化策略给网络带来的提升，结果如表 3-8 所示：

表 3-8 单个优化策略的效果对比

优化策略	1	2	3	4	5
Base	√	√	√	√	√
+多尺度结构		√	√	√	√
+Anchor 聚类			√	√	√
+MAM 模块				√	√
+损失函数					√
mAP	88.57%	89.83%	90.75%	91.79%	92.41%

观察每个策略的加入为 mAP 带来的提升，加入多尺度网络结构带来了 1.26%的提升，将 Anchor 改为针对数据集本身聚类出来的 Anchor 带来了 0.92%的提升，加入本文设计的空间域和通道域注意力模块带来了 1.04%的提升，将损失函数改为 GIOU 损失带来了 0.62%的提升。可以看出四种优化策略都为网络的提升做出了贡献，但是提升的幅度有所不同，接下来我们将对其进行展开分析。

首先，从提升的数值上来看加入多尺度网络结构的效果最好，这是因为 Faster R-CNN 中只使用了最后一层特征图进行检测，而很多小目标经过池化层的四次下采样压缩之后，几乎已经没有多少保留的信息了，所以很难被检测出来，而采用多尺度的网络结构，在下采样倍数低的特征图上预测小目标，在下采样倍数高的特征图上预测大目标，有效提高了网络对小目标的检出率，所以带来了较大的提高。表 3-8 中两类目标的召回率特别是轮船召回率的大幅提高也侧面佐证了这一观点，因为召回率的本质是图像中所有目标被模型成功检出的目标所占的比例，又称为查全率。在多尺度结构有效提高网络对小目标的检出率之后，目标的漏检情况得到改

善，召回率自然得到提升。

除了多尺度结构外，加入注意力机制也带来了较大的提升，这说明本文设计的MAM注意力模块确实在一定程度上增强了特征图中比较重要的特征，特别地，我们比较了对所有层级的特征图使用相同的MAM模块（都使用C_S_MAM模块或都使用S_C_MAM模块）和分别使用不同的模块（P2、P3层级使用C_S_MAM模块，P4、P5层级使用S_C_MAM模块）的效果，发现分别使用不同的模块效果更好，因为在较低的特征图中位置信息比较丰富，语义信息较为缺失，为了避免特征图乘上空间注意力权重后模糊了特征图的语义信息，所以先进行通道注意力增强语义信息再进行空间注意力，而较高的特征图相反，语义信息丰富，位置信息缺失，所以先进行空间注意力增强位置信息再进行通道注意力。

同样地，针对数据集的Anchor聚类和损失函数更改也都有效地提高了网络的性能，提升幅度没有前两个模块高可能是因为网络本身的检测任务就包含了对Anchor进行位置的调整，所以Anchor的起始状态对于性能的影响不是关键性的因素；而损失函数的更改带来的提升说明GIOU损失确实比$SmoothL_1$损失效果更好，但因为$SmoothL_1$损失本身对梯度下降的指导就有较好的效果，所以提升效果有限，可以尝试对分类损失和回归框损失进行进一步的改进。

3.4 本章小结

本章首先对遥感数据集的构建和分布进行了简单介绍，然后针对遥感目标的特点进行了分析。接下来，我们针对遥感目标提出了四个网络优化策略，分别是（1）借鉴FPN的思想，将网络结构修改为多尺度预测结构。（2）使用K-means算法针对数据集的真实框进行聚类，然后将聚类结果作为Anchor分配给不同层级的特征图。（3）基于通道注意力机制SE模块的结构，提出了空间注意力模块SAM和通道注意力模块CAM，并按顺序不同组合为C_S_MAM模块和S_C_MAM模块，分别应用在不同层级的特征图上。（4）对损失函数进行修改，将回归框损失从$SmoothL_1$改为GIOU损失。最后，将原始网络和优化后的网络进行实验比较和分析，并验证我们提出的四个优化策略确实为网络性能带来了不同程度的提升。

第四章 基于半监督的遥感目标检测算法研究

本章将对基于半监督学习的遥感目标检测算法进行研究。首先，对遥感数据集进行划分，分为少部分有标签数据和大部分无标签数据，然后使用上一章设计的RF R-CNN 网络作为检测模型，针对第二章中介绍的两种半监督学习方法引入三种改进方法：第一，本章对基于伪标签的简单自训练方法进行了实验，并通过对结果分析发现自训练方法在伪标签携带噪声过多的情况下效果并不好，于是提出了慢启动的自训练方法，分步挑选简单样本制作伪标签，实验证明带来一定的提升。第二，针对上一步提出的慢启动自训练方法发现的问题，在自训练方法的基础上引入了主动半监督学习框架，同时关注无标签样本中的难样本和简单样本，并提出一种基于委员会的不确定度采样策略，对目标的分类和定位不确定度进行刻画。第三，对一致性正则化方法中的 Mean Teacher 算法进行改进，提出了一种基于 Mean Teacher 的学生-教师半监督训练框架，并针对无标签样本设计对应的一致性正则损失加入整体损失中。本章进行的实验证明，只要方法设计得当，半监督学习算法可以通过对无标签样本的利用给遥感目标检测模型带来一定的提升。

4.1 数据集划分

第三章使用的数据集都是有标签数据，但是由于我们要对半监督学习方法进行实验，需要无标签样本，因此我们对数据集进行划分，分别取 10%、20%、30%作为有标签样本，剩下的作为无标签样本，表 4-1 为数据训练集划分情况：

表 4-1 训练集划分

比例	有标签样本	无标签样本
10%	745 张	6703 张
20%	1490 张	5958 张
30%	2235 张	5213 张

4.2 基于伪标签的自训练方法探究与改进

4.2.1 简单自训练方法

自训练方法的思想很简单，就是利用已有的标签数据训练出一个初始模型后，再利用初始模型通过某种方式对无标签样本产生伪标签，然后将伪标签样本加入

训练集中一同进行监督训练，该方法在分类任务上展现出了不错的效果。然而，检测任务相比于分类任务更加复杂，需要先找到目标候选框，然后再对框中的目标进行分类，所以在分类中获得一定效果的自训练方法不一定在检测中也能有效，本小节将对其进行实验探究。最基本的简单自训练方法的算法流程如表 4-2 所示：

表 4-2 简单自训练方法流程表

算法 4-1 简单自训练方法
Input：标注数据集 L，无标注数据集 U，测试集 V **Output**：模型权重 W 1. 超参初始化：总轮数 R，伪标签保留阈值 T 2. 使用 L 对模型进行训练获得初始权重 W 3. r=0 4. **while** r<R **do** 5. 根据 W 对 U 进行推断，得到预测结果 R 6. 根据阈值 T 对结果 R 进行筛选，只保留置信度超过 T 的框为伪标签集合 P 7. $L_r = P \cup L$ 8. 使用L_r对模型进行训练得到权重 W 9. r=r+1 10. **end** 11. **return** 检测器权重 W，在测试集 V 上测试结果

本文使用控制变量法对两个变量（自训练初始有标签数据和置信度阈值）进行实验，第一轮训练初始权重使用 40epoch，后面轮次皆为 20epoch，每轮初始学习率为 0.001，按指数衰减至 0.00001，batch size 设为 6，表 4-3 为实验结果：

表 4-3 简单自训练方法结果

初始状态	置信度阈值	R=0	R=1	R=2
10%有标签数据	0.5	40.63%	28.26%	15.29%
	0.7	40.63%	31.75%	23.56%
	0.9	40.63%	27.79%	12.84%
30%有标签数据	0.5	73.92%	69.83%	64.28%
	0.7	73.92%	71.64%	69.85%
	0.9	73.92%	68.59%	63.71%

R=0 时只使用了有标签数据进行监督训练，R=1 开始加入伪标签进行训练，从

实验结果可以看出随着训练轮数的增加，检测器 mAP 反而不断下降，说明直接在目标检测任务上进行简单的自训练来使用无标签样本是行不通的。观察表中结果可以发现，在置信度阈值保持不变时，10%有标签数据的 mAP 比 30%有标签数据下降更快，这是因为仅使用 10%有标签数据训练的初始精度比 30%数据更低，而初始精度越低，模型对样本的预测能力越差，伪标签中就会包含很多错误标签。另一方面，当保持训练初始标签数据不变时，可以发现 0.5 和 0.9 的置信度阈值的 mAP 下降速度比 0.7 置信度阈值更快，这是由于使用较高的置信度，会导致模型预测出的某些真实目标被过滤掉，没有加入伪标签集合中，造成下一轮训练时存在较多的假负例；使用较低的置信度，会导致某些模型预测错误的不是真实目标的框被加入伪标签集合，造成下一轮训练时存在较多的假正例，都会给训练带来噪声影响。因此，初始精度过低或置信度阈值设置得不好，将会导致伪标签中存在较大的噪声，特别是目标检测任务，比分类任务更加复杂，获得的标签除了类别信息外还有包围框的信息，可能会给训练带来更多的噪声，而在不准确的标签监督下进行训练，反而会阻碍检测器学习到正确的方向。

综上所述，简单自训练方法的在目标检测任务上的应用局限就是伪标签中含有的噪声会影响到模型的训练，而本文接下来将尝试对自训练方法进行改进。

4.2.2 慢启动的自训练方法

首先，我们发现简单自训练方法中添加伪标签的步骤是一次性将所有无标签样本都加入训练集中，但是样本的难易识别程度是有区别的，前期模型精度不高的情况下对于某些样本的识别是不准确的，这样的样本不应该加入标签集合中。因此我们想到将无标签样本分步添加至训练集中，先挑选出无标签样本中预测效果较好的部分样本，保证在前期初始精度不高的情况下添加的样本噪声很小，模型精度获得提升后再继续添加，所以本文设计了一种简单的挑选方式，即将图像检出的所有目标的平均置信度作为图像置信度，按图像置信度进行排序，挑选其中置信度较高的图像作为本轮加入标签集合的样本。使用平均置信度作为图像质量排序标准的出发点是因为同一张图像中的目标都是处于同样的拍摄分辨率下的，在多数情况下，同类目标（如飞机）在相同分辨率之下的大小差异不会很大，如果检出的所有框的置信度都很高，则倾向于认为本张图像的检出效果较好，漏检误检情况较少，可以加入训练。

其次，可以将有标签样本与无标签样本的损失分开计算，对无标签样本损失使用变化的权重，受 Dong-Hyun Lee[18]的启发，我们对 Faster R-CNN 的损失函数进行改进，添加无标签样本的损失，如公式（4-1）所示：

$$\mathrm{L}=\frac{1}{n}\sum_{i=1}^{n}L(y_i,f_i)+\alpha(t)\frac{1}{n'}\sum_{i=1}^{n'}L\left(y'_i,f'_i\right) \qquad (4-1)$$

其中 n 为 mini batch 中有标签样本的数量，n'为无标签样本的数量，$L(y_i,f_i)$即有标签样本损失，$L(y'_i,f'_i)$为无标签样本损失函数，包含类别损失与包围框损失，函数定义与公式（2-4）相同。$\alpha(t)$为权重系数，随着时间 t 的改变而改变，其定义如公式（4-2）所示：

$$\alpha(t)=\begin{cases}\alpha_s & t<T_1\\ \dfrac{t-T_1}{T_2-T_1}\alpha_f & T_1\le t<T_2\\ \alpha_f & T_2\le t\end{cases} \qquad (4-2)$$

这样改造的目的是想要前期学习不稳定的时候尽量加入少的噪声，获得一定的精度后再继续迭代，因此将本算法称为慢启动自训练算法。

4.2.3 实验与分析

对简单自训练方法进行改造后，我们进行实验验证，设置算法超参数总轮数 R 为 3，标签更新百分比为 10%，即每轮增加 10%的样本（745 张）至标签集合中，训练参数设置为第一轮训练初始权重使用 40epoch，后面轮次皆为 20epoch，每轮初始学习率为 0.001，按指数衰减至 0.00001，batch size 设为 6，使用 Adam 优化器进行反向梯度传播。损失函数$\alpha(t)$中α_s=0.5，α_f=1，T_1=8epoch，T_2=15epoch，实验结果如表 4-4 所示：

表 4-4 慢启动自训练方法结果

初始标签	0%伪标签	10%伪标签	20%伪标签	30%伪标签
10%有标签	40.63%	42.45%	43.57%	44.06%
20%有标签	62.87%	64.21%	64.93%	65.74%
30%有标签	73.92%	74.38%	74.89%	75.34%

可以看出相比于简单的自训练方法，慢启动自训练方法的学习更加稳定，分步将伪标签添加至标签集合中，获得了一定的提升，但是提升非常有限，观察表格数据，在初始有标签样本只有 10%的情况下，添加 30%的伪标签为模型的 mAP 带来了 3.43%的提高，而初始标签数据为 30%的情况下，添加 30%的伪标签只为模型的 mAP 带来了 1.42%的提高，初始精度越高的模型提升越少。本文认为这是因为在初始标签数据很少的情况下，模型对特征的学习远远没有达到饱和，因此添加的

带有少量噪声的伪标签数据可以帮助模型进一步对目标特征进行学习，而初始标签数据量较大，初始精度较高的情况下提升效果不明显，原因是我们只挑选了最容易被识别的样本加入数据集，这部分特征已经被精度较高的模型学习到了，所以对模型的性能提升十分有限。综上所述，只挑选简单样本加入训练是不够的，不能让模型学习到数据的整体分布，因此下一小节我们将同时关注样本中较难识别的样本和简单的样本，使用主动半监督学习框架进行同时训练。

4.3 与主动学习结合的自训练方法

通过对模型预测的结果进行分析，很容易发现不正确的标签通常都是一些较难分辨的目标，也就是我们所说的“难样本”。“难样本”是模型训练中难啃的硬骨头，也是影响模型精度的关键，但是通常“难样本”在数据集中的比例都不高，那么能否采取一种方法较为准确地区分出无标签样本中的难样本和简单样本，简单样本由训练好的检测器进行标记，难样本由人工进行标记，使得用于自训练的伪标签中的噪声大大下降呢？这样的方法涉及到主动学习（Active Learning）的思想，所谓主动学习就是在学习过程中，使用某种采样算法从无标签样本中主动地采样出部分“关键”的样本，请求人工进行标注。其目标是尽可能挑选到蕴含信息量大的样本，使用尽可能少的标注成本来取得好的学习性能。主动学习与半监督学习相结合，就成为了半监督学习的一条分支——主动半监督学习。

本文接下来首先将会对主动半监督学习的思想和框架进行介绍，然后提出一种针对目标检测任务的基于委员会的不确定度采样算法对无标签样本进行采样，并进行主动半监督目标检测实验。

4.3.1 主动半监督学习框架

主动半监督学习是将主动学习和半监督学习相结合的一个研究分支，在第二章中已经对半监督学习理论进行了详细介绍，半监督学习的目标是利用无标签样本中的数据分布信息来探明数据中的高密度区域和低密度区域，帮助修改学习器的决策边界尽可能穿过数据稀疏的区域，提升仅使用少量有标签样本时的学习性能。主动学习的目标与半监督学习的目标总体方向一致，都是利用无标签样本中含有的信息，但主动学习认为样本含有的信息量存在差异，决策边界附近的样本相比于簇中心的样本对于学习器找到准确的分类边界有着更大的帮助，主动学习就是采取某种策略采样出信息量更大的样本，人工进行标注然后训练。图 4-1 为主动学习的一个例子：

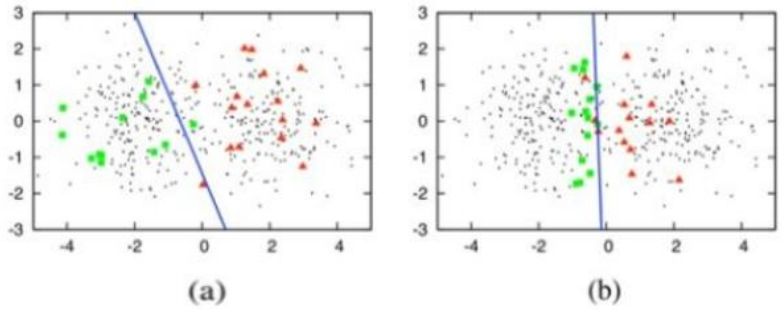


图 4-1 主动学习示例

图中绿色方形与红色三角形分别代表有标签的正负样本集合，黑色圆点代表无标签样本集合，蓝色实线为学习器学到的决策边界，图（a）为使用随机采样策略获得的结果，图（b）为使用某种主动学习采样策略获得的结果，可以看出使用同样的标注样本数量，图（b）获得了更好的分类性能。

图 4-2 为主动学习的流程图，和半监督学习自训练流程极其相似，都是利用已有标签数据训练出初始模型后再利用无标签数据扩充数据池进行迭代训练，唯一不同的地方在于主动学习选择无标签样本时会采取某种查询策略（select queries）挑选出样本中信息量（Informativeness）更大的样本，然后提交给专家进行人工标注。

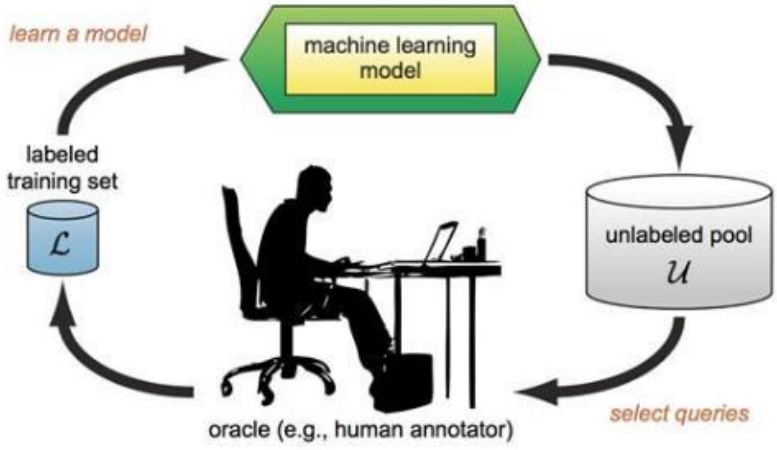


图 4-2 主动学习流程示意图

因此不需花费多少力气就可以将半监督学习与主动学习相结合，如图 4-3 为主动半监督学习的框架图，首先，使用某种采样策略从无标签数据池中采样出高不确定度（即模型识别困难）的样本和低不确定度（即模型识别容易）的样本，然后将高不确定度的样本交由人工进行标注，低不确定度的样本就直接使用模型进行打

标签，最终将生成的标签和对应样本合并到训练集中，对检测模型进行训练，得到更高精度的模型后再重复上述的采样-标记-训练过程，不断迭代直至达到要求的精度或无标签样本用完。

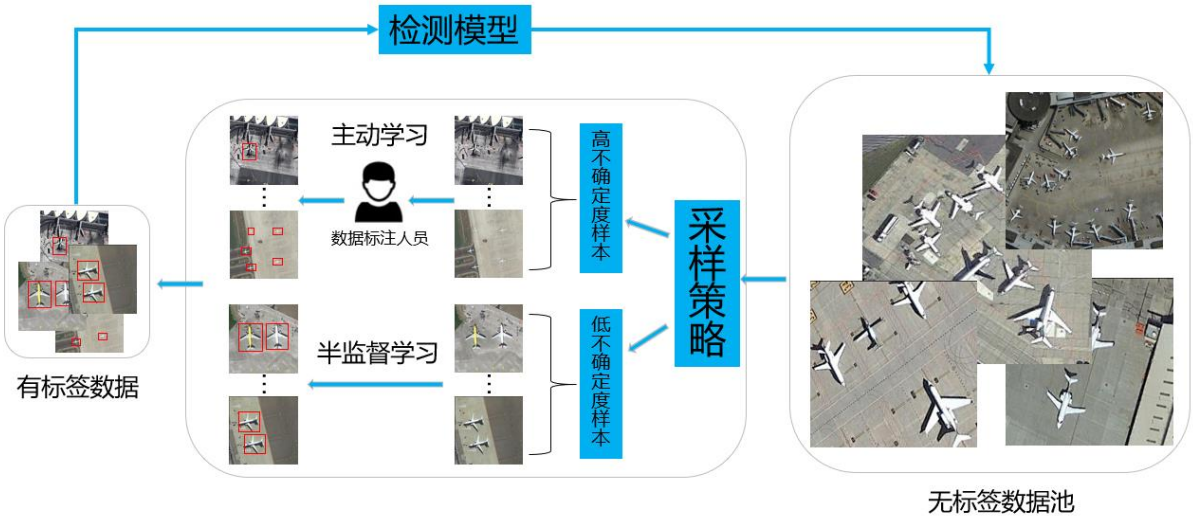


图 4-3 主动半监督学习框架图

4.3.2 主动学习采样策略介绍

如何定义样本的信息量和设计图 4-3 中的采样策略成为主动半监督学习问题的关键。主动学习领域常用的采样策略有不确定度采样的查询（Uncertainty Sampling），不确定度采样是使用最为广泛的查询策略，即使用某种方式来刻画样本所包含的信息量，衡量样本的不确定度，将“最容易混淆”的样本返回；基于委员会的查询（Query-By-Committee），即使用模型投票的方式，不同的模型组成委员会对样本进行预测，投票分歧最大的样本为“争议样本”，一般使用投票熵（Vote Entropy）公式来衡量；基于模型变化期望的查询（Expected Model Change），即认为对模型改变最大的样本就是高价值的样本，一般使用梯度的改变来衡量；基于误差减少的查询（Expected Error Reduction），即选择使得模型损失下降最多的样本、基于方差减少的查询（Variance Reduction），即选择那些使模型方差减少最多的样本、基于密度权重的查询（Density-Weighted Methods），即认为在较稠密区域的难区分样本比游离在外的难区分样本（可能是异常点）更具有价值。这些查询策略的形式有所不同，但是本质都是一样，就是在无标签的情况下找出对模型学习性能最有价值的样本。

本文选择了主动学习中最经典也是最常使用的采样策略——不确定度采样。不确定度采样中对于样本不确定度的刻画有三种经典策略，分别为最小置信度（Least Confident）、边缘采样（Margin Sampling）、信息熵（Entropy）。

最小置信度是指在多分类场景下，模型对每个分类都会进行概率预测，选择那些最大概率最小的样本进行标注。例如二分类场景下，对两个样本的类别预测概率为（0.9,0.1）和（0.6,0.4），两个样本都会按照最大置信度分为第一类，但是前一个样本第一类概率为 0.9，后一个样本概率为 0.6，则认为后一个样本更难被模型分辨，因此更具有被标注的价值。数学定义如公式（4-3）（4-4）所示：

$$x_{LC}^{*} = argmin_{x} P_{\theta}(\hat{y}|x) \tag{4-3}$$

$$\hat{y} = argmax_{y} P_{\theta}(y|x) \tag{4-4}$$

其中θ表示模型参数，$P_{\theta}(\hat{y}|\mathrm{x})$是输入 x 后模型预测概率最大的类对应的概率值。

边缘采样是指在多分类场景下，选择模型预测概率最大和第二大的概率差值最小的样本。例如三分类场景下，对两个样本的类别预测概率为（0.9,0.2,0.1）和（0.6,0.4,0.1），前一个样本的最大和第二大概率差值为 0.7，后一个样本的最大和第二大概率差值为 0.2，则认为后一个样本的不确定度更高，模型更难分辨。数学公式定义为：

$$x_{MS}^{*} = argmin_{x}\left(P_{\theta}(\hat{y}_1|x) - P_{\theta}(\hat{y}_2|x)\right) \tag{4-5}$$

其中θ表示模型参数，$P_{\theta}(\hat{y}_1|\mathrm{x})$和$P_{\theta}(\hat{y}_2|\mathrm{x})$表示输入 x 后模型预测概率最大的概率值和第二大概率值。对于本文实验所用的数据集来说，由于只有“飞机”和“船”两类目标，所以最小置信度采样和边缘采样在二分类的情况下效果是一致的。

信息熵的熵是物理学中的一个概念，最初是在热力学中用来描述能量退化的物质参数，后来随着统计物理、信息学的发展，熵的含义越来越广泛，但其本质就是描述一个系统“内在的混乱程度”的度量。熵越大表示系统的不确定性越大，上越小表示系统的不确定性越小，因此，可以认为熵最大的样本是不确定性最高的样本。熵的数学定义公式为：

$$x_{E}^{*} = argmax_{x} - \sum_{i=1}^{n} P_{\theta}(y_i|x) * lnP_{\theta}(y_i|x) \tag{4-6}$$

其中 n 为类别数量，θ表示模型参数，$P_{\theta}(y_i|\mathrm{x})$表示输入 x 后模型预测对应的第 i 类的概率值。

上述介绍的不确定度计算方法是针对分类任务的，即对于一张图像只输出一次分类概率预测，但是本文研究对象是目标检测任务，每张图像中可能有多个目标，每个目标都会输出分类概率，因此需要修改上述采样方式，在输出多目标分类概率的情况下对整张图像的不确定度进行刻画。并且上述方法在刻画图像的不确定度

时只考虑到了分类部分，没有考虑另一方面的定位部分，因此需要补充针对目标检测回归框定位的不确定度描述度量。

4.3.3 一种基于委员会的不确定度采样策略

针对上述提到的需要同时考虑分类和定位不确定度的问题，本文提出了一种基于委员会的不确定度采样策略，引入了委员会查询的思想，主要分为两个模块：目标委员会构建模块和图像不确定度计算模块，主要思想就是修改 NMS（Non-maximum suppression）算法，在模型对图像进行推理预测之后，在进行 NMS 操作之前，对模型检测出的候选目标进行目标提取，针对每一个目标构建一个委员会。构建委员会的目的是因为在没有标签监督的情况下，判断检测器是否正确定位目标的包围框是非常困难的，因此我们引入了委员会的思想，在两两成员之间计算 IOU 值。如果是简单易识别的目标，模型回归的包围框应该是高度一致的，如果是难以识别的目标，模型回归的包围框可能存在较大差异。因此在没有标签的情况下可以利用委员会成员间的不一致性刻画模型定位的不确定度。完成委员会构建后，在其内部对目标的分类和定位不确定度进行计算，最后聚合所有目标的不确定度来刻画图像样本不确定度。图 4-4 为基于委员会的不确定度采样策略计算流程图：

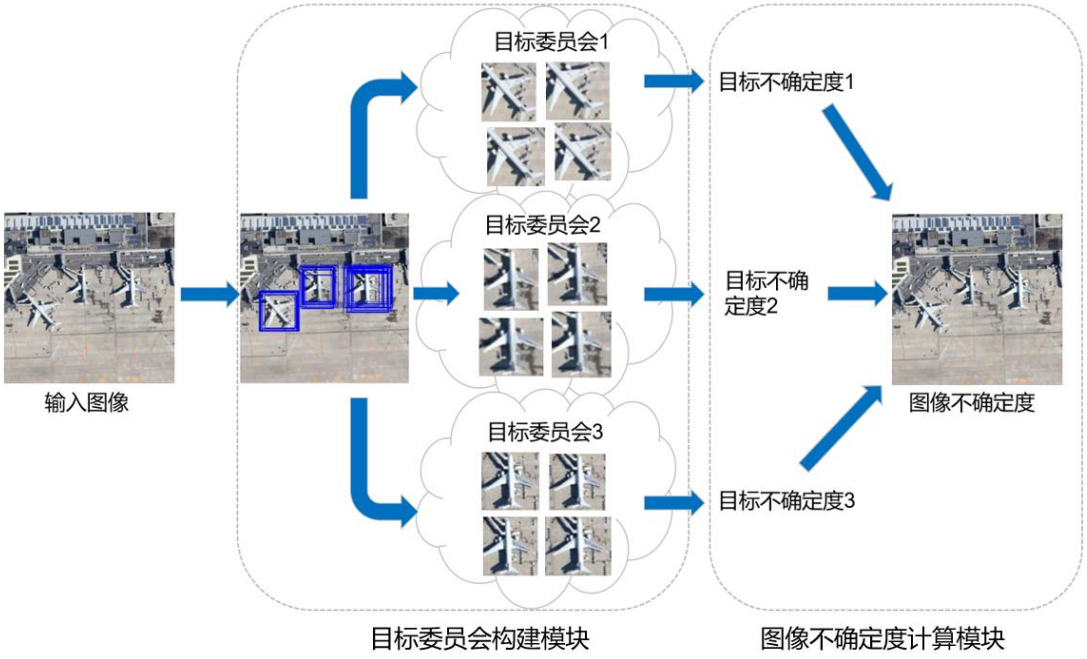


图 4-4 基于委员会的不确定度采样策略计算流程图

（1）目标委员会构建模块

对于任意一幅图像，其中含有的目标数量是未知的，要在目标级别对模型的不

确定度进行刻画，首先就需要确定模型输出的目标个数。在第二章目标检测模型的流程中已经介绍过，模型经过阈值筛选输出了所有认为是目标的候选框后还要送入 NMS 算法中对重叠度高的目标框进行移除，本文的目标委员会构建算法就是对 NMS 算法的思想进行反向思考，提出一种极大值扩散（Maximum Diffusion，MD）算法，将同一目标的包围框归纳到同一委员会中，方便下一步骤中进行目标级别的不确定度计算。图 4-5 为 NMS 算法之前的模型输出的候选框：

图 4-5 NMS 算法之前的模型输出的候选框示意图

一般来说目标的真实框经过网络卷积和池化操作后映射到特征图上仍然会占据一定面积，而特征图上每一个点都会预设 N 个不同尺寸的锚点（anchor），因此同一个目标在经过目标检测网络后会输出多个检测框，这些检测框在不同程度上都成功“命中”了目标。可以将每个检测框看做模型单独运行一次获得的结果，将分类概率最高的检测框作为局部极大值，与 NMS 思想相反，MD 算法将计算与局部极大值最接近的框，并将相邻检测框加入一个集合中（本文称为目标委员会），算法流程如表 4-5 所示：

表 4-5 基于极大值扩散的目标委员会构建算法

算法 4-2 基于极大值扩散的目标委员会构建算法
Input：B=$\{b_1, b_2, b_3 \dots b_N\}$← 检测框集合 S=$\{s_1, s_2, s_3 \dots s_N\}$← 置信度集合
Output：委员会集合 Commit
1. 超参初始化：IOU 阈值t_{iou}，分类阈值t_{cls}，委员会最小成员个数 K
2. Begin：
3. C ← {}
4. **while** B≠ ∅ **do**
5. m 为 S 中最高分类置信度对应的下标
6. **if** s_m<t_{cls} **then**
7. break;

```
8.          end
9.          将b_m加入委员会 C，B ← B-b_m，S ← S-s_i
10.         for b_i in B do
11.             if IOU(b_i, b_m)≥ t_iou and s_i ≥ t_cls then
12.                 将b_i加入委员会 C
13.                 B ← B-b_i , S ← S-s_i
14.             end
15.         end
16.         if num(C)>K then
17.             只保留委员会 C 中置信度排序前 K 个的候选框
18.             将委员会 C 加入委员会集合 Commit
19.         end
20.     end
21. end
22. return Commit
```

在 MD 算法中，每一轮都挑选置信度最高的框作为局部极大值加入到委员会中，剩余候选框加入该委员会的步骤如算法 10-15 行所述，需要同时满足两个条件（如公式（4-7）（4-8）所示）：

$$IOU(b_i, b_m) \geq t_{iou} \tag{4-7}$$

$$s_i \geq t_{cls} \tag{4-8}$$

其中t_{iou}为候选框与局部极大值框的最小 IOU 阈值，t_{iou}越大则要求两个框的重叠度越高，t_{cls}为候选框置信度的最小阈值，若小于该阈值说明模型对框中存在目标的置信度很低，则不应该加入到目标委员会中。

算法中还对一些特殊情况进行了处理，如算法 6-8 行：置信度最高的框的置信度已经低于t_{cls}，说明剩下的候选框置信度都低于t_{cls}，则没有必要再继续算法，退出循环。还有算法 16-19 行：如果加入委员会的候选框个数低于阈值 K，很有可能是模型特征识别出错出现的噪声框，因此低于 K 个候选框的委员会将被舍弃，高于 K 个候选框的委员会也只采样置信度最高的 K 个框，一方面是出于后续计算量的考虑，一方面是为了避免候选框中较差结果的影响。

（2）图像不确定度计算模块

构建好目标委员会后，就可以计算单个目标的分类和定位不确定度，由于委员会中的候选框都是针对同一目标的，不存在多分类概率的问题，不太适合上节介绍

的最小置信度和边缘采样方法，因此本模块中分类部分的不确定度由信息熵方法进行计算，定位部分的不确定度由委员会成员间的 IOU 进行描述。假设委员会所有成员两两组合可以构建 N 对组合，每对组合的不确定度计算公式如公式（4-9）（4-10）（4-11）所示：

$$d_{i,j} = d_{i,j}^{cls} + d_{i,j}^{loc} \tag{4-9}$$

$$d_{i,j}^{cls} = -\frac{1}{2}\left(p_i * lnp_i + p_j * lnp_j\right) \tag{4-10}$$

$$d_{i,j}^{loc} = 1 - IOU\left(b_i, b_j\right) \tag{4-11}$$

其中$d_{i,j}$表示委员会中第 i 个候选框与第 j 个候选框间的不确定度，$d_{i,j}^{cls}$表示分类的不确定度，使用信息熵公式计算，取两个候选框信息熵平均值，$d_{i,j}^{cls}$越大，模型对该目标的分类不确定度越大。$d_{i,j}^{loc}$表示定位的不确定度，通过计算两个候选框 IOU 值得到，重叠度越高，IOU 越大，$d_{i,j}^{loc}$越小，表示模型对该目标的定位不确定度越小，反之，$d_{i,j}^{loc}$越大，模型对该目标的定位不确定度越大。

所有组合的不确定度计算完毕后，对其求平均值作为该目标最终的不确定度，数学表达式如公式（4-12）所示，N 为组合数量，K 为委员会成员个数：

$$U^{obj} = \frac{1}{N}\sum_{i,j\in[1,\ldots,K],i\neq j} d_{i,j} \tag{4-12}$$

如图 4-6 为获得目标委员会后的目标不确定度计算流程图：

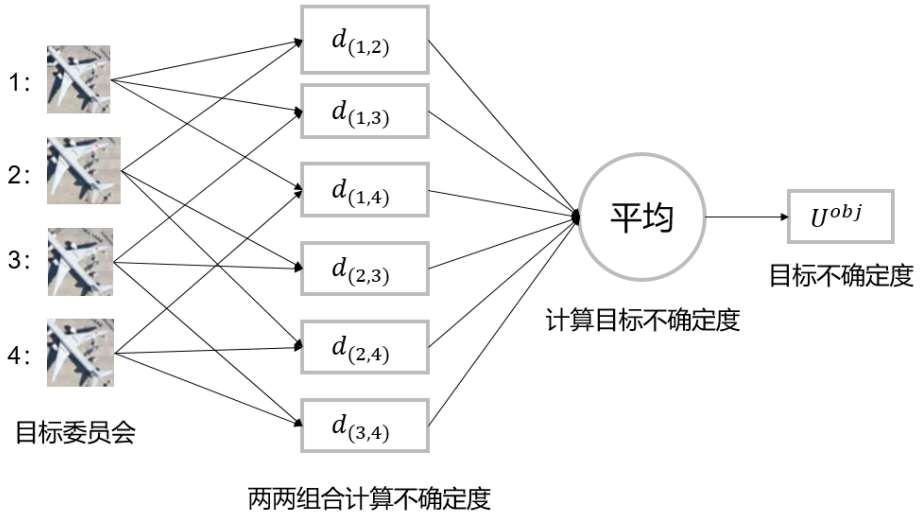


图 4-6 基于委员会的目标不确定度计算流程图

以上流程就是单个目标的不确定度计算过程，很多情况下图像中的目标都不止一个，因此还要聚合所有目标不确定度作为整张图像的不确定度。对图像的不确定度定义为公式（4-13）：

$$U^{img} = \sum_{k=1}^{m} U_k^{obj} \tag{4-13}$$

其中 m 为通过 MD 算法构建的目标委员会个数，U_k^{obj}是对应的第 k 个目标委员会计算得出的目标不确定度，对所有目标求和得到图像的不确定度。

上述两个模块组合即是本文提出的基于目标委员会的采样策略，其本质思想就是在不知道图像中目标个数和具体位置的情况下，借助模型预测出的候选目标构建目标委员会，利用平均投票的思想对其分类和定位不确定度进行计算，最后聚合为整张图像的不确定度。在主动半监督学习训练中，将使用该采样策略对无标签样本集合进行采样，高不确定度样本由于模型预测效果差，自动标注会对伪标签引入很高的噪声，将交由人工进行标注，低不确定度样本则直接由模型自动标注。图 4-7 为图像不确定度较高的样本示例。

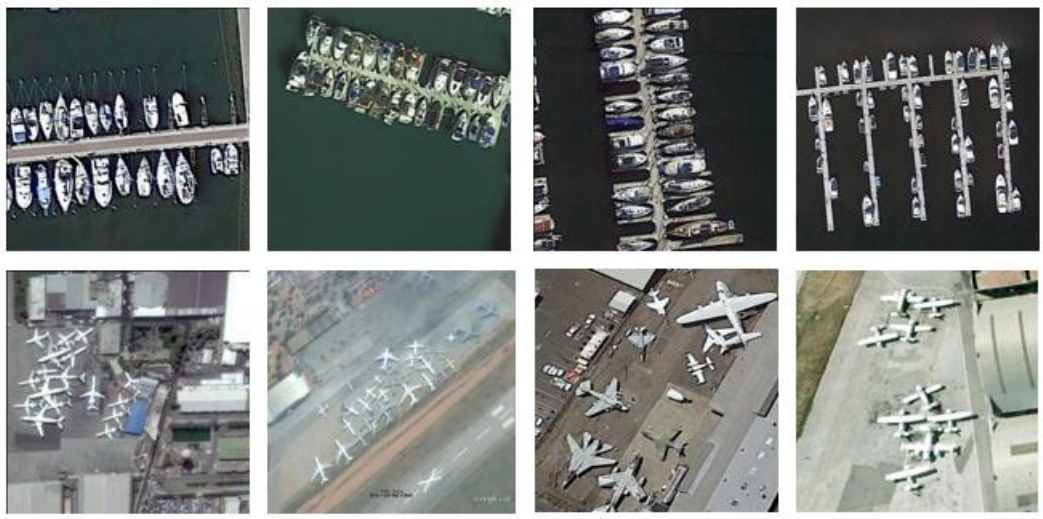


图 4-7 图像不确定度较高的样本示例

4.3.4 实验与分析

提出了针对目标检测任务的基于目标委员会的不确定度采样策略后，将其代入图 4-4 中的采样策略，进行主动半监督学习自训练实验。主动半监督学习自训练算法具体流程如表 4-6 所示：

表 4-6 主动半监督学习自训练方法

算法 4-3 主动半监督学习自训练方法
Input：标注数据集 L，无标注数据集 U，测试集 V
Output：模型权重 W
1. 超参数初始化：总轮数 R，采样百分比 K
2. 使用 L 对模型进行训练获得初始权重 W
3. r=1
4. **while** r<R **do**
5. 根据 W 对 U 进行预测，得到 NMS 之前的结果 R
6. 使用基于委员会的不确定度采样策略进行采样，得到 K%高不确定度样本D_H和 K%低不确定度样本D_L
7. D_{AL} ←人工对D_H进行标注
8. D_{SSL} ←检测器 W 对D_L进行标注
9. $L_r = L \cup D_{AL} \cup D_{SSL}$
10. 使用L_r对模型进行训练
11. 获得权重 W
12. r=r+1
13. **end**
14. **return** 检测器权重 W，在测试集 V 上测试结果

算法超参数 R 设为 3，采样百分比设置为 5%，即每轮增加 5%（372 张）的高不确定度样本标签和 5%（372 张）的低不确定度样本标签至标签集合，每轮总共增加 10%（746 张）的标签样本。第一轮训练初始权重使用 40epoch，后面每轮训练 20epoch，初始学习率为 0.001，按指数衰减至 0.00001，batch size 设为 6，使用 adam 优化器进行反向梯度传播。对于损失函数的选择，一方面，本算法中高不确定度样本由人工进行标记（在实验中直接使用真实标签），低不确定度样本由检测器自动标记，因此增加的标签中的噪声只可能由低不确定度样本引入，但检测模型对低不确定度样本预测效果优良，所以可认为增加的标签中噪声是较少的，可使用正常的监督学习损失函数。另一方面，主动学习的思想就是挑选出样本中信息量大的样本，更好地帮助学习器尽快找到准确的决策边界，所以对于新增加的标签不宜再使用“慢启动”的方法，否则就与主动学习的思想背道而驰了。基于以上两个方面的考虑，本算法的损失函数仍然使用模型原本的损失函数，将原有标签样本与新增标签样本混合进行无差别的监督学习训练。本实验总共进行三组实验，分别使用 10%（745 张）、20%（1490 张）、30%（2235 张）的有标签数据作为初始标签样

本，用于对比不同初始精度情况下本方法的效果，实验结果如表 4-7 所示：

表 4-7 主动半监督自训练方法结果

初始状态	0%标签扩增	10%标签扩增	20%标签扩增	30%标签扩增
10%有标签	40.63%	60.18%	71.94%	80.56%
20%有标签	62.87%	72.33%	81.07%	86.43%
30%有标签	73.92%	80.14%	85.22%	88.16%

从表中可以看出，相比于上一小节的简单自训练方法，本方法对模型初始精度依赖性很小，不论初始精度较低还是初始精度较高，本方法都带来了较大的精度提升，并且对初始精度越低的模型提升幅度越大，在第一轮增加 10%标签样本时，初始状态为 10%有标签数据的模型精度提升 19.55%（40.63%→60.18%），初始状态为 30%有标签数据的模型精度提升 6.22%（73.92%→80.14%），这样的差别是合理的，因为神经网络算法的本质是数据拟合算法，在前期训练数据很少的情况下，模型的能力没有达到饱和，甚至可能产生过拟合的现象，加入新数据后模型很快就能通过调整权重拟合新的数据分布，因此性能提升很快，到了后期数据变多，模型需要花费更多的代价去精调权重来拟合所有的数据，模型的“容量”逐渐达到饱和，所以性能提升较慢。

由于主动学习部分引入了人工标记，相当于引入了真实标签样本，提升模型性能是理所当然的，为了验证本方法的有效性并不只是因为人工标记保证了标签的正确性，也与主动学习使用的采样策略有关，我们采用控制变量的方法，在同样使用 10%标签数据作为初始模型训练集的情况下，将基于委员会的不确定度采样与随机采样进行对比，如表 4-8 所示：

表 4-8 基于委员会的不确定度采样与随机采样对比

采样算法	0%标签扩增	10%标签扩增	20%标签扩增	30%标签扩增
基于委员会的不确定度采样	40.63%	60.18%	71.94%	80.56%
随机采样	40.63%	58.24%	68.35%	77.19%

对应的折线图如图 4-8 所示：

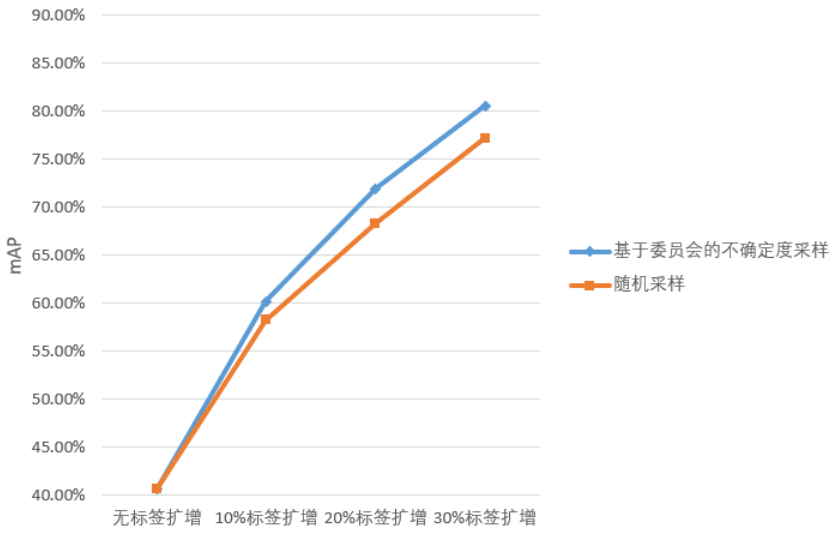


图 4-8 基于委员会的不确定度采样与随机采样对比折线图

可以看出使用基于委员会的不确定度采样策略的模型比随机采样的模型收敛更快，mAP 更高，分析其原因，是因为基于委员会的不确定度采样策略一方面有效地从无标签样本中采样到了不确定度高的样本，这样的样本往往包含更大的信息量，对于模型学习方向有更大的指导意义，另一方面采样到了不确定度低的样本，使用半监督学习自动标记，增加了训练集的数据量，并且相比于使用随机采样进行标记，带来的噪声更小。

4.4 一致性正则化方法

在进行实验前，我们先对自训练法和一致性正则法的区别进行分析判定。首先，先描述有标签数据的学习过程：1.获得数据标签。2.通过网络的预测值和标签值构造一个损失函数，描述模型预测性能的好坏。3.通过一系列最优化方式（如随机梯度下降）训练网络使损失函数达到最小。自训练方法就是针对上述学习过程的第一点，使用某种方式为无标签样本添加伪标签，然后进行监督学习训练。而半监督学习的另一种方法——一致性正则化方法，则是直接针对第二点，构造了不需要借鉴标签值的损失函数，直接利用无标签样本进行无监督训练。前文 3.2 节、3.3 节的实验都是基于自训练方法的基础上进行实验和改进的，本节则将对一致性正则化方法进行探究。

本文在 2.3.2 节中已经对一致性正则化的思想进行了介绍，其核心思想就是对于一个输入样本，即使受到微小的噪声干扰，模型对其输出都应该是一致的。在计算机视觉领域，对输入样本添加噪声干扰的操作被称为数据增强操作，因此本节首先将对实验所用的数据增强方案进行介绍，然后针对 2.3.2 节中介绍的 Mean

Teacher 方法进行改造，提出一种基于 Mean Teacher 的针对目标检测模型的学生-教师半监督训练框架，并设计出针对无标签样本的一致性正则损失加入整体损失函数，最后进行实验及分析。

4.4.1 数据增强方案

E. Cubuk[42]和 B. Zoph[43]等人已经证明了在图像分类和目标检测任务中，对已有的标签样本使用数据增强然后再进行训练，仍然会对模型精度有提升作用，这是因为数据增强扩增了训练的数据量，提高了模型的泛化性能，更重要的是为样本带来了噪声，增加了模型的鲁棒性。一致性正则化对无标签样本添加噪声的目的其实一样，也是为了增加模型的鲁棒性，学习到真正影响目标识别的关键特征。而针对无标签样本的数据增强方案与有标签样本有所不同，由于缺少真实标签，所以很多针对真实框的增强操作是无法使用的，接下来，本文将对所使用的数据增强方案进行介绍：

（1） 像素颜色变换类

色彩抖动：随机在一定幅度内调整图像的饱和度、亮度、对比度、锐度。

CoarseDropout：随机在图像上丢失面积大小可定、位置随机的矩形区域的信息，所有通道信息丢失产生黑色矩形块，部分通道的信息丢失产生彩色矩形块。

噪声扰动：在图像上随机添加黑色或白色噪声，例如高斯噪声、椒盐噪声。

模糊处理：在图像上随机进行模糊处理，如高斯模糊、平均模糊和中值模糊。

如图 4-9 为像素颜色变换的数据增强示例：

图 4-9 像素颜色变换的数据增强示例

（2） 空间几何变换类

随机翻转：包括水平翻转、垂直翻转、水平和垂直结合翻转。

随机旋转：对图像做各种角度的旋转操作，由于真实框由左上角右下角定位，某些角度旋转后不好计算框的位置，所以本文只进行 90 度、180 度、270 度旋转。

缩放变形：随机选取图像的一部分，然后将其放大到原图的尺寸。或者将原图缩小后，图像边界使用背景色填充到原图大小。

平移填充：将图像沿着横轴、纵轴或横轴纵轴同时移动，移动后出现的空缺使用背景色填充。

如图 4-10 为空间几何变换的数据增强示例：

图 4-10 空间几何变换的数据增强示例

4.4.2 基于 Mean Teacher 的学生-教师半监督训练框架

由于一致性正则化方法直接针对无标签样本构造损失函数，因此我们的训练应该分为两个部分，分别是有标签样本的监督训练和无标签样本的无监督训练，监督训练部分与第三章内容一致，就不再赘述，无标签样本的训练我们借鉴了 Mean Teacher 的结构，使用两个相同的网络，一个作为学生网络，一个作为教师网络，由于 Mean Teacher 是分类半监督算法，我们相应地做了适应目标检测任务的修改，具体的模型框架如图 4-11 所示：

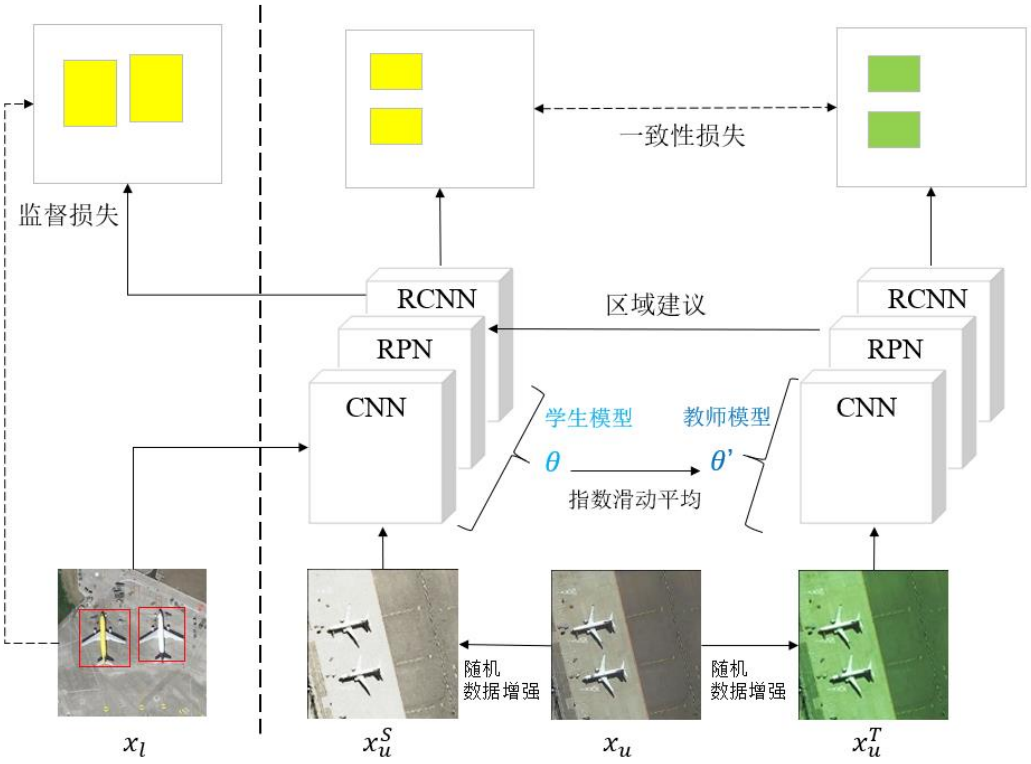


图 4-11 基于 Mean Teacher 的学生-教师半监督训练框架

图中虚线左边部分为有监督学习，x_l代表有标签样本，右边部分为无监督学习，x_u代表无标签样本，随机对x_u进行两次数据增强操作，获得x_u^S和x_u^T，分别输入学生网络和教师网络中。其中教师网络的权重通过对学生网络的权重进行指数滑动平均而来，θ_t为每一步学生网络的权重，θ_t'为每一步教师网络的权重，其更新公式如公式（4-14）所示，其中α为更新系数：

$$\theta_t' = \alpha\theta_{t-1}' + (1-\alpha)\theta_t \tag{4-14}$$

由于教师网络是由连续多个学生网络滑动平均而来，所以教师网络包含更多的特征信息，权重更新更加平稳，比学生网络更加健壮，不容易受到微小噪声的干扰。对同一个无标签样本进行两次随机添加噪声的操作（即上一小节的数据增强操作），然后分别输入学生网络和教师网络中，对学生网络在噪声干扰下的输出和教师网络在噪声干扰下的输出的一致性进行计算，并作为损失对学生网络权重进行更新，可以约束学生网络有和教师网络一致的输出，提升学生网络在噪声干扰下的预测稳定性，增加学生网络的鲁棒性和健壮性。

综上所述，学生网络的整体损失同样分为两个部分，第一部分为监督学习损失，第二部分为无监督学习损失，函数设计如公式（4-15）所示：

$$L = \frac{1}{|D_L|}\sum_{y,x_l\in D_L} L_{sup}(y, f_\theta(x_l)) + \lambda\frac{1}{|D_u|}\sum_{x_u\in D_u} L_{cons}\left(f_\theta(x_u^S), f_{\theta'}(x_u^T)\right) \tag{4-15}$$

其中D_L为训练 Batch 中的有监督样本集合，D_u为无监督样本集合，θ为学生网络的权重，θ'为教师网络的权重，λ 为一致性损失的权重调节系数，在训练前期模型不稳定时 λ 设置为 0.5，随着训练时间增加逐渐增大至 1。

学生网络同时使用有监督损失和一致性损失进行权重更新，而因为教师网络的权重是由学生网络滑动平均而来，所以教师网络的性能也会随着学生网络的性能提升而提升，如图 4-12 为两个网络的更新过程，两个网络互相影响，呈螺旋式上升。

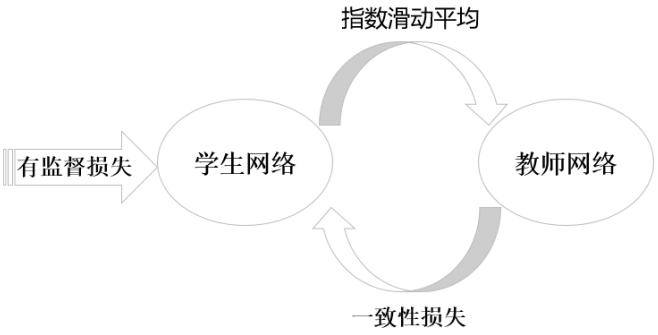


图 4-12 学生-教师网络更新过程示意图

4.4.3 一致性正则损失

上一小节对算法的整体框架进行了概括性介绍，本小节将对其中的一致性损失的具体计算方式进行详细介绍。从上一节中可以看出，本算法最大的特点就是，由于无标签样本不存在预设的标签，所以无法像监督学习那样对 Anchor 预先判定正负，然后与真实框做 IOU 对比计算损失，因此我们的无监督损失是通过计算学生网络与教师网络的输出间的一致性来获得的，又称为一致性损失。两个网络的结构完全一致，是两阶段网络，但是损失函数与两阶段网络的损失函数不同，在这里我们将整个网络看作一个端到端的整体，只对最后的 RCNN 层的输出进行一致性损失计算。不对 RPN 阶段的区域建议进行框损失计算是因为 RPN 阶段生成的区域建议非常繁多，很难采取一种合适的方式将两个网络生成的所有区域建议进行匹配来计算损失。特别地，为了控制 RCNN 阶段学生网络和教师网络输出的框个数一致，学生网络 RCNN 阶段输入的区域建议使用的是教师网络的 RPN 生成的区域建议，即共享区域建议。

对 RCNN 生成的回归框进行损失计算，就是将学生网络生成的预测框分别与对应的教师网络生成的预测框计算 GIOU 损失，值得注意的是，对于空间几何变换类的数据增强操作，会对图像进行翻转旋转等操作，会改变目标在图像中的位置，同时也就会改变产生的检测框的位置，所以我们要保证学生网络的预测框和教师网络的预测框对应在图像上的语义信息应该是一致的，我们称为预测框对齐，需要进行预测框对齐的情况及对应的处理策略分为三类：（1）学生网络使用了空间几何变换，教师网络没有使用，那么就对教师网络产生的预测框做同样的空间几何变换。（2）学生网络没有使用空间几何变换，教师网络使用了，那么就对教师网络产生的预测框做反向的空间几何变换。（3）学生网络使用了空间几何变换，教师网络也使用了，那么就先对教师网络产生的预测框做反向的变换，然后再做与学生网络同样的空间几何变换。这样才能保证两个网络对比的预测框是对应在同样的图像语义上的，因此 GIOU 损失计算需要在经过了预测框对齐之后再进行。

RCNN 阶段的损失除了回归框损失之外还会计算分类损失，整体损失公式定义如公式（4-16）所示：

$$L_{cons} = L_{Softmax}(p^S, p^T) + L_{GIOU}(b^S, b^T) \qquad (4-16)$$

其中p^S, p^T分别为学生网络和教师网络的分类预测，b^S, b^T分别为学生模型和教师模型的框预测。

4.4.4 实验与分析

提出了基于 Mean Teacher 的目标检测训练框架后，我们对该训练框架进行实验，每轮训练 30epoch，初始学习率为 0.001，按指数衰减至 0.00001，batch size 设为 4，使用 adam 优化器进行反向梯度传播，IOU 设为 0.5。特别地，教师滑动更新系数α在前 12 个 epoch 设为 0.9，剩余的 epoch 设为 0.999，之所以采用这样的策略，是因为学生网络在训练初期进步很快，因此教师网络应该很快忘记那些陈旧的、不准确的学生权重。后来，学生网络的提升变慢了，老师网络应该采用缓慢的更新以拥有长期的“记忆”。对于监督学习，我们选择了使用 10%有标签数据（745 张）和 20%有标签数据（1490 张）两种情况，对于无监督学习，我们选择了使用 10%无标签样本（745 张）、20%（1490 张）、30%（2235 张）和所有剩余无标签样本四种情况，实验结果如表 4-9 所示：

表 4-9 一致性正则化方法结果

初始状态	初始 mAP	10%无标签	20%无标签	30%无标签	所有无标签
10%有标签	40.63%	43.18%	45.22%	46.17%	50.83%
20%有标签	62.87%	65.74%	68.15%	70.23%	73.46%

观察表中的结果，可以看出在相同的有标签数据下，无标签样本越多，模型达到的 mAP 越高，说明加入针对无标签样本的一致性正则损失确实有效地帮助了模型权重朝着更好的方向更新。将本方法与慢启动的自训练方法进行比较，发现使用数量相同的无标签样本，一致性正则方法达到的效果更好（初始状态都为 10%有标签数据，且都使用 30%的无标签样本时，一致性正则化方法与慢启动方法的 mAP 对比为 46.17% ：44.06%，高出 2.11%）。对比两个方法的区别，自训练算法挑选无标签样本制作伪标签加入数据集合中进行全监督训练，一致性正则化方法不产生伪标签，直接使用无标签样本进行无监督训练，并且一致性正则化方法约束学生网络和教师网络在引入噪声的情况下输出一致的预测，增强了网络对噪声的鲁棒性，平滑了网络的输出，在一定程度上防止了网络过拟合，因此获得了更高的 mAP 结果。

4.5 本章小结

本章首先对实验所用的遥感数据集进行了划分，然后开始探究不同的半监督学习方法对遥感目标检测算法的适用性。

首先，我们对简单自训练方法进行了实验，发现在目标检测任务下简单自训练方法对于伪标签质量要求很高，依赖于检测器性能，并且需要对超参数置信度阈值进行仔细地调参，使用的成本过高，并且效果也很难保证。于是接下来我们开始对

自训练方法进行改造，提出了慢启动的自训练方法，分步挑选噪声更少的伪标签加入训练集中，并将伪标签损失与真实标签损失分开计算，经过实验发现改造后的自训练方法获得了一定的效果，但是性能提升有限，分析原因是因为添加的样本都是容易识别的简单样本，对网络学习整体数据分布帮助不大。于是下一步我们开始同时关注数据中的难样本和简单样本，引入了主动半监督学习框架，并提出了一种基于委员会的不确定度采样策略，该策略针对网络的预测采用 MD 算法构建每个目标的委员会，然后在委员会内部计算目标的分类和定位不确定度，最后将目标不确定度聚合为样本不确定度。主动半监督学习框架采样出样本中不确定度高和不确定度低的样本后，利用人工标记和半监督学习模型自动标记两种方式一同产生质量很高的伪标签加入训练集中进行训练。本文通过实验证明主动半监督自训练方法效果很好，但存在一定的局限性就是需要人工的参与，但本文通过实验证明，在相同的人工参与下，本文提出的采样策略比随机采样的策略获得了更好的结果。

其次，我们对半监督学习的另一种方法——一致性正则方法进行实验，提出了一种基于 Mean Teacher 的教师-学生网络半监督训练框架，同时对无标签样本设计了一致性损失函数加入网络损失中，通过实验证明该方法也能对模型性能带来一定的提升。

第五章 总结与展望

5.1 本文工作总结

近年来，随着深度学习技术的不断发展，目标检测任务作为计算机视觉的研究热点，正被广泛应用于各行各业。而目标检测与遥感图像领域的结合，进一步推动了遥感图像处理与应用的发展。对于遥感目标检测任务而言，由于遥感目标与自然场景下的目标存在较大差别，在自然场景下效果良好的模型在遥感场景下并不一定适用。所以，针对遥感目标的特点进行网络定制化修改，是本文研究的一大重点。同时，由于遥感数据的海量特性以及目标检测数据标签的标注复杂性，要完成所有数据的标注往往会耗费极大的时间和资金成本。因此，引入半监督学习技术，有效地利用无标签数据帮助模型训练，降低人工标注成本，提高数据利用效率，对于遥感目标检测任务也存在着极大的研究意义。因此，本文主要工作分为两大部分：

第一部分集中在第三章，内容为针对遥感目标特性的网络优化设计。本章首先对遥感目标的特点进行了分析，然后针对遥感目标的特点，分析 Faster R-CNN 网络本身的不足，提出了四点优化策略：第一点，改进网络结构，借鉴 FPN 的思想，将网络结构修改为多尺度预测结构，不同层级的特征图负责预测不同大小的目标。第二点，使用更符合数据集分布的 Anchor 设置，使用 K-Means 算法针对数据集的 Ground Truth 框进行聚类，将结果作为网络的 Anchor 设置。第三点，添加注意力机制，基于 SE 结构进行修改，设计出了空间注意力模块 SAM 和通道注意力模块 CAM，并按不同顺序应用在不同的层的特征图上。第四点，改进损失函数，将 $SmoothL_1$ 损失改为与回归框评价指标更一致的 GIOU 损失。将原始的 Faster R-CNN 网络与我们改进后的 RF R-CNN 网络进行比较，mAP 获得了 3.84%的提升。另外我们也通过控制变量实验证明了这四个优化策略都分别在一定程度上为模型性能带来了提升。

第二部分集中在第四章，内容为基于半监督的遥感目标检测算法研究。分别对半监督学习中的自训练方法和一致性正则化方法进行了实验和改进。其中基于自训练方法的实验是递进式的研究：第一步，我们将最基本的简单自训练方法迁移到目标检测任务后，通过实验结果分析得出，自训练方法对生成的伪标签质量要求很高，而伪标签质量依赖于检测模型的初始精度和对超参数置信度阈值的精细调参。第二步，我们提出了一种慢启动的自训练方法。不同于简单自训练中一次性使用全部无标签样本生成伪标签，慢启动自训练方法使用某种方法挑选出部分伪标签质量较好的样本分步加入训练集，并且将真实标签损失与伪标签损失分开计算。该方

法对模型的性能带来了一定的提升，在初始数据为 10%有标签样本的情况下，添加 30%的伪标签为模型的 mAP 带来了 3.43%（40.63%→44.06%）的提高，但因为挑选出的样本都是容易识别的简单样本，对于模型学习数据的整体分布帮助不大，所以提升幅度有限。第三步，由于简单样本对模型提升帮助有限，所以我们开始同时关注数据中的难样本和简单样本。由于通过半监督方法对难样本产生的伪标签中一定带有很大噪声，所以我们引入了主动学习的思想，引入主动半监督自训练框架，采样出高不确定度的样本交由人工产生标签，采样出低不确定度的样本交由模型产生伪标签，再一同加入训练集中进行训练。在此过程中，本文提出一种基于委员会的不确定度采样策略，该策略针对模型的预测框采用 MD 算法构建每个目标的委员会，然后在委员会内部计算目标的分类和定位不确定度，最后将目标不确定度聚合为样本不确定度。实验证明主动半监督学习的自训练方法可以对模型带来很大的提升，在初始数据为 10%有标签样本的情况下，进行 30%的标签样本扩增（其中 15%为主动学习标签扩增，15%为半监督学习标签扩增）为模型的 mAP 带来了 39.93%（40.63%→80.56%）的提升。由于引入了人工的标记，相当于加入了真实标签，所以提升较大是合理的。本文通过实验，比较了使用基于委员会不确定度采样策略和使用随机采样策略的效果，在初始数据为 10%有标签样本的情况下，进行 30%的标签样本扩增，使用本文提出的采样策略获得的 mAP 比使用随机采样的 mAP 高 3.37%，证明本文提出的采样策略比随机采样效果更好。

综上所述，总结本文对自训练方法的研究，要使该方法有较好的效果需要注意两个方面的问题：（1）产生的伪标签噪声要少。（2）加入训练的无标签样本应该对模型性能提升有关键影响。

另外，除了自训练方法外，本文还对不需要产生伪标签的一致性正则化方法进行了改造和实验。提出了一种基于 Mean Teacher 的半监督训练框架，对有标签样本进行监督训练，对无标签样本进行无监督训练，并设计了针对无标签样本的一致性损失。实验结果证明该方法也在无标签样本的帮助下对模型的性能带来了提升，在初始数据为 10%有标签样本的情况下，使用 30%的无标签样本为模型的 mAP 带来了 5.54%（40.63%→46.17%）的提高。并且该方法与同样没有人工参与的慢启动的方法相比，在使用的无标签样本数量相同的情况下，mAP 提高幅度高出 2.11%（44.06：46.17%），效果更好。

5.2 未来工作展望

半监督学习在遥感目标检测领域的应用具有十分广阔的前途，本文虽然在两个方面对该任务进行了改进和实验，但依然存在很多工作可以探索，如：

1. 更贴近目标的旋转框检测。由于本文使用的数据集限制，我们仍然使用的是平行于图像四边的正框进行检测，但如果采用带有角度的旋转框进行检测，就可能更好地贴近目标，检测框中的背景像素就更少，会对后续的任务有一定的帮助。

2. 自训练与一致性正则结合的半监督学习。半监督学习在分类领域上已经获得了一定的突破，近两年提出的效果优秀的 FixMatch[44]等论文都是将自训练方法与一致性正则方法进行结合，我们也可以考虑在目标检测领域尝试对两种方法进行结合。

3. 更好的主动学习采样策略。本文提出了一种基于委员会的不确定度采样策略，在图像的目标级别对目标的不确定度进行刻画，然后聚合为图像的不确定度，我们还可以尝试其他的方法对图像的不确定度进行刻画，设计出更好的策略采样出对学习影响效果最大的样本。

4. 引入带噪学习的方法。由于半监督学习对于无标签样本的使用常常会产生质量不佳的伪标签，因此我们可以引入带噪学习的方法，增强模型对噪声的鲁棒性，缓解伪标签质量不佳对模型训练带来的影响。

致 谢

回首三年研究生时光，如白驹过隙，转瞬即逝。我们也即将踏出校园，步入社会，面对人生新的阶段。在这三年校园学习时光里，我收获了很多，也成长了很多。在此，我想对这三年里一直帮助支持我的人表达最衷心的感谢！

首先，我要感谢我的导师段翰聪老师。感谢您为我们提供了一个氛围良好、有很多学习分享、大家一起共同进步的学习环境，给了我们很多展示和锻炼自己的机会。您对我们的指导和关怀，不仅体现在平时的学习和生活中，还有在人生道路的选择上。您会不吝地为我们这群初出茅庐的学生传授经验，给出建议。您对待工作的努力，对待知识的热情，让我明白了学习是一个永无止境的过程，在今后的日子里，我也会像您一样继续在自己的专业领域内不断钻研，不断提升，成为更好更优秀的自己！

其次，我要感谢实验室里的师兄师姐们、同学们。在刚刚开始接手项目时我毫无经验，感谢师兄师姐们的指导和帮助，让我很快对工作内容熟悉起来。还有感谢各位同学们在周末的分享会和平时的交流分享，在和大家的交流中我获益良多！

然后，我要感谢我的大家庭的家人们。感谢父母多年来对我的养育和栽培，感谢家里长辈们对我的关怀和支持，感谢哥哥姐姐们对我的照顾和爱护，正是因为有家人作为我最坚实的后盾，我才能对自己充满信心，对未来充满希望！

最后，感谢各位评审专家，感谢你们的辛勤工作和对本篇论文的专业指导意见！

感谢以上提到的所有老师、同学和家人们！谢谢你们！

参考文献

[1] Y. Wang. An Analysis of the Viola-Jones Face Detection Algorithm[J]. Image Processing On Line, 4 (2014), pp. 128–148. 2014

[2] R. Girshick, J. Donahue, T. Darrell, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C].IEEE Conference on Computer Vision and Pattern Recognition,2014:580-587.

[3] R. Girshick. Fast r-cnn[C]. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015: 1440-1448.

[4] J. Redmon, S. Divvala, R. Girshick, et al. You Only Look Once: Unified, Real-Time Object Detection[J].Computer Vision and Pattern Recognition,2016:779-788.

[5] T. Lin, P. Goyal, R. Girshick, et al. Focal Loss for Dense Object Detection[C]. IEEE International Conference on Computer Vision (ICCV), 2017.

[6] N. Dalal, B. Triggs. Histograms of Oriented Gradients for Human Detection[C]. International Conference on Computer Vision Pattern Recognition, 2005, pp.886–893, 10.1109

[7] A. Krizhevsky, I. Sutskever, G. Hinton. Imagenet classification with deep convolutional neural networks[J]. In Advances in Neural Information Processing Systems 25, 2012.

[8] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition[C]. International Conference on Learning Representations, 2015.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[10] K. He, X. Zhang, S. Ren, et al. Deep Residual Learning for Image Recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[11] X. Zhang, X. Zhou, M. Lin, et al. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices[J]. 2017.

[12] Z. Li, C. Peng, G. Yu, et al. DetNet: A Backbone network for Object Detection[C]. European Conference on Computer Vision. 2018.

[13] Kai Han, Yunhe Wang, Qi Tian, et al. GhostNet: More Features from Cheap Operations. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[14] B. Li, J. Yan, W. Wu, et al. High Performance Visual Tracking with Siamese Region Proposal Network[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[15] A. Etten. You Only Look Twice: Rapid Multi-Scale Object Detection In Satellite

Imagery[EB/OL]. http:// arxiv.org/abs/1805.09512. 2018.

[16] 王彦情, 马雷, 田原. 光学遥感图像舰船目标检测与识别综述[N]. 自动化学报, 2011.

[17] 屠恩美, 杨杰. 半监督学习理论及其研究进展概述[N]. 上海交通大学学报, 2018.

[18] D. Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks[C]. International Conference on Machine Learning, volume 3, page 2, 2013.

[19] S. Laine, T. Aila. Temporal Ensembling for Semi-Supervised Learning[C], International Conference on Learning Representations, 2017.

[20] M. Takeru, M. Shin-Ichi, I. Shin, et al. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018:1-1.

[21] A. Tarvainen, H. Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results[C]. Conference and Workshop on Neural Information Processing Systems. 2017

[22] D. Berthelot, N. Carlini, I. Goodfellow, et al. MixMatch: A Holistic Approach to Semi-Supervised Learning.Conference and Workshop on Neural Information Processing Systems. 2019.

[23] 杜泽星, 殷进勇. 基于半监督学习的遥感飞机图像检测方法[J]. 激光与光电子学进展. 2020.

[24] M. Everingham, L. Van-Gool, et al. The Pascal Visual Object Classes (VOC) Challenge[J]. International Journal of Computer Vision, pages 303-338, June 2010.

[25] Y. LeCun, L. Bottou, Y. Bengio, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE. 86(11): 2278 - 2324.1998.

[26] S. Ren, K. He, R. Girshick, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]. In Proc. Advances in Neural Information Processing Systems (NIPS). 2015: 91-99.

[27] W. Liu, D. Anguelov, D. Erhan, et al. SSD: Single Shot MultiBox Detector[J]. IEEE Conference on Computer Vision and Pattern Recognition,2016:21-37.

[28] Z. Zou, Z. Shi. Random Access Memories: A New Paradigm for Target Detection in High Resolution Aerial Remote Sensing Images[J]. in IEEE Transactions on Image Processing, vol. 27, no. 3, pp. 1100-1111, March 2018.

[29] Y. Zhang, Y. Yuan, Y. Feng, et al. Hierarchical and Robust Convolutional Neural Network for Very High-Resolution Remote Sensing Object Detection[J]. in IEEE Transactions on Geoscience and Remote Sensing, vol.57, no.3, pp.5535-5548, 2019.

[30] H. Zhu, X. Chen, W. Dai, et al. Orientation Robust Object Detection in Aerial Images Using Deep

Convolutional Neural Network[C]. IEEE Int'l Conf. Image Processing, 2015.

[31] Y. Long, Y. Gong, Z. Xiao, et al. Accurate Object Localization in Remote Sensing Images Based on Convolutional Neural Networks[J]. in IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 5, pp. 2486-2498, May 2017.

[32] M. Zeiler, R. Fergus. Visualizing and Understanding Convolutional Networks[C]. European Conference on Computer Vision. Springer, Cham, 2014:818-833.

[33] T.Y. Lin, P. Dollár, R. Girshick, et al. Feature Pyramid Networks for Object Detection[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[34] M. Jaderberg, K. Simonyan, A. Zisserman. Spatial transformer networks[C]. Advances in Neural Information Processing Systems. 2015: 2017-2025

[35] J. Hu, L. Shen, G. Sun. Squeeze-and-Excitation Networks[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[36] F. Wang, M. Jiang, C. Qia, et al. Residual Attention Network for Image Classification[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[37] J. Yu, Y. Jiang, Z. Wang, et al. UnitBox: An Advanced Object Detection Network[C]. Proceedings of the 24th ACM international conference on Multimedia. October 2016.

[38] H. Rezatofighi, N. Tsoi, J. Gwak, et al. Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[39] L. Liu, Z. Pan, B. Lei. Learning a Rotation Invariant Detector with Rotatable Bounding Box[C]. Computer Vision and Pattern Recognition. 2017.

[40] G. Xia, X. Bai, J. Ding, et al. DOTA: A Large-scale Dataset for Object Detection in Aerial Images[C]. Conference on Computer Vision and Pattern Recognition. December 2018.

[41] S. Ioffe, C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]. Proceedings of The 32nd International Conference on Machine Learning, pages 448–456, 2015

[42] E. Cubuk, B. Zoph, D. Mane, et al. AutoAugment: Learning Augmentation Policies from Data[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[43] B. Zoph, E. Cubuk, G. Ghiasi, et al. Learning Data Augmentation Strategies for Object Detection[EB/OL]. https://arxiv.org/abs/1906.11172. 2019.

[44] K. Sohn, D. Berthelot, C. Li, et al. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence[EB/OL]. https://arxiv.org/abs/2001.07685. 2020.

攻读硕士学位期间取得的成果

【攻读硕士期间参与的科研项目】

[1] ******技术研究研制（保密）