中国科学院大学

University of Chinese Academy of Sciences

硕士学位论文

基于动态神经网络的弱小目标检测

作者姓名：彭潇珂

指导教师：韩振军 副教授

中国科学院大学

学位类别：工程硕士

学科专业：电子与通信工程

培养单位：中国科学院大学电子电气与通信工程学院

2022 年 6 月

A Tiny Object Detection Method Based on Dynamic Neural Network

A thesis submitted to

University of Chinese Academy of Sciences

in partial fulfillment of the requirement

for the degree of

Master of Engineering

in Electronic and Communication Engineering

By

Peng Xiaoke

Supervisor: Han Zhenjun

School of Electronic, Electrical and Communication Engineering

University of Chinese Academy of Sciences

June, 2022

中国科学院大学
学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明或致谢。

作者签名：彭潇珂

日　　期：2022年5月20日

中国科学院大学
学位论文授权使用声明

本人完全了解并同意遵守中国科学院有关保存和使用学位论文的规定，即中国科学院有权保留送交学位论文的副本，允许该论文被查阅，可以按照学术研究公开原则和保护知识产权的原则公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、汇编本学位论文。

涉密及延迟公开的学位论文在解密或延迟期后适用本声明。

作者签名：彭潇珂　　导师签名：韩振华

日　　期：2022年5月20日　　日　　期：2022年5月20日

摘 要

弱小目标检测（Tiny Object Detection, TOD）作为一个新兴的任务，在诸多领域都有应用前景，在研究中也极富科研价值。科技的快速发展使得弱小目标检测已经应用到如自动驾驶、医学图像分析、安防监控等多个领域中。对于弱小目标检测任务，放大图像具有十分重要的作用。然而，若只是一味地放大图像，在检测目标的过程中，计算成本和负样例数量将会大幅增加，从而导致检测性能降低，并限制检测器的适用性。为了缓解上述问题，本文对此问题进行了深入探索，并提出采用一种自适应的方式调整网络检测时特征图的大小，以权衡检测的准确性和效率，实现计算资源的动态分配。本文主要工作如下：

1、提出了一种用于弱小目标检测的动态池化网络（Dynamic Pooling Network, DPNet），将动态神经网络应用于目标检测任务中。DPNet 通过引入降采样因子，从而在特征图上实现灵活的降采样策略，将卷积神经网络中固定的降采样过程改变为更自由的方式。

2、设计了一个轻量级预测器网络来调整检测器中骨干网络的降采样因子，从而自适应地降低特征图的大小，实现了对目标尺寸具有感知的降采样操作。设计了混合降采样因子训练，在此训练过程中采用自适应标准化模块（Adaptive Normalization Module, ANM）使一个统一的检测器可以兼容不同的降采样因子。同时本文还设计了指导损失来为预测器网络的监督信息生成提供了帮助。

3、TinyCOCO 上的实验结果表明，本研究设计的方法可以适用于检测器中不同的骨干网络，在降低计算量的同时不降低性能，甚至有提升性能的潜力，值得一提的是，该方法还可与轻量级骨干网络共同使用，具有适用性与可叠加性。

关键词：计算机视觉，深度学习，弱小目标检测，动态神经网络

Abstract

As an emerging task, tiny object detection (TOD) has application prospects in many fields and is also of great scientific research value. With the rapid development of science and technology, tiny object detection has been applied to many fields, such as automatic driving, medical image analysis, security monitoring and so on. Zooming in on images plays an essential role in object detection, especially for tiny objects. However, simply doing so causes a substantial increase in the computational cost and negative samples, which heavily deteriorates detection performance and limits its applicability. In order to alleviate the above problems, this paper makes an in-depth exploration, and adopts an adaptive way to adjust the size of the feature maps during inference, so as to make a trade-off between accuracy and efficiency and realize the dynamic allocation of computing resources. The main contribution of this work is as follows:

1. This paper proposes a *Dynamic Pooling Network (DPNet)* for tiny object detection, which is the first attempt to introduce the dynamic neural network to object detection task. *DPNet* uses a flexible down-sampling strategy by introducing down-sampling factor to relax the feature map's fixed down-sampling process to an adjustable fashion.

2. Furthermore, we design a lightweight predictor to adjust down-sampling factor in the backbone, decreasing the feature map. Therefore, we reach an object-size-aware down-sampling. We design the mixed scale training with an *adaptive normalization module (ANM)* to make a union detector compatible with different down-sampling factors. At the same time, we also design the guidance loss to help the generation of supervision information of predictor network.

3. Experiments on TinyCOCO demonstrate that our *DPNet* can save computation cost while maintaining comparable detection performance. It is worth mentioning that our method can also be combined with lightweight backbone, which shows its practicability.

Keywords: Computer Vision, Deep Learning, Tiny Object Detection, Dynamic Neural Network

目 录

图形列表

表格列表

符号列表

缩写

DPNet	Dynamic Pooling Network
ANM	Adaptive Normalization Module
MST	Mixed Scale Training
DFP	Down-Sampling Factor Predictor
AP	Average Precision
AR	Average Recall

第 1 章　引言

1.1　研究背景及意义

计算机视觉（Computer Vision）是指运用各种不同成像系统来作为输入敏感手段，用计算机代替大脑对图像输入进行处理和解释。计算机视觉的最终目标是使计算机像人类一样通过视觉来观察和理解世界，并具有独立适应环境的能力。但能够真正使计算机通过摄像机感知世界是十分困难的，因为虽然摄像机拍摄的图像和平时所见一致，但对于计算机来说，任何图像都只是像素值的排列组合，是数字。如何让计算机从这些死板的数字里面读取到有意义的视觉线索，是计算机视觉应该解决的问题。在达到最终目标前，当前研究的中期目标是建立一个合理的视觉系统，能够根据视觉敏感和反馈的某种程度的只能完成一定的任务，例如自动驾驶车辆的视觉导航。计算机视觉系统的基本处理流程为由图像采集设备（相机或摄像头等）获取图像，利用计算机进行一些预处理和特征提取操作，接而对图像中的目标进行感知识别以及语义上的理解，最终处理输出人类可以理解的信息。计算机视觉既是工程领域，也是科学领域中一个极具挑战性的重要研究领域。作为一门综合性的学科，计算机视觉已经吸引了多个学科的学者参与到对其的研究之中。

目前，计算机视觉技术发展迅速，已具备初步的产业规模。在工业界，诸如人脸识别、字体车牌等任务也已经逐渐成熟应用于日常生活。计算机视觉技术涉及到的场景和产业诸多，由此分化的任务也多种多样，除了上面简要描述的以外，计算机视觉领域还有一个典型任务——目标检测。目标检测任务的目标是给定一张图像或是一个视频帧，让计算机找出其中所有目标的位置，并给出每个目标的具体类别。这个任务在图像分类的任务基础上增加了图中物体位置回归这一任务。图像分类是根据图像的语义信息对不同类别图像进行区分，而目标检测将图片级别的分类任务升级为物体级别的分类与回归任务，产生的位置信息对实际应用有重大意义。因此目标检测被作为一个基础任务被广泛研究，且衍生了不同场景下的具体目标检测任务，如通用检测、行人检测、人脸检测、弱小目标检测等。

作为目标检测任务的一个研究分支，弱小目标检测具有广泛的落地应用前

景。在视频监控、自动驾驶辅助和快速海上救援等场景中，需要检测的目标具有弱小目标的特征和复杂的多样性。另外，对于无人驾驶这一任务，越清晰的摄像头使得机器能够获取更远距离的图像，从而能够更早地做出一些对应的处理和操作，但众所周知，越远的目标在画面中就会越小，弱小目标检测在这种场景中就是一个亟待解决的问题。除此之外，在很多国防军事任务中，如精确防卫、精确打击或边境安全监测等任务中，弱小目标检测都是场景需求，比如在遥感图像处理中，船只和汽车往往只占很小部分的像素点，同时这类任务也对弱小目标检测的精度也提出了比较高的要求。通用目标检测一般多是研究大尺度的物体，很难适用于这些任务，因此弱小目标的研究可以帮助目标检测在尺度上形成更全面的体系，同时对于实际应用场景中更是意义重大，其重要性不亚于对现有通用目标检测的研究。图1.1为弱小目标检测典型场景，图片来源于 TinyPerson 数据集 [3]。

图 1.1 弱小目标检测场景示例

Figure 1.1 Sample of tiny object detection scene

从研究的角度，弱小目标检测作为目标检测的子任务，是通用目标检测在尺度研究上的一个细化拓展，通用目标检测中往往对这一尺度的物体很少涉及，因为其数据集都是近距离拍摄的图片，图中的目标尺度都比较大，不会涉及弱小目标检测，但弱小目标检测在实际的视觉任务中是一个常见的现象，所以对弱小目标检测的研究是对通用目标检测的一个很好补充。

另外，无人机无论在军用或者民用的领域中应用也越来越广泛 [6]，随着无

人机技术的成熟与商业化，基于无人机的相关技术也越来越受到重视，基于航拍目标检测完成自动化监测等就是其中之一。航拍通常会配备极度高分辨率的摄像头以获得高质量的图像，但在进行高空拍摄地面目标时，目标分辨率依旧会很低，因为图像是远距离广视角下产生，目标所占像素点数就非常少，导致目标所占像素边界模糊、易受噪声干扰，因此航拍目标检测就是天然的弱小目标检测。人作为经常被检测的对象，对于人体的检测是弱小目标检测的一个重要应用分支，其应用场景也很多，比如用于海难或其他灾难中的搜救任务，比如用于监控分析中或者用于日常娱乐中的定位任务，还可以用于海岸情况监控，控制海滩人员密度，保持安全的社交距离。另外，弱小人体目标检测虽然研究的对象是人体，但其中人体姿态、角度等具有很大的多样性，可以为其他的弱小目标检测提供参考。在 VisDrone[7] 数据集中详细描述了无人机航拍的应用场景，比如用于太阳能发电厂对太阳能电池板的缺陷检测，机器检测替代人工排查可以大幅度提高工作效率；或者用于植物的早期病害的检测；还可以用来公共安全领域的鲨鱼侦测。对于这些应用场景中的目标也同样具有弱小目标的特征和复杂的多样性，因此很多在弱小人体目标检测中的方法应当也能对这个问题起到很大的参考作用。对于很多国防军事任务，例如精确防卫或精确打击或者是舰船检测，或边境安全监测等任务中，图片卫星采集，或者为不同信息源的遥感目标，相较于无人机拍摄的图片，距离更远导致目标也更小，这也是弱小目标检测任务的一种。但是在这种任务中，由于任务的特殊性，对目标的识别也提出了更高的精度要求，也说明了弱小目标检测的重要性。因此对于弱小目标的研究对目标检测在尺度上形成更全面的体系有很大意义，无论在军用、商用、民用或者日常生活领域中，弱小目标都是一直存在的情况，对各种应用场景都有着重大意义。

1.2　本文研究内容

弱小目标检测一直以来在目标检测研究领域中都是一个具有挑战性的课题。弱小目标检测任务旨在精准地检测出尺度小、信噪比低的目标。随着对卷积神经网络（Convolutional Neural Network, CNN）研究的逐步深入，如今的检测器[8–12] 的性能有了大幅的发展和进步。

相比起通用目标检测，弱小目标检测任务中目标的绝对尺度小，一个合理的方法是在预训练时调整目标尺度的大小到和弱小目标基本一致，SM[3] 和

SM+[13] 就是通过这样的途径来提升小目标检测的性能。但是这样的方法需要在预训练时对图像或目标进行一定的缩小操作，该操作会带来信息的丢失。

现有的检测器多基于卷积神经网络实现，图片在输入网络后，会在不同的阶段被降采样，这不可避免地带来了一定的信息丢失，而对于弱小目标检测任务来说，这样的信息丢失比通用目标检测任务更为致命。所以在弱小目标检测任务中，一个简单解决该问题的方法就是放大输入图像。

以图1.2为例，图中绿框内的物体为原图和放大输入图像训练过的检测器均能检测出的目标，红框内的物体为仅在经过放大图像训练的检测器能正确检测出的目标。可以从此图看出，当放大检测网络的输入图像时，检测器将能更好地检测出小尺度的目标。因此，合理地放大输入图像或者减少网络中的降采样操作均能提升检测器在弱小目标上的检测性能。

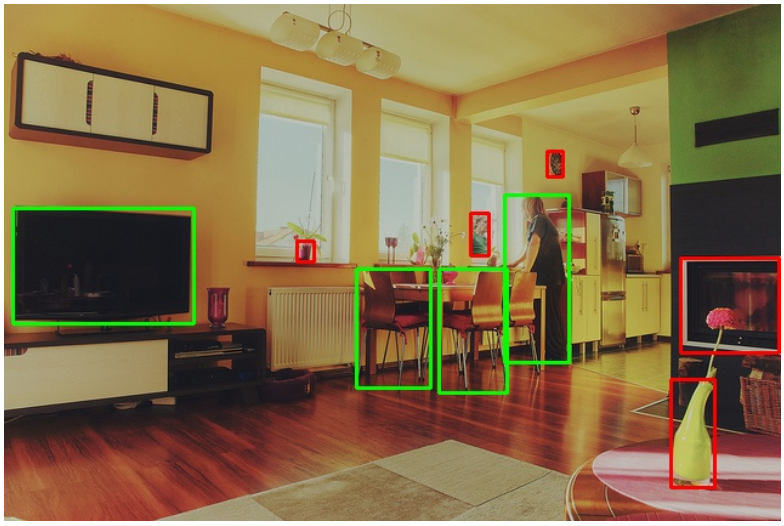


图 1.2 放大输入图像可以更好地检测小物体

Figure 1.2 Small objects can be better detected through zooming in on images

然而一味地放大输入图像也会带来一些副作用：

1）更多的计算开销，对图像的尺度放大一倍，检测网络的计算量也会成倍地增长；

2）更多的负例样本，由于基于 anchor 的检测器的滑窗特性，放大图像会无法避免地增加负例样本的数量；

3）更多的冗余，以图1.3为例，图中橙色的部分表示由原图和放大图像分别训练过的两种检测器共同正确检测出的目标的尺度分布图，蓝色的部分为仅由

放大图像训练过的检测器正确检测出的目标尺度分布图，图1.3里的左中右分别表示放大图像时为 2 倍，4 倍和 8 倍尺度大小，该图可以说明，对于一些目标，不需要将输入图像放大也能被正确检测。

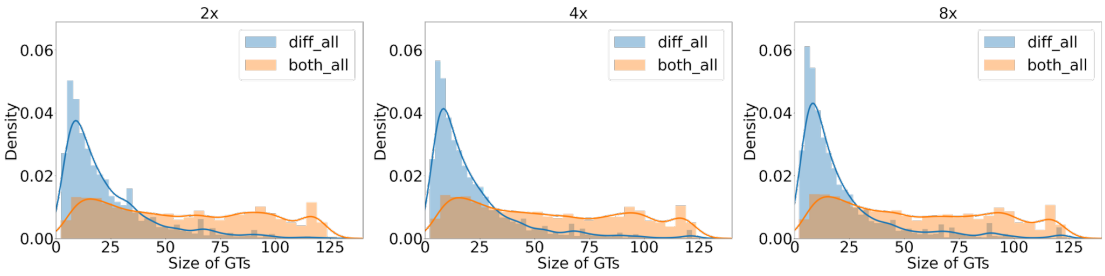


图 1.3 放大输入图像会产生一定的冗余

Figure 1.3 Zooming in on the image creates a certain amount of redundancy

为了在计算速度和性能之间取得一个好的平衡并消除简单放大数据集中所有图像造成的冗余，本文设计了一个具有可变降采样因子的动态神经网络，称为动态池化网络（Dynamic Pooling Network, DPNet）。传统的（静态）神经网络测试期间对所有输入图像使用相同的降采样因子进行推理，与之不同，DPNet 在在降采样因子方面采用了自适应推理。为了获得适当的降采样因子，本文还设计了一个降采样因子预测器（Down-sampling Factor Preditor, DFP），并将其嵌入到检测器的骨干网络中。在实际操作中，几个不同的降采样因子被设置为了候选。预测器的输入为特征图，在候选降采样因子上产生一个概率分布作为输出，并选择最佳的降采样因子。然后在骨干网络的下一个阶段，特征图将按所选的降采样因子进行缩放。DFP 的参数量和计算量足够小，所以几乎可以忽略其计算成本。对于检测任务，识别样本的固有难度水平以及图像中实例的大小和数量可能会影响正确检测所需的信息量。因此，在设计 DFP 时，本文还引入了一个分支来预测输入图像中目标的统计值，有助于预测的提高合理性和有效性。通过利用本文所提出的 DPNet 推理方法，网络可以从每个特征图的大小中挖掘其空间冗余并进行消除。

1.3　本文主要贡献

本文针对弱小目标检测在放大时可能会增加性能但增大计算量的问题进行了研究和改进，图1.4简要展示了本文所做工作的内容、关系和意义。

本文主要贡献总结如下：

1）提出了动态池化网络（DPNet），这应该是已有研究中首次将动态神经网络引入目标检测任务的尝试，并且该网络能实现检测性能和计算之间的权衡。

2）为了解决混合降采样因子训练网络的过程中带来的尺度差异加重问题，设计了自适应标准化模块（Adaptive Normalization Module, ANM）。在推理过程中，设计了降采样因子预测器（DFP）自适应选择降采样因子，并为它的监督信息生成设计了指导损失。

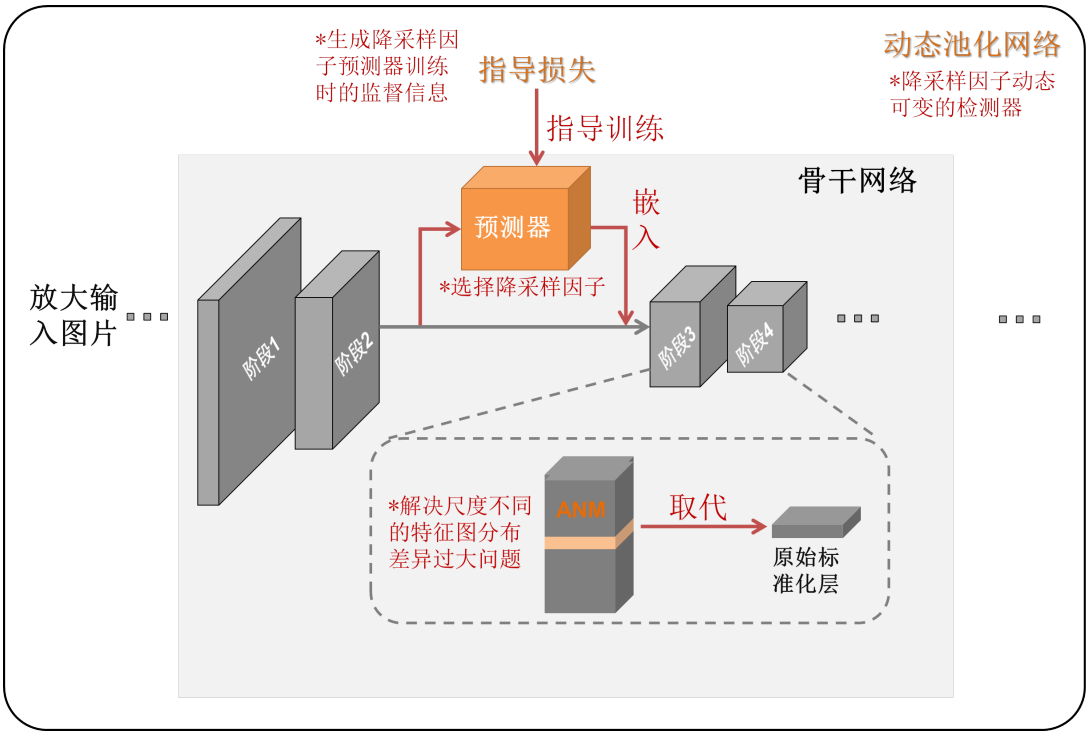


图 1.4 本文工作示意图

Figure 1.4 Working diagram of this paper

本文对于上述的设计内容进行了实验验证和分析。实验结果表明，本文提出的方法能够在维持放大输入图片带来的必需信息量的同时消除其中的冗余。

1.4 本文组织结构

第一章引言，分为研究的背景及意义、本文研究内容、本文主要贡献与本文组织结构四部分。首先简单介绍了计算机视觉这一学科，接而介绍了弱小目标检测的场景、应用前景、科研价值等，描述了目前小目标检测中常见的问题。然后在本文研究内容中阐明了放大输入图像对于小目标检测的具体优势，指出了其

隐含的缺点和不足，同时提出了降采样因子来取代检测器骨干网络中的固定降采样操作，并借此简单介绍了文中设计的动态池化网络。本文主要贡献简要说明了文章的工作和意义。本文组织结构部分按照章节简略介绍了论文各章的内容。

第二章国内外本学科领域的发展现状和趋势，分为了四个部分。第一个部分介绍了目标检测算法的发展以及相关算法和重要影响的工作；第二部分介绍了用于目标检测领域的几种典型数据集，划分为通用目标检测数据集，行人检测数据集，人脸检测数据集和弱小目标检测数据集并介绍代表工作；第三部分介绍了介绍小目标检测相关算法，分为用于多尺度目标检测中的小目标检测和弱小目标检测两个类别；第四个部分介绍了动态神经网络的优势以及相关算法和产生了重要影响的工作，主要从动态结构网络和与本文研究内容相关的分辨率动态网络来阐述。

第三章研究内容与方法，研究内容方面首先简要介绍了弱小目标检测的难点和特点，接着介绍了动态池化网络的研究背景，研究动机。在研究方法方面，首先介绍了动态池化网络的总体框架、降采样因子的作用机制和原理，然后按照训练和推理两个步骤进行详细介绍和算法说明。其中，训练步骤又分为了训练检测器和训练预测器两步，会对训练过程中针对混合尺度训练而特别设计的自适应标准化模块进行详细说明，还有预测器的结构设计和指导损失也会专门做阐释。

第四章实验结果与分析，实验结果方面首先介绍了目标检测领域及论文采纳的评价指标的原理和计算方式，然后介绍了实验的设置。接下来首先通过更换检测器骨干网络的不同参数的实验说明动态神经网络在特征图尺度这一方面进行“动态”是更为合适的，验证了研究动机的合理性，然后分别介绍了实验方法结果对比，DPNet 在更换不同骨干网络的情况下都能体现出计算量的优势且基本维持性能。另外，文章针对每个模块的设计都做了消融实验并验证了各个模块的合理性和可用性。实验分析方面为各个模块有效性的原理分析和起作用的原因分析。

第五章结论与展望，总结了全文的行文逻辑，并且从论文各个不同章节的角度来概括每一章的重点，并且从不同角度展望了未来的研究工作。

第 2 章　国内外本学科领域的发展现状和趋势

在深度学习出现之前，计算机视觉的任务主要是以图像数据计算、特征工程为主的图像压缩、增强、去噪、识别、配准等。深度学习出现之后，基于卷积神经网络的图像处理成为主流，在各个应用方向上均取得了长足的进步，比如图片分类、目标检测、场景分割、目标跟踪等。

本章将从相关的检测算法发展开始阐述，并对有关数据集进行介绍与特点比对，然后介绍小目标检测算法的前沿研究；接而是对动态神经网络的优势以及当前相关研究进行介绍；最后阐述一些用于分类任务或检测器骨干网络的轻量级的卷积神经网络。

2.1　目标检测算法的发展

目标检测是计算机视觉领域中长期存在的一个基本任务，近些年也是研究的重点领域。目标检测的任务一般定义为给定一张图像，判断图中是否存在预先定义的类别实例，如果存在则返回类别和空间的最小外接矩形的坐标，是图像级别的分类任务的扩展，对图片的处理结果也从图像级别转化为实例级别。在 2012 年前，目标检测领域的方法还是以提取人工设计的特征为主流，包括 SIFT[14]、HOG[15]、DPM[16]、HOG-LBP[17] 等，这时候的处理问题流程一般为: 区域选择，特征提取、分类回归三步，虽然在一些问题上取得了初步的成果，但是却有两个难以处理的问题: 一是区域选择算法效果差，计算量大，基于滑动窗口策略需要进行像素级的计算，这就导致窗口数量太大，而且随着像素的数量呈指数级增长，如果为了涵盖多尺度或者不同长宽比的物体，计算量又会成倍增长，无法做到实时性的检测；二是基于人工设计特征提取泛化能力不好，同一个方法在不同的数据上表现差异可能很大。Krizhevsky[18] 等在 2012 提出了用于图片分类的深度卷积神经网络，标志着目标检测也正式进入深度学习时代。

深度学习之后，基于卷积神经网络的检测算法按照处理阶段分类主要可以分为两种: 一是不使用区域提议阶段的单阶段 (one stage) 算法，最后的结果选自于预先在图中密集设定的点（框）；二是基于区域提议的两阶段 (two stage) 算法，这类算法第一步得到无类别差异的前景提议区域，然后对提议区域进行分类和

回归。

在两阶段算法中，Ross Girshick 在 2013 首次提出了影响重大区域卷积神经网络 (RCNN)[19]，这也是卷积神经网络第一次应用于目标检测任务，后续的工作基本都遵循了其两阶段算法的处理思路。RCNN 对传统的基于滑动窗口的算法改进在于:1. 传统的检测算法一个窗口就会完成一次检测过程，但相邻窗口像素重叠大，因此带来了计算冗余，RCNN 使用了一个启发式方法 (Selective search)，先生成候选区域再检测，降低信息冗余程度，从而提高检测效率；2. 使用了卷积网络提取了目标特征避免了传统算法提取特征鲁棒性不够。但是这个算法仍有许多不足，首先，其训练是多阶段的，较为繁琐和耗时；其次，由于在高密度的候选区域上反复进行特征提取，其检测速度很慢 (GPU 下每张图 40 秒，640×480 像素)[20]。

何恺明在 2014 年提出了 SPPNet[21] 进一步改进了 RCNN:1. 提出了先卷积计算再生成候选区域的策略，之前的先生成候选区域再卷积计算的方法仍然在临近区域有大量的重复计算，特征被重复提取，如果先卷积计算就可以通过一次计算满足特征提取的需求；2. 提出了空间金字塔池化层 (Spatial Pyramid Pooling) 的特征池化方法，由于全连接层的存在，经过深度卷积处理后的特征必须处理成相同大小才能输入到全连接层，这就要求图片在输入网络的时候要统一到相同的尺寸，为了将不同大小归一化，一般会采取裁剪或者缩放的手段，但无论是哪种方法都会强制图片变形，破坏图片的结构，而空间金字塔池化层允许输入不同大小的图片输入网络，但在全连接计算前可以将不同大小的特征归一到同一大小，打破图片输入的尺寸必须统一这一束缚。

在 2015 年 Ross Girshick 又提出了 FastRCNN[22] 在 RCNN 的基础上又充分吸收了 SPPNet 的优点，摒弃了单独训练 SVM 的分类器的方式，并且将分类器和回归器并行设计，极大提高了计算速度。同年任少卿提出了第一个端到端的两阶段检测器 FasterRCNN[23]，两阶段算法中第一阶段生成的预选区域的质量直接决定了第二阶段的效果，在以往选择候选区域都采用的是启发式的算法，如果生成的候选区域太多会造成计算冗余，如果太少又会产生误检，而且卷积计算都是由 GPU 实现的的，而启发式算法是在 CPU 实现，这也降低了计算效率，在 Faster RCNN 首次提出了 RPN(Region Proposal Networks)，将提取候选区域用卷积网络实现，大大减少了计算量和耗费时间。并且首次引入了 Anchor(锚点框) 的概念，

通过设定密集分布的 Anchor 判断对应区域是否可以作为目标候选区域，为后来的基于锚点框的方法打下来基础。何恺明在 2017 年又提出了 Mask RCNN[24]，就 ROI Pooling 中造成的特征不对齐问题提出了 ROIAlign 解决，并且将目标分割和特征金字塔引入同一框架中也显著提升了性能。之后有很多文章对这些进行了一处或者多处的改变，比如 RefineDet[25]，CascadeRCNN[8]，Grid RCNN[26]，LibraRCNN[10] 都取得了不错的性能提升。

单阶段算法没有采用了两阶段算法的“粗检测 + 细检测”的模式，直接在由卷积提取的特征上进行分类回归，经过后处理后产生检测结果，和双阶段检测算法相比，虽然性能有所下降但是却极大提升了速度，可以满足实时性需求，代表算法有 YOLO[27] 系列、SSD[28]、RetinaNet[29] 等。

YOLO(You Only Look Once)[27] 是第一个单阶段检测算法，由 Joseph 和 Girshick 等人在 2015 年提出。算法直接将整张图像统一到固定大小作为网络的输入，经过一系列卷积计算后再接全连接层直接得到预测结果。并且 YOLO 中没有锚点框 (Anchor) 的概念，引入了 grid 来做区域性划分达到分而治之的目的，但是子区域 (grid) 的划分过大这就导致 YOLO 很容易丢失小目标，导致最后的检测精度不足，好处是大幅度提高了检测速度，该算法的增强版本在 GPU 上速度为 45 帧/秒，快速版本速度为 155 帧/秒 (640×480 像素)[20]。

WeiLiu 等人于 2015 年提出 SSD[28]。SSD 算法在 YOLO 速度快和 Faster RCNN 的基础上做了进一步改进。主要贡献有三点:1. 在单阶段检测算法中首次引入了锚点框 (Anchor) 的概念；2. 实现了在不同尺度和深度的特征图上做检测，这让不同大小的物体在适合的特征层上做检测，提高了检测效率；3. 采用了难样本挖掘的策略一定程度上解决了单阶段检测算法中的正负例样本数量不平衡问题。在 VOC2007 上取得了接近 Faster RCNN 的准确率 (mAP=72%)，同时保持了极快的检测速度 (58 帧/秒，640×480 像素)。在 Tsung-YiLin 提出 RetinaNet[29] 之前，单阶段检测器虽然在速度上优于两阶段检测器但是性能一直低于同期的双阶段检测器。Tsung-YiLin 等人分析了原因认为由于单阶段算法没有候选框提取这个过程，没有经过初步筛选的样本全部参与到分类和回归任务中导致正负例样本比例不平衡，单个简单负样本虽然产生的梯度较小，但是由于数量巨大在梯度中起了主导地位给网络学习带来负面影响。为了解决这个问题，RetinaNet 中提出了 FocalLoss，通过两个参数的限制降低网络训练过程中简单负样本的学习

权重，使网络更专注于其他样本的学习，取得了和两阶段检测器类似的性能。

近两年来，不基于锚点框的方法（anchor-free）作为目标检测算法的新思路，已经得到越来越多的关注。在典型的基于锚点框的算法中，模型的效果往往受限于 anchor 的配置参数，如 anchor 大小、正负样本采样、anchor 的宽高比等，要求开发者要十分了解数据，才能配置好 anchor 参数，从而训练得到一个好的模型。anchor-free 的算法无需配置 anchor 参数，即可训练得到一个好的检测模型，减少了训练前对数据复杂的分析过程。

anchor-free 算法又可分为基于 anchor-point 的算法和基于 key-point 的算法。anchor-point 算法本质上和 anchor-based 算法相似，通过预测目标中心点 (x,y) 及边框距中心点的距离 (w,h) 来检测目标，典型的此类算法有 FSAF[30]，FCOS[12] 等；而 key-point 方法是通过检测目标的边界点（如：角点），再将边界点配对组合成目标的检测框，此类算法包括 CornerNet[31], RepPoints[32] 等。RepPoints 使用一系列采样点取代了 bounding box 来提取目标特征，并进行分类和根据采样点生成矩形伪框来实现定位，通过采样点来选择目标特征相对 bounding box 更具代表性，更精确。

2.2 用于目标检测的数据集

目标检测是计算机视觉中的基础任务，主要解决的是图片中的目标分类与坐标回归。目标检测在实际应用中有着丰富的场景，针对不同的场景问题需求有不同的数据集已经被发表出来。按照任务的类型主要有以下:1. 通用目标检测如 PASCAL VOC[33]、MS COCO[9]、LVIS[34] 等；2. 人脸检测如 WIDER FACE[35]；3. 行人检测如 Caltech USA[36], CityPerson[37]；4. 弱小目标检测如 TinyPerson[3]。各种任务的代表场景如图。图2.1为不同种类的数据集的典型场景示例 (a) 弱小人体检测——TinyPerson；(b) 行人检测——Citypersons；(c) 通用目标检测——MS COCO；(d) 人脸检测——WIDER FACE。

通用目标检测最早的代表数据集是 PASCAL VOC[33] 数据集 (简称 VOC) 有 VOC2007 和 VOC2012 两种版本，在其上举办的比赛对目标检测领域的发展起到了极大的推动作用。数据集总共分为 4 个大类 vehicle，household，animal，person，总共 20 个小类，两个数据集训练和验证部分总计有 16551 张图片 40058 个标注框，测试部分有 16492 张图片和 39482 个标注框，总计有 33043 张图片和 79540

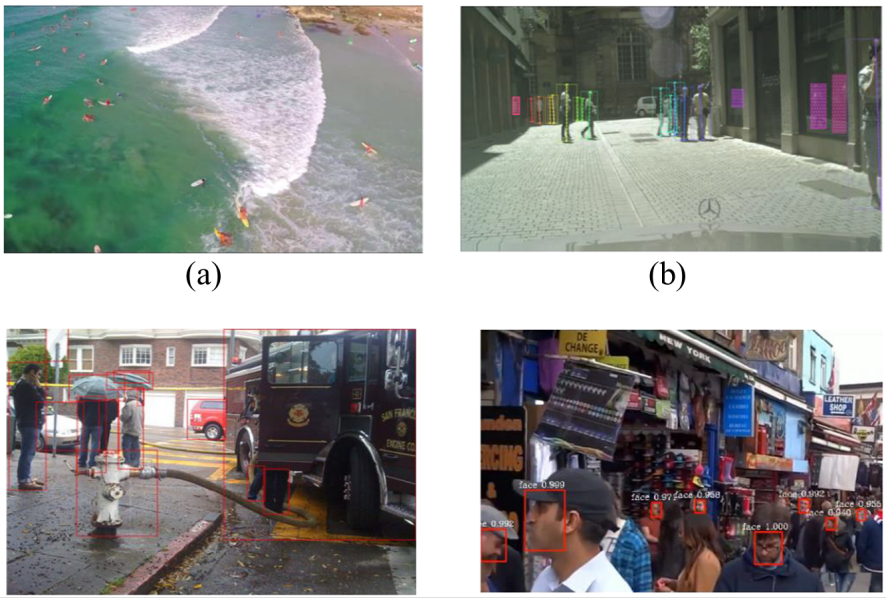


图 2.1 不同数据集的典型图例

Figure 2.1 Examples of different datasets

个标注。

微软团队在 2014 年发布了 MSCOCO[9] 其全称是 MicrosoftCommonObjects inContext，这是一个跨尺度数据集。其标注信息十分丰富，可以用来进行目标检测、关键点检测、语义分割、字幕生成等任务。MSCOCO 数据集中的图像分为训练、验证和测试集。由于巨大的数据体量、丰富的类别和完整的标注信息，是通用目标检测中最具有影响力的数据集。该数据集包含 33 万张图像、150 万目标实例、80 个目标类、91 个物品类以及 25 万目标关键点。相较于 VOC，MS COCO 有着更细致的类别划分，更多的标注数量和更大的尺度覆盖，是目标检测领域目前最有影响力的数据集。

FAIR 开放了 LVIS[34]，一个大规模细粒度词汇集标记数据集，包含了 164k 图像，并针对超过 1000 类物体进行了约 200 万个高质量的实例分割标注。由于细致的分类标注，所以这个数据集的物体类别数量具有天然的长尾分布特性 (大部分类别物体数量丰富，但一些类别的物体数量非常少) 也满足目标检测的实际任务场景，即如何让网络有效地从小样本中学习，这也对检测任务提出更大的挑战。2019 年，旷视研究院发布了新的大型目标检测数据集 Objects365[38]，它拥有超过 600，000 个图像，365 个类别和超过 1000 万个高质量的边界框，进一步

提升了数据集的体量，对推动了目标检测领域发展。

行人检测在实际场景中应用广泛，一直是计算机视觉领域的热点应用方向。随着研究的深入，具有更大的容量、更丰富的场景和更好注释的行人检测数据集相继被发表出来，如 INRIA[15]、ETH[39]、Daimler[40]、Caltech USA[36]，KITTI[41] 和 CityPersons[37]，同时这也代表了对泛化能力更好的算法和更高的性能的追求。Caltech USA 是加州理工学院发布的行人检测数据集，来自在城市环境中正常行驶的车辆拍摄的视频的切帧图片，视频大约 250，000 帧 (137 分钟左右)，共计

350，000 个标注框和 2300 个被标注的独立行人。张珊珊在 Cityscapes[42] 数据集上建立了 CityPersons 数据集，在 5000 张图像上标注了 35000 个行人，13000 个忽略区域，同时遮挡场景进行了很好的标注，而且图像在不同的季节采集于德国 27 个不同的城市，保证了图像场景的丰富程度。

WIDER FACE[35] 数据集是人脸检测的一个 benchmark 数据集，由香港中文大学选择了 61 个事件类别，共包含 32203 图像，以及 393，703 个标注人脸，其中，158，989 个标注人脸位于训练集，39，496 个位于验证集，检测难度划分为 Easy，Medium，Hard。WIDER FACE 场景丰富，涉及了人脸在不同的尺度，姿态，光照，表情，妆容，遮挡条件下的情况。

作者作为参与者构建了弱小目标检测代表数据集 TinyPerson[3]，TinyPerson 发表于 2019 年，共计 1610 张图片，72651 个标注框，图片由无人机在远距离广视角下拍摄，因此图片中的目标天然具有绝对尺度小这个特点，整个数据集的目标平均绝对大小为 18 个像素，这也是弱小目标检测的最大特点。数据集还对标注类别进行了细致的划分，包括位于陆地上的人，位于海里的人和不确定为人的区域，可满足不同任务的需要。

TinyNet[43] 涉及远程遥感目标检测。VisDrone[7] 数据集由无人机采集于中国 14 个不同城市的不同城市/郊区，保证了场景的多样性。整个数据集共包括 288 个视频片段，261908 帧和 10209 张图像，共计 260 多万个常用类别的标注框。从数据的目标平均大小来看，遥感目标检测和人脸检测都与弱小目标检测相近，但是在其他方面仍存在差异。以 WIDER FACE 为例，WIDER FACE 的数据集目标平均大小为 32.8 像素，接近于弱小目标的标准的 20 像素，但是人脸检测的目标长宽比固定而且人脸检测问题可以利用上下文信息来提高识别精度，但

是对于视角多样，信息量弱，长宽比不一的弱小目标检测数据集，这些信息便无法利用。遥感目标检测在尺度上更接近于弱小目标检测，但是遥感图像中的目标多为舰船、飞机等目标，具有特定的长宽比和密集排列等特点，这些也是不同于弱小目标检测的地方。表2.1[3] 为各种典型数据集的统计特征对比，其中正负号前边为数据的平均数，后边为数据的标准差。

表 2.1 典型数据集统计特性 [3]

Table 2.1 Statistical characteristics of typical datasets

数据集	绝对大小	相对大小	长宽比
TinyPerson	18.0±17.4	0.012±0.010	0.676±0.416
COCO	99.5±107.5	0.190±0.203	1.214±1.339
WIDER FACE	32.8±52.7	0.036±0.052	0.801±0.168
Citypersons	79.8±67.5	0.055±0.046	0.410±0.008

2.3　小目标检测

2.3.1　基于多尺度的小目标检测

针对跨尺度检测中的小目标的大多数研究都是为了获取尺度不变性时，发现小目标性能很差，再对小目标进行处理，因此绝大多数关于小目标检测的研究都是将小目标检测作为通用目标检测中一个附带的子问题来进行研究。已有许多学者对小目标的检测也进行了广泛的研究。SNIP[44] 和 SNIPER[45] 利用图像金字塔同时使用尺度正则化策略来保证目标的大小在一个固定范围内，将小目标放大提升检测精度。SNIPER 采用区域抽样的方法进一步提高训练效率。

小目标性能精度比较低的重要原因是小目标自身尺度小，信息量少，细节特征不够多。超分辨率 (Super Resolution) 常用于恢复低分辨率目标的信息，用网络的学习能力自动补充小目标缺乏的细节特征，提高其分辨率，因此一些工作已经将其引入到小目标检测中。NohJ[46] 认为小目标检测在 ROI 特征提取过程中容易失真，因此提出了一种利用高分辨率目标特征作为监督信号的超分辨率方法，通过用相对容易被网络学习的大尺度目标引导小尺度目标的学习，并且在小尺度目标上取得了性能提升。Chen[47] 等人分析了 COCO 中不同尺度目标所占损失的比重，发现小目标贡献的损失并不主导网络的学习，因此提出了一种反馈驱

动的数据提供策略，将包含大目标的图片拼接产生小目标，利用损失统计信息来平衡小目标检测的损失。

三叉戟网络 (TridentNet)[1] 中分析了影响检测器性能的三个因素: 网络深度、网络下采样倍率、感受野大小，所提网络结构如图2.2[1]，构建了不同感受野的平行多分支，并生成了更具辨别力的小目标特征，同时采用不同分支共享参数的办法降低参数量，并且让不同大小的目标适应于不同的感受野分支学习，提高了小目标检测的性能。IPG-Net[48] 提出了浅层特征富含空间和细节特征但是缺少高级的语义信息，深层有语义信息但缺少空间特征，这种信息不平衡是阻碍性能提升的原因。因此作者提出了图像金字塔引导网络，通过图像金字塔信息转化模块和图像金字塔融合引导模块解决信息不平衡问题从而减轻了小目标特征的信息丢失。这些上述方法在一定程度上提高了小目标检测的性能，但都权衡了其他尺度的性能变化，保证了检测器在各尺度目标的检测精度。

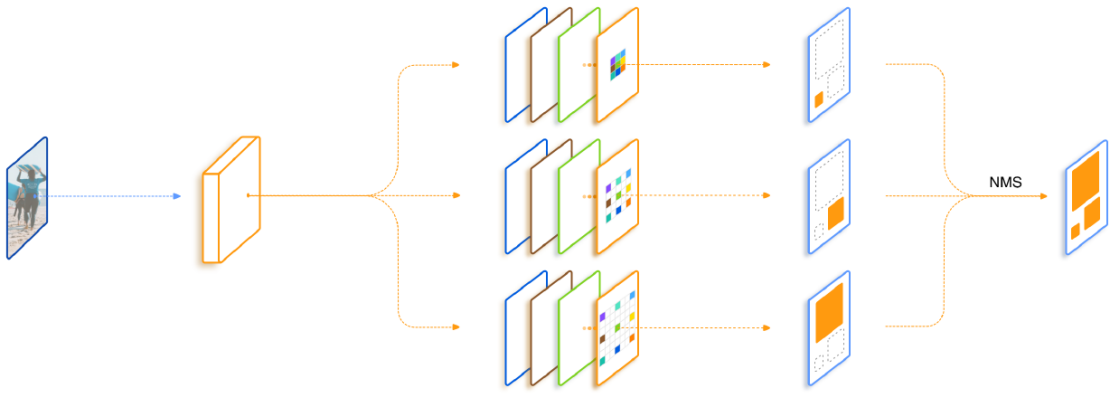


图 2.2 三叉戟网络结构示意图 [1]

Figure 2.2 Schematic diagram of TridentNet structure

2.3.2 基于单尺度的小目标检测

基于单尺度的小目标检测即弱小目标检测，因为弱小目标数据集的物体尺度分布的平均大小集中于 20 像素左右，整体数据集为单一小尺度的数据集，因此设计算法的时候只考虑提升小目标的检测精度即可。扩展特征金字塔 (EFPN)[2] 的作者认为小尺度目标和中尺度目标都集中于 FPN 中的浅层特征层，对小目标检测不利，因此利用超分辨率的思想构造了一个具有更多几何细节的特征层，如图2.3[2]，通过特征纹理转移模块将适合做小目标检测的特征充分融合，并进一

步接受来自更浅层骨干网络的特征信息组成适合小目标检测的特征层。

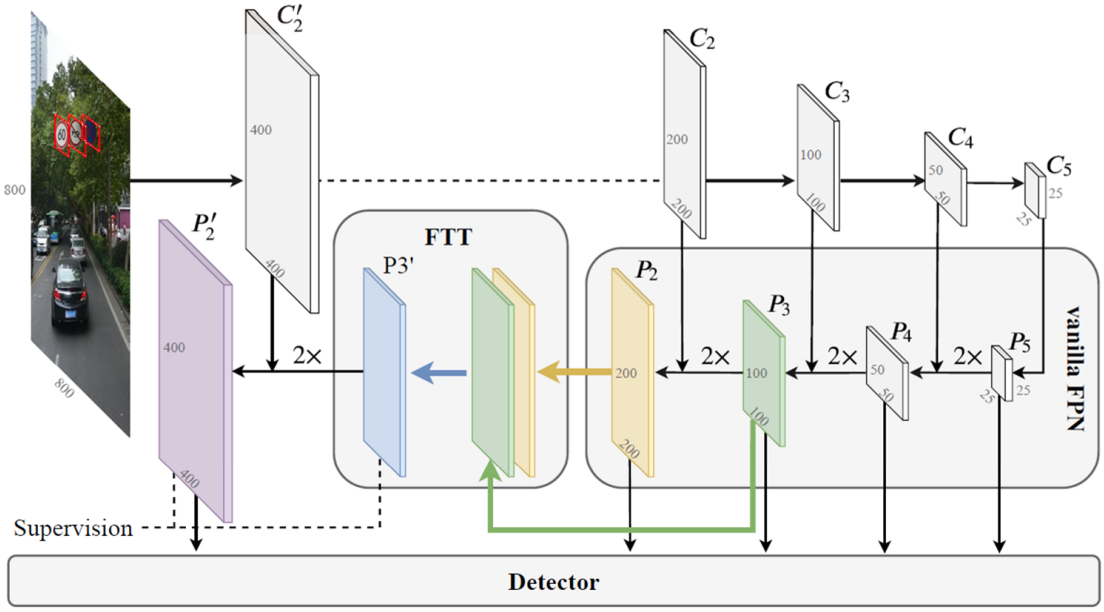


图 2.3 扩展特征金字塔结构示意图 [2]

Figure 2.3 Schematic diagram of extended feature pyramid structure

在 TinyBenchmark[3] 中，提出了一个迁移预训练数据集的尺度分布向目标数据集的尺度分布接近的方法来提升性能，从数据集预训练的角度为弱小目标检测提供了一种新的研究思路。作者在实验中发现，用于网络的权重初始化的预训练数据集和用于检测器训练的目标数据集之间的尺度分布的不匹配会弱化网络的特征表示能力和降低检测器检测精度。因此，作者提出了一种简单而有效的尺度匹配方法，将两个数据集之间的尺度分布对齐，在预训练的过程中让网络学习到目标数据集的尺度分布特性的相关知识，以获得对有利于弱小目标特征表示能力。实验结果证明基于尺度匹配的算法在不同的检测器有显著的性能提高。具体过程如下图 [3] 所示:

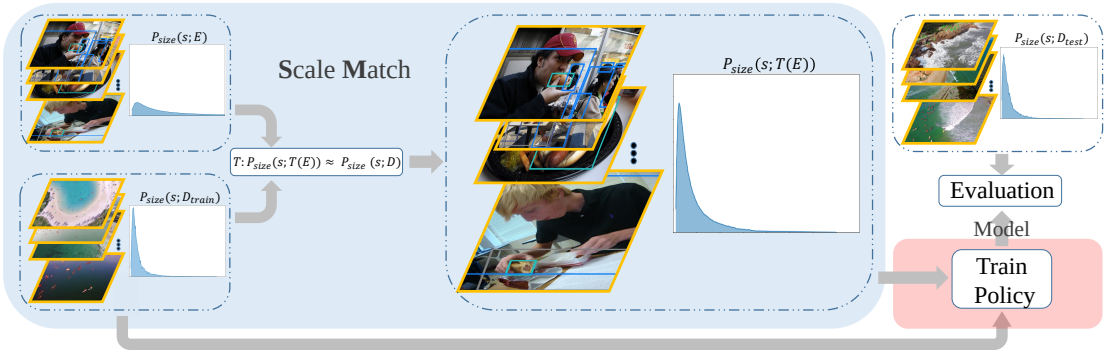


图 2.4 尺度匹配算法流程示意图 [3]

Figure 2.4 Schematic diagram of scale matching algorithm

2.4 动态神经网络

深度神经网络已经在计算机视觉、自然语言处理等领域取得了较大的成功。这些年来人们不断见证越来越强大、高效的神经网络模型设计。然而，大多数当前流行的深度网络都具有相同的静态推理范式：一旦完成训练，网络的结构与参数在测试阶段都保持不变，这在一定程度上限制了模型的表征能力、推理效率和可解释性。动态网络则可以在推理阶段根据输入样本自适应地调节自身的结构/参数，从而拥有诸多静态网络无法享有的良好特性。

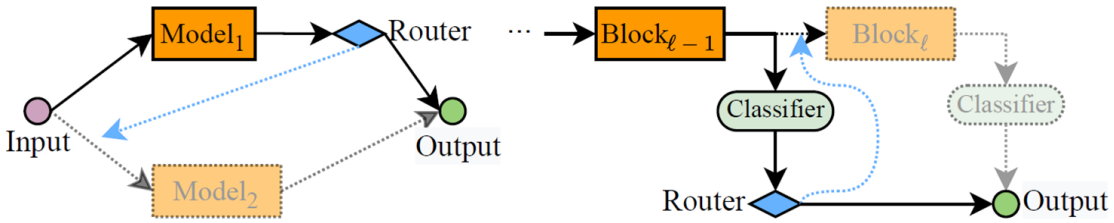
实际上，动态网络的核心思想——自适应推理，在如今的深度网络流行之前已经被部分研究者研究。本节将主要从动态结构和分辨率积动态网络两个方面进行对动态神经网络的介绍。

2.4.1 动态结构网络

流行的深度网络结构往往包括深度（网络层数）和宽度（通道数、并列的子网络个数等）这两个维度。因此，具有动态结构的网络又包括动态深度、动态宽度两个类型。

由于几乎所有的网络都由多个网络层堆叠而成，一个比较自然的实现动态结构的思路就是针对不同样本，选择性地执行不同的网络层。具体地，实现动态深度又主要包含两类思路：“早退”机制和“跳层”机制。

“早退”机制的核心思想是，在模型中间层设置出口，并根据每个样本在这些中间出口处的输出，自适应地决定该样本是否“早退”。不同的工作设计了不同的网络结构来实现这样的“早退”，包括将多个模型串联 [49]、在单个 backbone



(a) 模型串联　　(b) 在中间层添加分类器

图 2.5 早退机制的两种基本实现思路 [4]

Figure 2.5 The early-existing scheme

中间层添加分类器 [50] 等（如图2.5所示）。值得一提的是，有研究表明如果在链式结构的 CNN 中添加中间出口，则这些出口会干扰彼此的性能 [51]。为了解决这一问题，多尺度密集连接网络 (MSDNet，见图2.6) 采用了多尺度特征以及密集连接 (dense connections)，有效地提升了多个分类器的协同训练效果。

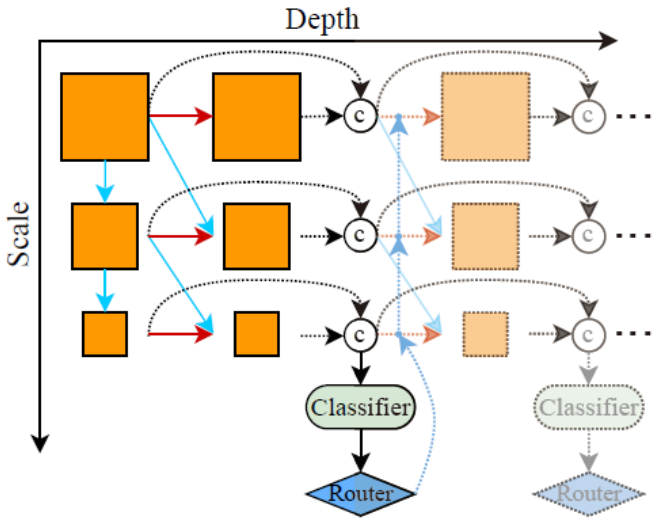


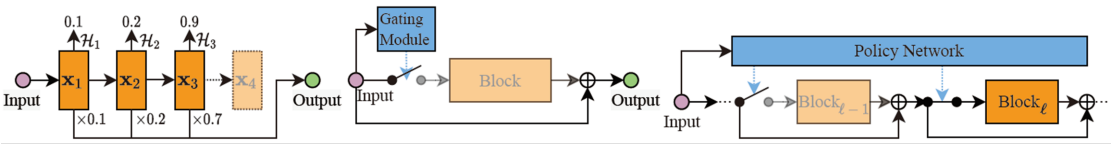
图 2.6 MSDNet 基本结构 [4]

Figure 2.6 Multi-scale DenseNet

“早退”机制相当于跳过了某一分类器之后所有层的运算。第二类具有动态深度的网络则更加灵活：针对每个输入样本，自适应地决定网络的每个中间层是否执行。根据决策方式，动态“跳层”主要有三种实现（如图2.7所示）：基于 halting score[52]，基于门函数 [53] 和基于策略网络 [54]。

而关于动态宽度的网络，现如今常用的方法为多专家混合系统（MoE）以及卷积神经网络中的动态通道剪枝。

有一类工作通过并行结构建立多个“专家”(可以是完整模型或者网络模块)，

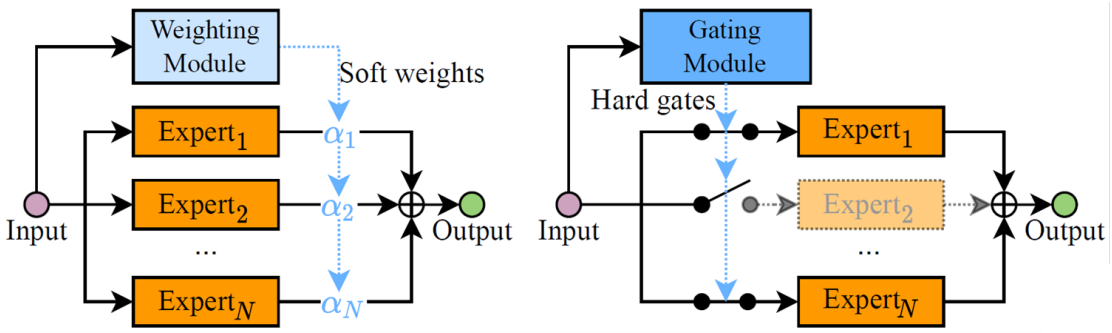


(a) 基于halting score (b) 基于门函数 (c) 基于策略网络

图 2.7 动态跳层的几种实现方式 [4]

Figure 2.7 Dynamic layer skipping

并对这些“专家”的输出结果进行动态加权来得到最终预测。早期的相关工作往往对这些结果进行“软”加权（如图2.8(a) 所示）来提升模型的表征能力 [55, 56]。然而，网络的运算量随着“专家”个数线性增长，带来了大量的冗余计算。因此，近期的一些研究通过控制门（gates）来选择性地激活这些“专家”子网络，从而实现模型效率的提升（见图2.8(b)）。



(a) 以软权重动态加权 (b) 选择性激活MoE分支

图 2.8 MoE 结构 [4]

Figure 2.8 MoE structure

MoE 结构可以被运用于多种网络中，包括 CNN，LSTM 和 Transformer。

卷积神经网络中的动态通道剪枝这一课题在近年来被广泛研究。相较于静态剪枝方法将某些“不重要”的通道永久性的去除，此类方法可以根据样本自适应地激活不同的卷积通道，从而在保持模型容量的情况下实现计算效率的提升。文章将此方向的研究分为三类：通道维度的多阶段结构 [57]，基于门函数的动态剪枝 [58]，以及直接基于特征激活的动态剪枝 [59]。

类似于动态跳层机制，由于门函数的“即插即用”特性，其在 CNN 的动态剪枝中也是当前较为流行的主要方法。不同的工作采用了不同的门函数设计，以及训练方法等。更多细节的讨论这里不再赘述。

值得一提的是，已经有工作利用门函数同时控制网络的深度和宽度 [60]，[61]。这些方法通常先决定某个网络层是否执行，若执行，则进一步对该层的不同通道进行更细粒度的挑选。

2.4.2　分辨率动态网络

分辨率动态网络将每张输入样本作为整体处理，为了减少“简单”样本的高分辨率表征带来的冗余计算，对不同的输入图像采用动态分辨率进行数据表示。现有工作中，动态分辨率主要有两种实现思路：

第一种是动态的放缩比例。DRNet[62] 动态地调整了输入到分类器中的图像大小，具体分辨率是通过一个分辨率预测器来实现的。

第二种是采用多尺度架构。分辨率自适应网络 (resolution adaptive network)[63] 建立如图2.9所示的多尺度架构，用不同的子网络处理不同分辨率的特征，并将这些子网络从小到大顺序执行。通过允许“早退”，“简单”的图像样本可以以较低的分辨率被处理，从而避免调用处理更大分辨率特征的子网络。

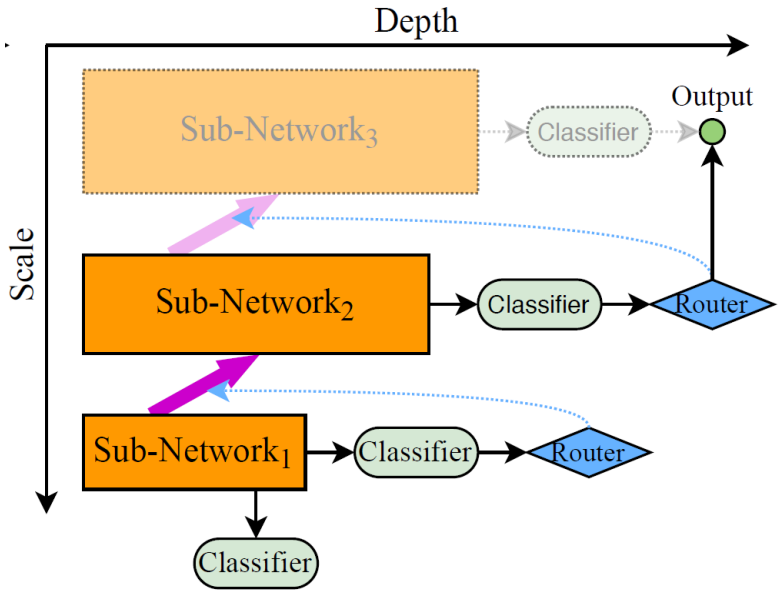


图 2.9 分辨率自适应网络 [4]

Figure 2.9 Resolution Adaptive Network

上文所述的动态结构网络和动态分辨率网络均在解决分类任务时提出，本文设计的动态池化网络主要用于解决目标检测任务，是动态网络适用领域的一个拓展和迁移。此外，动态池化网络可被视为一种在卷积步长层面动态的方法，是对现有的动态网络的一种补充。

2.5 本章小结

本章从四个方面详细地阐述了与本研究相关的国内外本学科领域的发展现状与趋势，分别是检测算法的发展历程、检测相关的数据集、小目标检测、动态神经网络。检测算法方面主要介绍了主流检测算法的原理和发展历程；数据集方面介绍了用于不同目标检测任务的数据集并进行尺度特点的比对；小目标检测介绍了多尺度和单尺度下的两种小目标检测；动态神经网络介绍了动态结构网络和分辨率动态网络。下一章将介绍研究内容与方法。

第 3 章　基于动态神经网络的弱小目标检测方法

3.1　动态池化网络的研究背景及意义

当前常用的检测算法除去输入图像和 NMS 后处理流程之外，还有一个重要的部分就是检测器网络。检测器网络一般由三个部分构成：骨干网络、颈部网络和头网络。骨干网络是目标检测任务的基本特征提取器，其主要任务是将图像作为输入并输出相应输入图像的特征图。大多数用于检测的骨干网络是用于分类任务的网络，这些网络将最后的全连接层替换为后续的颈部网络和头网络。

对于精度与效率的不同要求，人们可以选择不同深度和连接密集程度的骨干网络，如 ResNet[5]、ResNeXt[64] 或轻量级骨干网络 MobileNet[65]、MobileNetV2[66] 等。

表 3.1 ResNet50 的结构 [5]

Table 3.1 Architecture of ResNet50

layer name	output size	layer
conv1	112 ×112	7 ×7, 64, stride 2
conv2_x	56 ×56	3 ×3 max pool, stride 2
		$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28 ×28	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$
conv4_x	14 ×14	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$
conv5_x	7 ×7	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$

不同的骨干网络都可以被大致分成不同的阶段（stage），每个阶段由卷积层或若干块（block）组成。以 ResNet50 为例，其结构如表3.1所示，每个阶段中的块由不同的卷积核构成，若卷积核的步长为 2，那么输入的特征图将在此阶段完

成降采样操作（降采样因子等于 0.5）。

当骨干网络的深度、宽度变化时，对应的特征提取能力也会相应地变化。但本文选择研究网络中的降采样因子选择，而不是对可变结构进行研究，是基于目标检测任务的特性而决定的。在目标检测任务尤其是小目标检测任务中，检测器网络对图像中的目标大小极为敏感，也就是说，目标检测任务是一个尺度敏感的任务，其性能与图像/目标尺度高度相关，所以研究降采样因子可变的动态神经网络比研究深度/宽度可变的动态神经网络对于目标检测任务来说更具有意义。

3.2 动态池化网络

为了利用放大图像的优势，检测器将以统一放大的图像作为输入，而在检测器的骨干网络中进行自适应降采样操作以消除冗余，同时保留应有的信息。该网络被命名为动态池化网络（Dynamic Pooling Network，DPNet），DPNet 的推理流程如图3.1所示。

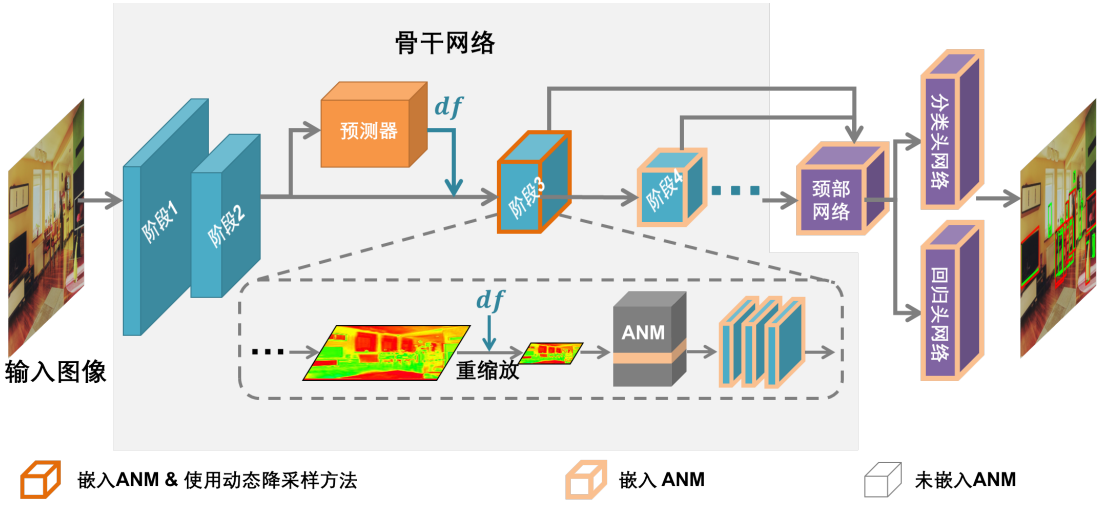


图 3.1 动态池化网络框架图

Figure 3.1 Framework of DPNet

从图3.1可以看出，DPNet 主要包含两个部分，第一个是常见的基于卷积神经网络的检测器，例如 Reppoints[32]，另一个是降采样因子预测器，该预测器是为了找到合理且最小的降采样因子来调整特征图的分辨率，以达到检测性能和效率的权衡。以 ResNet 为例，骨干网络的每个阶段（stage）都会有一个降采样操作。对于任意的特征图，本文首先用预测器对其进行合适的降采样因子的预测，然后在骨干网络的下一个阶段，特征图将会用此降采样因子进行缩小。当降采样

因子比原本的降采样因子（卷积步长为 2 等价于降采样因子为 0.5）小时，计算量（FLOPs）将会明显减少。

降采样因子控制特征图分辨率大小的具体实现如图3.2所示，骨干网络中需要用到降采样因子的阶段将把卷积步长为 2 先设置为 1，再利用降采样因子进行重缩放操作，这样就可以将网络从固定降采样倍率的操作转变为更灵活的降采样方式，以实现各种倍率的降采样。

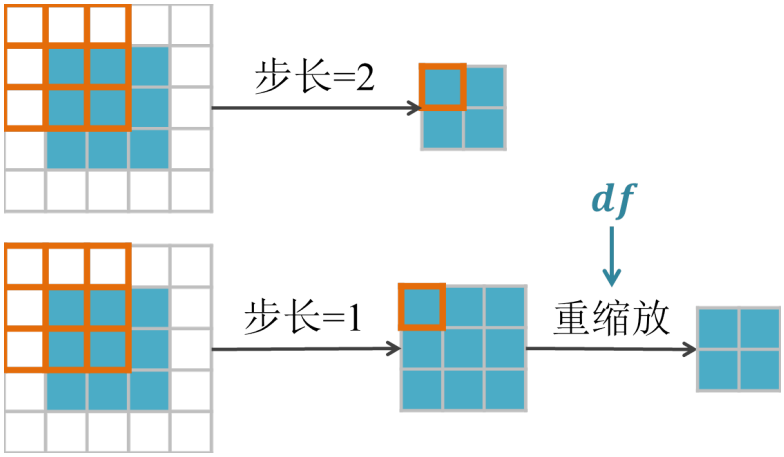


图 3.2 降采样因子的作用机制

Figure 3.2 The mechanism of down-sampling factor

3.2.1　自适应标准化模块

考虑到计算资源的限制，DPNet 将在训练中采用不同的降采样因子但共享参数的检测器。这种训练方式在文中被称为混合降采样因子训练（Mixed Scale Training），其具体训练流程将在3.3.1做出详细阐述。经过混合降采样因子训练的检测器将拥有如下特点：可以仅在储存一个模型的条件下完成不同降采样因子的推理。这样的模型可以在不同设备条件下调整为不同的推理计算量。

而混合降采样因子训练过程中，即便对于同一张输入图片，网络同一阶段产生的特征图大小也不是固定的。为了解决不同降采样操作后的特征图由于尺度不同而分布差异过大的问题，本文设计了自适应标准化模块（Adaptive Normalization Module, ANM）嵌入到 DPNet 中。

标准化方法（normalization）在解决深度神经网络学习中的内部协方差变化问题中起着重要作用 [67]，常用的基于卷积神经网络的检测器也离不开标准化层。标准化可以将条件信息编码为特征表示 [68, 69]，它还可以通过重中心化和

重缩放来对输入特征进行标准化操作，以加速收敛，提高稳定性 [70]：

$$y' = \gamma \frac{y - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta, \tag{3.1}$$

其中，y 是要标准化的输入，y' 是输出 γ，β 是可学习的量表和偏差，μ，σ^2 是输入的均值和方差。检测器中最常用的两种归一化方法是批标准化（Batch Normalization, BN）和组标准化（Group Normalization, GN）。批标准化方法在公式（3.1）中的 μ，σ^2 是训练期间当前批的平均值和方差。在测试过程中，取而代之的是所有训练图像的均值和方差的滑动平均统计。组标准化方法则将通道分组，并对分组内的特征进行归一化，在公式（3.1）中的 μ，σ^2 是分组的平均值和方差。与批标准化不同，组标准化不利用批维度，其计算与批大小无关。图3.3展示了两种标准化的归一化维度，其中，**C** 表示通道，**N** 表示批大小（batch size），**H, W** 表示图像高 **H** 和宽 **W** 拉长成一维，即该维度大小为 **H** ×**W**。

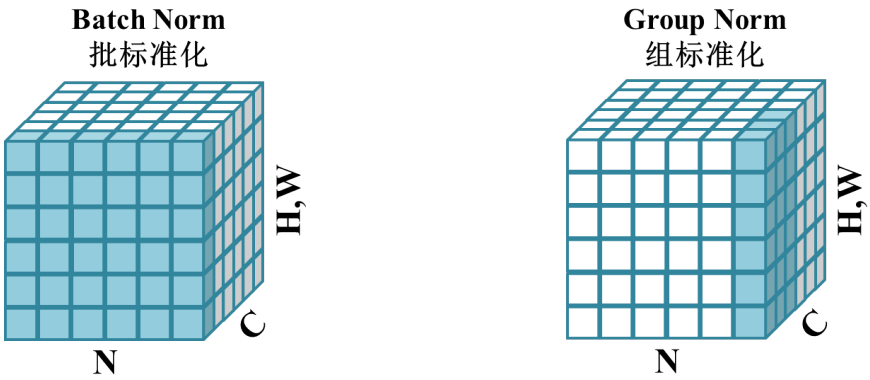


图 3.3 标准化方法

Figure 3.3 Normalization methods

要对网络进行混合降采样因子训练，自适应标准化模块将一个统一的检测器网络中每个降采样因子对应的分支的所有标准化层进行了独立化操作。该算法通过对测试过程中的不同的降采样操作处理过的特征特征均值和方差进行分开标准化，解决了不同分支之间的特征聚合不一致问题。自适应标准化模块中的比例和偏差可能能够编码当前分支所对应的降采样因子的条件信息。

此外，与增量学习相比，使用自适应标准化模块，网络可以以不同的降采样因子统一训练所有分支；因此，所有权重都会联合更新，以实现更好的性能。

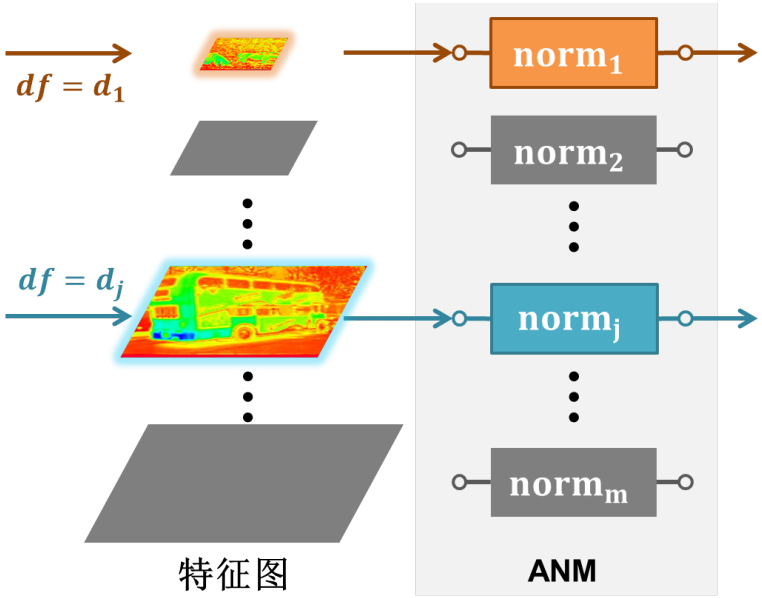


图 3.4 ANM 的作用机制

Figure 3.4 The mechanism of the ANM

如图 3.4所示，自适应标准化模块对每个降采样因子的标准化进行解耦，并选择相应的标准化层来标准化特征：

$$y' = \gamma_j \frac{y - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}} + \beta_j, j \in \{1, 2, \ldots, m\}, \tag{3.2}$$

其中 ϵ 是使数值稳定的一个小数值，μ_j 和 σ_j^2 是每个降采样因子下才单独激活统计的均值和方差，β_j 和 γ_j 是针对于不同降采样因子独立的可学习的偏置值和缩放值。不同降采样因子缩放后的不同特征图将被输入 ANM，ANM 将切换到降采样因子对应的标准化层。

3.2.2　降采样因子预测器

为了选择适合于特征图的降采样因子，本文设计的方法在特征输入到骨干网络的下一阶段之前插入了一个降采样因子预测器（Down-sampling Factor Predictor, DFP）网络，该网络可以被视为一个轻量级分类器。预测网络以特征图作为输入，输出各个降采样因子的概率，最终选择概率最大的降采样因子作为特征图在下一个阶段的缩放比例，所以预测降采样因子的任务可以被视为一个分类任务。

本文的设计将预测器插入到了骨干网络的中间（以 ResNet 为例，预测器被插入到了 Stage2 之后，Stage3 之前），而不是在输入图像之前或骨干网络里更偏后的位置。之所以插入到这一位置，是因为考虑到既要充分利用部分骨干网络提取特征的能力，又要保证减少的计算量足够。而骨干网络的前两个阶段具有提取特征的能力，预测器可以利用骨干网络的这部分进行特征提取而不引入大量新参数和计算量。骨干网络后几个阶段的输出特征图会连接到 FPN，特征图大小会影响颈部网络和头网络的计算量，自适应降采样操作在该部分之前插入能够极大程度上减少计算量。

DFP 的目标是预测一个合适的降采样因子，它既能尽可能减少计算消耗，又能保持较高的检测性能。需要注意的是，理论上来说，降采样因子候选值可以从 0 到 1 的任意值，但如果把预测对象设置为此范围内的某一值，降采样因子预测器将很难在这样一个连续的范围里面探索到十分具体的取值，这样的预测任务较为耗时。作为实际需求下的一种简化策略，本文选择 m 个离散的降采样因子 $\{d_1, d_2, ..., d_m\}$ 来缩小探索范围，其中 d_m 表示通用探测器中的默认降采样因子（以 ResNet50 为例，Stage3 中的降采样因子等效于 0.5，即 $d_m = 0.5$）。降采样因子预测器 $DFP(\cdot)$ 的前传函数可表示如下：

$$DFP(M) = p_s = [p_{d_1}, p_{d_2}, \dots, p_{d_m}], \tag{3.3}$$

其中 M 是输入到 DFP 的特征图，$p_s \in \mathbb{R}^M$ 是 DFP 的输出，代表每个候选降采样因子的概率。然后，选择对应于最高概率 p_{d_j} 的因子 d_j 作为输入到骨干网下一阶段的降采样因子。

由于 DFP 带来了额外的计算量，所以必须将预测器网络的模型大小尽可能地保持在一个较小的水平。如果引入的额外计算量甚至超过了从较小的降采样因子中节省的计算量，那么实现这样一个模块是不会带来优势的。故本文在设计 DFP 时参考并简化了学界公认实用的 ResNet 的结构，网络分为不同的阶段，每个阶段均采用 ResNet 中的基本块和残差结构连接方式，但每个阶段只用一个基本块，简化了 ResNet 每一阶段的结构。

DFP 主要被视为分类器，但本文为其另添加了一个分支，用于预测输入图像中实例的各类特性，这是因为输入图像中实例的大小对图片的降采样倍率也有影响。具体来说，若输入图像中的目标均为小目标，那么检测器在高分辨率的

特征图上应该能获得更好的检测结果。故当 DFP 对图像中实例的大小具有感知能力时，能更好地提升它选择降采样因子的能力。另添加的分支称为统计分支（Statistic Branch）。统计分支 $Sta(\cdot)$ 将用于预测一个向量 $\hat{v}$，其中包含输入图像 I_i 中的实例数及其大小的统计值（例如平均值、最大值和最小值等）：

$$Sta(f) = \hat{v} = [\hat{amt}_i, \hat{mean}_i, \hat{max}_i, \hat{min}_i, \hat{median}_i], \tag{3.4}$$

其中，f 为输入到分支的特征向量，而 $\hat{amt}_i, \hat{mean}_i, \hat{max}_i, \hat{min}_i, \hat{median}_i$ 分别表示统计分支所预测的图像 I_i 中目标实例的数量，均值，最大值，最小值和中位数。

如图3.5所示，预测器的结构由 4 个（或更少数量的）基本块（ResNet 中的 Basic Block）组成。每个基本块的结构如图3.6所示，由两个卷积层组成，卷积层后接批标准化层和 ReLU 激活函数。

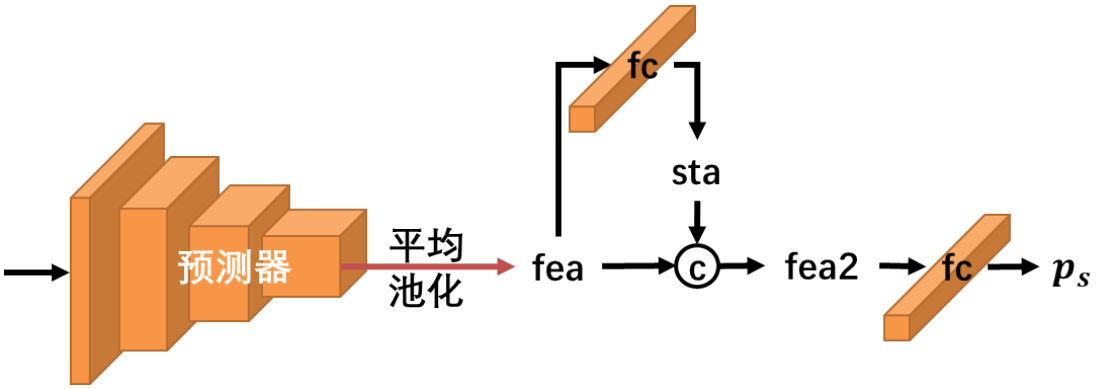


图 3.5 降采样因子预测器的结构

Figure 3.5 The structure of DFP

从图3.5可知，当特征图通过统计分支预测出图像中的实例特性时，这些特性将会和主支路提取的特征一起参加对降采样因子概率的映射，这一操作能够增强实例大小及数量信息和降采样因子的联系。

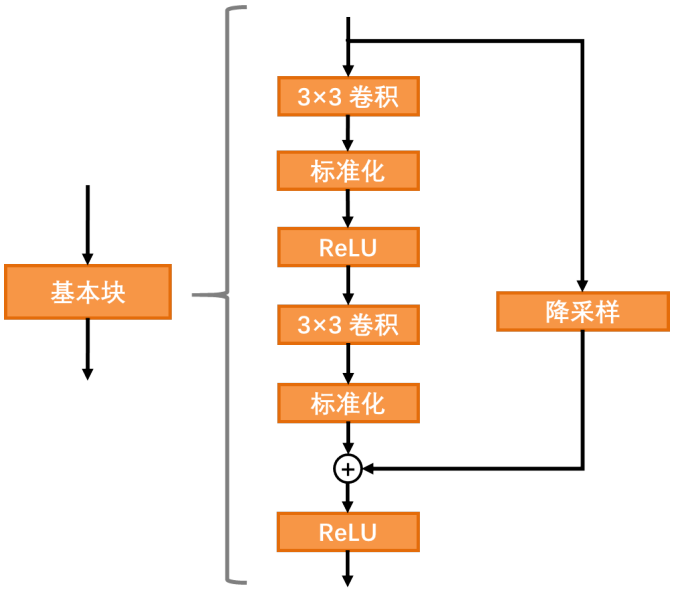


图 3.6 基本块的结构

Figure 3.6 The structure of basic block

3.3 优化目标

如前文所述，DPNet 由一个经典的检测器网络和嵌入其中的降采样因子预测器网络组成。为了更好地优化 DPNet，本文将其优化方法设计为按步优化的方式，具体的网络训练方式可见算法1。整个训练流程将遵循先训练检测器再训练降采样因子预测器的流程。

算法 1 DPNet 的训练

Input: 训练集 D_{train}

Output: 模型 *Net*

// 检测器 D 的训练步骤

初始化检测器 D 共享的各卷积层和全连接层的权重

初始化 df list（$\{d_1, d_2, ..., d_m\}$）中的各 d_j 对应的独立的标准化层的系数

for $i = 1, \ldots, n_{iters\_pretrain}$ **do**

    得到当前批次的数据 x 和标签 y

    清空权重的梯度 $optimizer.zero\_grad()$

    for d_j in $dflist$ **do**

        切换到当前的 d_j 所对应的标准化层分支

        对特征层进行比例因子为 d_j 的缩放

计算当前降采样因子 d_j 对应的损失 $\mathcal{L}_{MST_j} = \alpha_{cls} L_{cls_j} + \alpha_{reg} L_{reg_j}$
end for
计算该批次下的损失 $L_{MST} = \sum_{j=1}^{m} L_{MST_j}$
计算梯度, $\mathcal{L}_{MST}.backward()$
更行网络权重, $optimizer.step()$
end for
// 预测器 P 的训练步骤
初始化降采样因子预测器 P
冻结检测器 D 的参数
for $i = 1, \ldots, n_{iters\_pretrain}$ **do**
计算预测器的损失 $\mathcal{L}_P$
计算梯度, $\mathcal{L}_P.backward()$
更行网络权重, $optimizer.step()$
end for
return *Net*

3.3.1　混合降采样因子训练

DPNet 在训练检测器时，会将同一张图像输入到网络中用 m 个不同的降采样因子 $\{d_1, d_2, ..., d_m\}$ 分别前向传导 m 次，其中，每次的损失函数计算如下：

$$L_{MST_j} = \alpha_{cls} L_{cls_j} + \alpha_{reg} L_{reg_j}, \tag{3.5}$$

其中 L_{MST_j} 对应降采样因子 d_j 对应的损失值，由分类损失 L_{cls_j} 和回归损失 L_{reg_j} 加权求和而来。

每次迭代将会由网络在 m 个降采样因子下前向传导计算得到的损失求和反向传导更新：

$$L_{MST} = \sum_{j=1}^{m} L_{MST_j}, \tag{3.6}$$

这种训练方式被称为混合降采样因子训练（Mixed Scale Training，MST）。

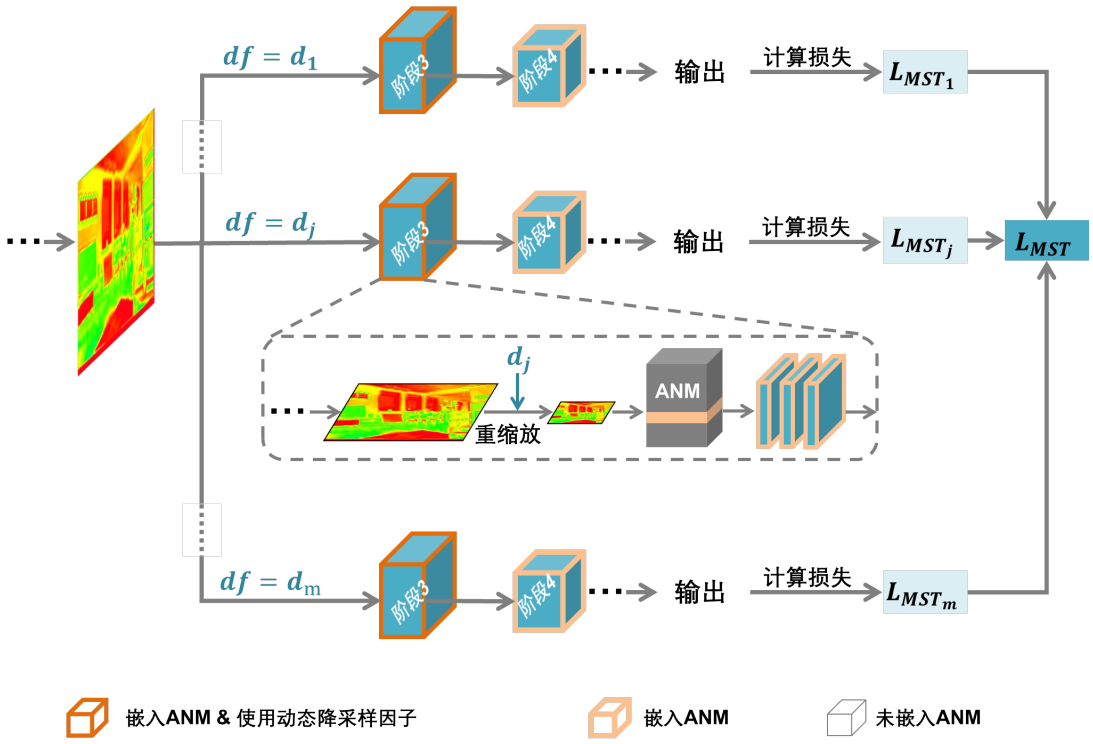


图 3.7 混合降采样因子训练的流程

Figure 3.7 The pipeline of Mixed Scale Training

这样的训练方式结合自适应标准化模块，可以让一个参数共享的单一检测器网络在不同的降采样因子下都有不错的检测性能，图3.7为混合降采样因子训练的具体实现。

3.3.2 降采样因子预测器训练

训练降采样因子预测器时，检测器网络的参数已固定，降采样因子预测器的监督可分为两个部分：一个部分来自于对统计分支的监督，另一个部分来自于对预测器分类结果的监督。

统计分支的预测目标 v 可直接由统计图像中目标实例的数量和统计值得到，统计分支的损失函数计算基于平滑 L1 损失实现，输入为预测的 $\hat{v}$ 和目标标签 v，如下所示：

$$\mathcal{L}_{sta} = SmoothL1(\hat{v}, v), \tag{3.7}$$

对统计分支的训练流程如图3.8所示，由于在统计分支上加了一个更为直接的监督，降采样因子预测器也将被训练得更有效。

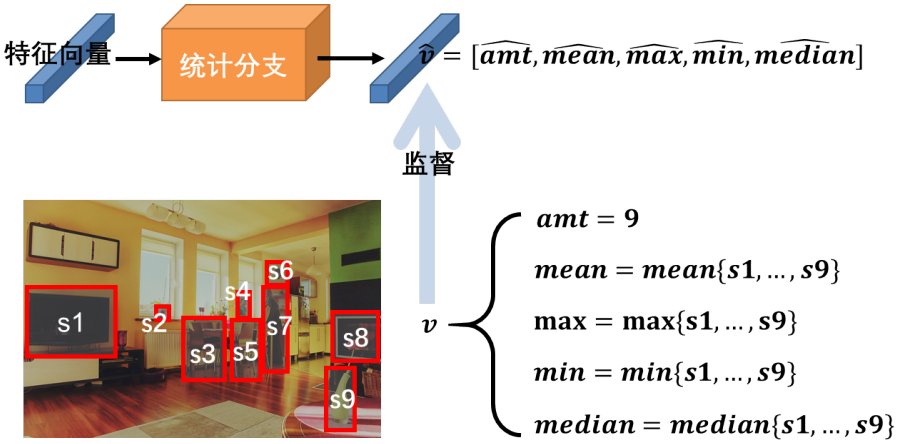


图 3.8 统计分支的训练

Figure 3.8 Training of Statistic Branch

而降采样因子预测器本质为一个分类器，输出预测概率中最大的降采样因子即为被选择的降采样因子。预测器的分类标签和检测器相关，简单来说，当图片采用不同的降采样因子输入到检测器中时，应当选择检测性能好的降采样因子，故可以将每张图片输入到检测器中获得降采样因子的标签，性能好则该降采样因子在预测器中的标签即为“1”，若性能相较于其他降采样因子得到的结果过差则标签为“0”。一个最有用的判定方式就是用检测器的 AP 性能来进行判定，若某降采样因子为图片在所有降采样因子缩放下 AP 性能最好的那一个，那么该降采样因子则为这张图片在预测器中对应的分类标签。

但是，如果用 AP 的方法对每张图片都取降采样因子的标签的话，耗时十分漫长，这是不现实的。所以只有通过直接计算损失更为快速，而损失函数的设计需要和 AP 尽可能地相关和接近，以至于本文可以直接通过这个损失值来判定输入图像在进入检测器后在某降采样因子下的检测性能好坏。因此，为了加快获得监督信息的时间，本文设计了一种名为监督损失的损失，它相较于检测器的原本损失而言，与 AP 的联系更紧密。如果每张图在检测器网络下都可以获得与 AP 相关的监督损失值，该值将被直接作为合适与否的评判标准：如果损失值越小，那么选择该降采样因子就更为合适。

监督损失的计算流程如算法2：

算法 2 监督损失的计算

Input: 训练集 D_{train} ，训练好的检测器 D

Output: 监督损失

for **do** I_i in D_{train}
  for d_j in $dflist$ **do**
    获得 $D(I_i)$ 中的结果框 $bboxes_i$
    对 $bboxes_i$ 进行正负例匹配得到 pos_i 和 neg_i
    $neg_i = sorted(NMS(neg_i))$
    $pos = []$
    for gt in I_i **do**
      pos_i 中与 gt 匹配的分数最高的框为 $pos*$， $pos \cup pos*$
    end for
    $k = len(pos)$
    $neg = top_k(neg_i)$
    仅计算 pos 与 neg 产生的损失，记为 $L_i^{(j)}$
  end for
end for
return 训练集中所有 $L_i^{(j)}$

监督损失的意义并不是用于优化检测器，而是用于获取降采样因子预测器的监督信息，它只是一个与 AP 由负相关联系的函数，而不是实际作为优化的损失函数。

得到监督信息后，即可对降采样因子预测器进行训练，预测器的损失采用常见的多标签损失函数计算：

$$\mathcal{L}_{df} = \sum_{j=1}^{m} -y_{d_j} \log\left(p_{d_j}\right) - \left(1 - y_{d_j}\right) \log\left(1 - p_{d_j}\right). \tag{3.8}$$

公式（3.8）中标签 y_{d_j} 的计算方式如下：

$$y_{d_j} = \begin{cases} 1, \dfrac{L^{(j)}}{L_{min}^{(j)}} \leq T, \\ 0, \dfrac{L^{(j)}}{L_{min}^{(j)}} > T, \end{cases} \quad j = 1, 2, ...m \tag{3.9}$$

其中，T 被默认设置为 1.1，$L^{(j)}$ 是降采样因子为 d_j 时计算的监督损失，$L_{min}^{(j)}$ 是所有 $L^{(j)}$ 中 $(j = 1, 2, ...m)$ 的最小值。当 $L^{(j)}$ 和 $L_{min}^{(j)}$ 的比值在设定的阈值范围内时，就判定降采样因子 d_j 也是一个合适的（正确的）降采样因子。

3.4　本章小结

本章主要介绍了研究的内容与方法，研究内容方面介绍了典型骨干网络中降采样操作的实现方式和缺点，还阐述了研究降采样因子可变的动态池化网络的意义。研究方法方面先是总体概述了动态池化网络的框架，再将检测器中设计的自适应标准化模块和预测器的结构作了详细说明，最后对训练的流程和各个步骤进行了详细的阐述。下一章将通过实验论证该方法在不同的骨干网络上的有效性，以及证明了各个模块和部分设计的有效性，通过实验的性能介绍和实验结果的可视化来进一步证明方法的泛化能力。

第 4 章　实验结果及分析

4.1　评价指标选择

本研究主要使用类别平均精度（mean Average Precision，mAP）、类别平均召回率（mean Average Recall，mAR）、计算量（floating point operations，$FLOPs$）和参数量（$Params$）。

mAP 是在各种检测任务中广泛使用的指标，反映了 $Recall$（召回率）和 $Precision$（准确率）在类别上平均后的检测结果的指标。并且将评测选取的交并比（IoU）的阈值设置为 0.5 和 0.75。按照 COCO 数据集的标准评测指标将目标按尺度（$\sqrt{长 \times 宽}$）划分为几个区间分为 3 个子区间：small[0，32)，medium[32，96)，large[96，∞)。本研究更多地关注是否可以找到对象，而不是位置精度。因此，实验选择 IoU = 0.5 作为评测的主要阈值。首先介绍 Recall 和 Precision 的计算过程。在训练集上学习到检测模型之后，测试集上的每一张图片经过网络的前向计算后都会得到本张图的检测结果集合 B，B 中每个检测样本外接矩形框 $b(x, y, w, h, c, score)$，k 代表 d 在 B 中的位序，x、y、w、h、c、$score$ 具体含义分别是代表矩形框的中心坐标的 (x, y)，矩形框宽和高 (w, h)，还有预测的类别 c，以及得分置信度 $score$。根据 x、y、w、h 可以计算出当前样本与图中标注框（ground truth）的 IoU。IoU 的计算如下：

$$IoU = \frac{\text{预测框区域} \cap \text{标定框区域}}{\text{预测框区域} \cup \text{标定框区域}} \tag{4.1}$$

根据样本的得分 score 依据预先设定的得分阈值可以判断样本是正例还是反例，对所有样本的正反例评判有如下四种情况：1.True Positive（TP）：正确的正例，正例样本被检测模型正确的判定为正例的样本；2.False Positive（FP）：错误的正例，反例样本被检测模型错误的判定为正例的样本；3.True Negative（TN）：正确的反例，反例样本被检测模型正确的判定为反例的样本；4.False Negative（FN）：错误的反例，正例样本被检测模型错误的判定为反例的样本。基于上述四个定义，$Recall$ 和 $Precision$ 可以用如下公式计算：

$$Recall = \frac{TP}{TP + FN} \tag{4.2}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.3}$$

在给定 IoU 阈值以及确定类别情况下，对所有同一类测试样本的得分降序排序，得分越高的排序越靠前。接下来划定一个得分阈值，依照降序排序遍历所有的检测结果，每个检测结果和标注框（ground truth）都会计算出一个 IoU，如果大于给定 IoU 阈值则计为匹配成功（TP），并且对应的 ground truth 不会参与到后续匹配过程，如果没有超过阈值则为 FP，重复匹配过程，最后小于给定得分阈值但是大于给定 IoU 阈值的检测结果则为 FN。给定一个得分阈值就可以得到一个 $Recall$ 和一个 $Precision$，通过不断的调整正反例样本的得分阈值形成不同的 $Recall$ 和 $Precision$，最后会得到一条 PR 曲线。对 PR 曲线有两种计算方法:1. 将横坐标的 $Recall$ 采样 11 个等分点，然后将 11 个等分点对应的 $Precision$ 计算平均值得到 mAP；2. 对 PR 曲线下的面积求积分得到 mAP。第二种方法求出来的 mAP 更为精确，但是考虑到不同数据集的数量差异对计算结果的影响，一般采用第一种方法，即通过采样 11 个等间距点的方法可以对 PR 曲线下求面积得到 AP，对于每个类别都会计算得到一个 AP，对于所有的类别的 AP 计算平均值就可以得到实验采用的评测指标 mAP。

按照类似的方式，可以计算获得一条 $Recall\text{-}IoU$ 曲线，AR 可以通过计算 $Recall\text{-}IoU$ 曲线下面积的两倍获得，而类别平均召回率（mAR）可以通过所有类别的 AR 平均值获得。

$FLOPs$ 是指浮点运算数，可以用来衡量算法/模型的复杂度。$Params$ 可以用于衡量模型的大小，它的单位为个，但是由于很多模型参数量太大，所以一般取单位兆（M）来衡量。模型的参数量和计算量主要来自于卷积层和全连接层部分，以下将简要介绍卷积层和全连接层的计算量和参数量的计算方式。

对于卷积层，假设输入通道数为 C_i，输入特征图大小为 H_i, W_i，卷积核的大小为 $K1 \times K2$，步长为 $S1 \times S2$，输出通道数为 C_o，输出特征图大小为 H_o, W_o，padding=$P1 \times P2$ 卷积 bias 为 True 时，那么：

$$\begin{aligned} H_o &= \lfloor \frac{(H_i - K1 + 2 \times P1)}{S1} \rfloor + 1, \\ W_o &= \lfloor \frac{(W_i - K2 + 2 \times P2)}{S2} \rfloor + 1, \end{aligned} \tag{4.4}$$

$$
\begin{aligned}
FLOPs &= C_o \times H_o \times W_o \times (C_i \times K1 \times K2 + 1), \\
Params(weight) &= C_o \times (C_i \times K1 \times K2 + 1), \\
Params(bias) &= C_o.
\end{aligned} \tag{4.5}
$$

对于全连接层，假设输入神经元数量为 I，输出神经元数量为 O，那么在有 bias 时，全连接层的计算量为：

$$
\begin{aligned}
FLOPs &= (2 \times I - 1) \times O, \\
Params(weight) &= O, \\
Params(bias) &= I \times O + O.
\end{aligned} \tag{4.6}
$$

4.2　实验设置

本实验基于 COCO benchmark。如果没有特殊声明，网络初始权重选择 ImageNet 预训练的骨干网络权重。实验用的数据集为 TinyCOCO，TinyCOCO 是由 COCO 数据集 [71] 进行保持长宽比的条件下重缩放至短边为 100 获得的数据集。它类别数与 COCO 检测集一致，具有 80 种目标类别，分别包括训练集图片数量 118287，但目标平均尺度降至 24.21，具体数据统计如表4.1。

实验将对检测器和降采样因子预测器分别训练 12 个周期，取最后一个周期结束后的测试性能为整个训练的结果。训练采用的优化器为 SGD，动量为 0.9，权值衰减系数为 0.0001，初始学习率均为 0.01，至 8 和 11 个周期时分别降为上个周期的 0.1 倍。训练采用的卡为 GeForce RTX 3090 ×8。为了充分发挥放大图片的优势，如未特殊说明，图片将被统一放大为短边尺度 800 作为输入大小。

实验部分，检测器采用单阶段 anchor-free 的检测器 RepPoints，骨干网络采用了不同的分类网络，选择了 ResNet50，ResNet101，ResNeXt50 和 MobileNetV2。表4.2、4.3、4.4罗列了不同骨干网络下检测器的部分设置细节。

表 4.1 TinyCOCO 和 COCO 对比

Table 4.1 Comparison of TinyCOCO and COCO

	TinyCOCO	COCO
图片数量	118287	118287
标注数量	788466	860001
目标平均尺度	24.21	99.50
目标最小尺度	2	0
目标最大尺度	129.19	640

表 4.2 ResNet 实验设置

Table 4.2 ResNet experimental setting

参数	值
model.backbone.type	ResNet_DR
model.backbone.out_indices	(0, 1, 2, 3)
model.backbone.norm_cfg.type	ABN
model.backbone.strides	(1, 1, 2, 2)
model.neck.type	FPN
model.neck.start_level	1
model.neck.in_channels	[256, 512, 1024, 2048]
model.neck.out_channels	256
model.neck.norm_cfg	AGN
model.neck.num_groups	32
model.bbox_head.type	RepPointsHead_DR
model.bbox_head.norm_cfg.type	AGN
model.bbox_head.norm_cfg.num_groups	32
model.bbox_head.point_strides	[8, 16, 32, 64, 128]
model.train_cfg.init.assigner.type	PointAssigner_DR

表 4.3 ResNeXt 实验设置

Table 4.3 ResNeXt experimental setting

参数	值
model.backbone.type	ResNeXt_DR
model.backbone.norm_cfg.type	ABN
model.backbone.strides	(1, 1, 2, 2)
model.neck.type	FPN
model.neck.start_level	1
model.neck.in_channels	[256, 512, 1024, 2048]
model.neck.out_channels	256
model.neck.norm_cfg	AGN
model.neck.num_groups	32
model.bbox_head.type	RepPointsHead_DR
model.bbox_head.norm_cfg.type	AGN
model.bbox_head.norm_cfg.num_groups	32
model.bbox_head.point_strides	[8, 16, 32, 64, 128]
model.train_cfg.init.assigner.type	PointAssigner_DR

表 4.4 MobileNetV2 实验设置

Table 4.4 MobileNetV2 experimental setting

参数	值
model.backbone.type	MobileNetV2_DR
model.backbone.norm_cfg.type	ABN
model.neck.type	FPN
model.neck.start_level	1
model.neck.in_channels	[24, 32, 96, 1280]
model.neck.out_channels	256
model.neck.norm_cfg	AGN
model.neck.num_groups	32
model.bbox_head.type	RepPointsHead_DR
model.bbox_head.norm_cfg.type	AGN
model.bbox_head.norm_cfg.num_groups	32
model.bbox_head.point_strides	[8, 16, 32, 64, 128]
model.train_cfg.init.assigner.type	PointAssigner_DR

4.3 实验结果

本节将介绍文中设计方法的实验结果，以及用一些实验阐述研究的意义和设计模块的有效性。本节实验如无特殊说明，默认采用骨干网络为 ResNet50 的 RepPoints 检测器。

4.3.1 研究与尺度相关的动态神经网络的意义

如3.1所阐述，检测任务的检测精度会极大地受到图像/目标尺度的影响，而不对网络的深度/宽度十分敏感。

表 4.5 改变输入大小的性能比较

Table 4.5 Performace comparison of different input sizes

输入尺寸	$mmAP$	mAP_{50}	mAP_s	mAP_m	mAP_l
(100, 167)	18.7	34.2	10.7	34.1	53.8
(150, 250)	23.4	40.2	14.6	41.4	57.7
(200, 333)	24.8	43.1	16.5	41.7	54.8

表 4.6 改变骨干网络深度的性能比较

Table 4.6 Performace comparison under different depths of backbone

骨干网络深度	$mmAP$	mAP_{50}	mAP_s	mAP_m	mAP_l
34	17.1	30.6	08.9	32.5	53.7
50	18.7	34.2	10.7	34.1	53.8
101	18.4	32.7	09.6	36.5	52.3
152	19.6	34.0	10.2	39.0	56.3

表 4.7 改变骨干网络宽度的性能比较

Table 4.7 Performace comparison under different widths of backbone

宽度 / ResNet50 原始宽度	$mmAP$	mAP_{50}	mAP_s	mAP_m	mAP_l
0.5	16.4	29.4	08.4	31.9	50.6
1	18.7	34.2	10.7	34.1	53.8
1.5	19.2	34.1	10.3	37.3	54.5

表4.5、4.6、4.7分别是改变输入图像尺寸、骨干网络深度、骨干网络宽度的性能比较。从这三个表的结果可以看出，只要适量增大输入图像的尺度，即可极大地增大检测性能，尤其是小目标的性能，当输入尺寸从（100，167）到（150,250）再到（200，333）时，mAP_s 将会从 10.7 增长至 14.6 再至 16.5，每次增大尺寸性能都有惊人的飞升。反观 mAP_m 和 mAP_l，AP 性能未有如此显著的增长，这也侧面说明了增大图像会带来一些冗余，不能在每一个尺度的目标检测精度上都有一致性的提升。但整体来说，$mmAP$ 都是提升明显的，说明放大图像带来的总体效益大过于它的缺陷。而把骨干网络的层数从 50 层加深至 152 层，$mmAP$ 仅从 18.7 增长至 19.6，通道数加宽至原本的 1.5 倍，$mmAP$ 仅从 18.7 增长至 19.2。在这两个维度上将网络结构复杂化并不能对 $mmAP$ 带来很明显的提升。由此可以说明，研究在特征图尺度层面上可变的动态神经网络是有意义的且收获巨大的。

4.3.2　DPNet 在不同骨干网络上的实现结果

DPNet 是在实验中基于 RepPoints 实现，表4.8和4.9首先展示了 DPNet 在不同的骨干网络上实现的结果和其对应的基准结果，然后展示了其他目标检测器在 TinyCOCO 上的检测结果。其中，† 表示 TinyCOCO 数据集放大图像训练后的测试结果，表中采用的检测器均为 RepPoints。从实验结果可以看出：1）放大输入图像可以很大程度上提升小目标检测性能，放大图像在 ResNeXt50、MobileNetV2、ResNet50 和 ResNet101 上的 $mmAP$ 提升分别为 10.4、10.3、10.4 和 13.3，$mmAR$ 提升分别为 15、14.2、14.9 和 17.5，其中，在小尺度目标上的性能提升最为明显，mAP_s 提升分别为 12.3、10.4、11.8 和 15.0，mAR_s 提升分别为 17.6、15.6、17.4 和 20.3；2）放大输入图像对大目标的检测性能提升不一定能起到作用，mAP_l 在 ResNeXt50、MobileNetV2、ResNet50 和 ResNet101 上没有一致性的效果，有一部分目标在不放大图像的情况下也能被很好地检测出来，即放大图像操作是有一定的计算冗余的；3）DPNet 甚至可以在一定程度上提升性能，和单纯放大图像操作相比，DPNet 在 ResNet50 的 $mmAP$ 提升为 0.3，也就是说，DPNet 不仅可以帮助网络自适应地调整降采样因子，更能帮助提升检测性能，因为它一定程度上调整了更适合输入图片进行检测的尺度。中大尺度的目标在放大到一定程度后就可以获得最佳的检测性能，当一味放大时，放大带来的缺点（负例数增多）反而会影响检测这些尺度上的目标的性能。

表4.10表示 DPNet 在不同的骨干网络上实现后的模型计算量和参数量，以及

表 4.8 不同网络的 AP 性能

Table 4.8 AP performance of different networks

模型	$mmAP$	mAP_{50}	mAP_s	mAP_m	mAP_l
RepPoints					
RepPoints-ResNeXt-50	19.5	34.5	10.7	37.9	59.0
RepPoints-ResNeXt-50†	29.9	48.3	23.0	44.5	52.8
DPNet-ResNeXt-50†	29.8	48.2	22.9	44.7	53.1
RepPoints-MobileNet-v2	14.9	26.7	8.3	27.3	43.8
RepPoints-MobileNet-v2†	25.2	41.7	18.7	38.6	50.4
DPNet-MobileNet-v2†	25.1	41.9	17.5	40.0	52.3
RepPoints-ResNet-50	18.7	34.2	10.7	34.1	53.8
RepPoints-ResNet-50†	29.4	47.7	22.5	44.2	53.6
DPNet-ResNet-50†	29.7	48.5	22.0	46.5	56.6
RepPoints-ResNet-101	18.4	32.7	9.6	36.5	52.3
RepPoints-ResNet-101†	31.7	50.7	24.6	47.3	56.1
DPNet-ResNet-101†	31.6	50.8	23.8	48.8	59.8
其他					
RetinaNet-ResNet-50	11.1	22.8	05.1	20.2	39.9
RetinaNet-ResNet-50†	26.3	41.7	19.7	41.2	47.8
Faster R-CNN-ResNet-50	15.9	31.1	08.6	30.2	45.9
Faster R-CNN-ResNet-50†	28.2	46.6	21.2	43.5	51.5
Cascade R-CNN-ResNet-50	18.0	32.3	09.7	34.4	53.7
Cascade R-CNN-ResNet-50†	30.7	46.5	23.0	47.8	54.1

表 4.9 不同网络的 AR 性能

Table 4.9 AR performance of different networks

模型	$mmAR$	mAR_s	mAR_m	mAR_l
RepPoints				
RepPoints-ResNeXt-50	32.5	22.9	55.0	73.5
RepPoints-ResNeXt-50†	47.5	40.5	63.0	74.5
DPNet-ResNeXt-50†	47.4	40.4	63.2	74.7
RepPoints-MobileNet-v2	30.1	21.0	509	74.9
RepPoints-MobileNet-v2†	44.3	36.7	60.4	74.6
DPNet-MobileNet-v2†	44.5	36.7	60.8	76.0
RepPoints-ResNet-50	32.3	22.6	54.7	72.4
RepPoints-ResNet-50†	47.2	40.0	63.0	75.2
DPNet-ResNet-50†	46.3	38.4	64.3	77.7
RepPoints-ResNet-101	31.7	21.8	54.3	70.0
RepPoints-ResNet-101†	49.2	42.1	64.9	73.5
DPNet-ResNet-101†	49.2	41.7	65.3	77.2
其他				
RetinaNet-ResNet-50	24.2	16.2	40.0	62.2
RetinaNet-ResNet-50†	43.3	35.7	61.2	72.4
Faster R-CNN-ResNet-50	27.1	18.7	44.1	54.0
Faster R-CNN-ResNet-50†	40.9	33.5	57.5	61.9
Cascade R-CNN-ResNet-50	29.1	19.9	48.9	63.2
Cascade R-CNN-ResNet-50†	41.7	33.4	59.6	68.0

基准网络的计算量和参数量。由于检测器网络的计算量较大，所以此处采用的单位为 GFLOPs。

表 4.10 模型大小与复杂度比较

Table 4.10 Comparison of model size and complexity

模型	Params	GFLOPs	↓FLOPs
RepPoints-ResNeXt-50†	35.44 M	168.25	-
DPNet-ResNeXt-50†	38.46 M	108.44	↓35.55%
RepPoints-MobileNet-v2†	8.52 M	95.71	-
DPNet-MobileNet-v2†	11.06 M	56.08	↓41.40%
RepPoints-ResNet-50†	36.62 M	168.25	-
DPNet-ResNet-50†	39.64 M	107.80	↓35.93%
RepPoints-ResNet-101†	55.62 M	217.54	-
DPNet-ResNet-101†	58.64 M	157.78	↓27.47%

单纯放大图片是不会增加网络的的参数量的，DPNet 引入了降采样因子预测器，所以增加了网络的总体参数量。但从表中的数据可以看出，DPNet 仅在增加极少网络参数量的情况下就能显著减少网络的计算量，DPNet 在 ResNeXt50、MobileNetV2、ResNet50 和 ResNet101 分别减少了 35.55%、41.40%、35.93% 和 27.47% 的 FLOPs。再结合表4.8和表4.9可以看出，DPNet 不仅在降低计算量上一致性有效，还能够维持网络原有的性能甚至超过原有的性能，真正实现了计算量和性能的双赢。此处值得一提的是，MobileNet 是一类轻量级的网络，十分适用于移动设备，本文设计的方法可以在轻量级网络上使用见效，一定程度上也说明，未来它也许能够结合网络压缩方法得到锦上添花的作用。

4.3.3 消融实验

4.3.3.1 ANM 的作用

为了验证自适应标准化模块的有效性，本小节将比较采用 ANM 和不采用 ANM 的检测结果。表4.11和4.12是关于 ANM 的作用的实验结果。

表 4.11 ANM 的作用-AP 性能

Table 4.11 Influence of ANM.

训练方法	降采样因子（测试）	$mmAP$	mAP_{50}	mAP_s	mAP_m	mAP_l
SF	0.5	29.4	47.7	22.5	44.2	53.6
	0.33	25.3	41.5	17.4	42.9	52.9
	0.25	19.5	32.5	11.4	38.3	48.4
MF	0.5	24.1	40.4	17.9	37.1	45.9
	0.33	23.1	39.2	15.7	38.8	48.7
	0.25	21.3	36.2	13.6	37.4	50.3
MF+ANM	0.5	29.5	47.9	22.6	43.8	52.2
	0.33	28.9	47.6	21.0	46.8	57.1
	0.25	27.4	45.0	18.5	47.2	59.5

其中，“SF”表示单一降采样因子训练，即只采用 0.5 作为唯一参与训练的降采样因子，“MF”表示混合降采样因子训练方式，具体流程见3.2.1。表格的第二列表示模型统一采用了同一降采样因子做测试。如无特殊说明，DPNet 在本文实现时预测的候选降采样因子为 [0.5，0.33，0.25]。

实验结果表明：1）混合降采样因子训练方式有提升网络在不同尺度的降采样因子上测试的潜力，单一降采样因子训练过的检测器在 0.25 的降采样因子上做测试时 $mmAP$ 仅为 19.5，$mmAR$ 为 38.0，而在混合降采样因子训练过后为 27.4 和 43.2；2）虽然单纯的混合降采样因子训练方式不能提升模型在 0.5 和 0.33 两个降采样因子上的测试性能，但是加上自适应标准化模块以后，模型的性能在不同的测试降采样因子上都得到了一致性的性能提升，其中，相比于没加自适应标准化模块前，模型在 0.5、0.33 和 0.25 三个降采样因子上的 $mmAP$ 提升分别为 0.1、3.6 和 7.9，mAP_{50} 提升分别为 0.2、6.1 和 17.5，$mmAR$0.33 和 0.25 两个降采样因子上提升分别为 3.4 和 5.2；3）混合降采样因子 + 自适应标准化模块训练过的模型在不同降采样因子上检测各有优势，0.5 的降采样因子更适合检测小目标，其 mAP_s 为 22.6，mAR_s 为 40.1，而 0.33 和 0.25 的降采样因子更适合检测中

表 4.12 ANM 的作用-AR 性能

Table 4.12 Influence of ANM.

训练方法	降采样因子（测试）	$mmAR$	mAR_s	mAR_m	mAR_l
SF	0.5	47.2	40.0	62.9	75.4
	0.33	42.0	33.4	61.9	73.5
	0.25	38.0	32.0	60.4	73.0
MF	0.5	42.6	35.5	58.6	69.0
	0.33	40.4	32.4	59.3	72.5
	0.25	38.3	29.5	59.3	73.7
MF+ANM	0.5	47.1	40.1	63.0	74.8
	0.33	45.4	37.3	64.4	77.3
	0.25	43.2	34.3	64.5	78.2

大尺度的目标，其 mAP_m 为 46.8 和 47.2，mAR_m 为 64.4 和 64.5，mAP_l 为 57.1 和 59.5 其 mAR_l 为 77.3 和 78.2，这也侧面说明，为不同输入图像选择不同的降采样因子确实有提高网络整体性能的潜力。

4.3.3.2 指导损失的作用

为了验证指导损失的有效性，实验首先用指导损失和原始损失获得的监督信息分别作为降采样因子直接传递给检测器，这样得到的性能称为该监督下的性能上界，性能上界越高，则说明该监督信息更好。

表 4.13 加入指导损失的 AP 性能上界影响

Table 4.13 AP Performance upper bound influence of guidance loss

方法	$mmAP$	mAP_{50}	mAP_s	mAP_m	mAP_l
原始损失	29.8	47.8	21.6	48.2	60.4
指导损失	30.9	50.1	23.0	47.8	59.4

表 4.14 加入指导损失的 AR 性能上界影响

Table 4.14 AR Performance upper bound influence of guidance loss

方法	$mmAR$	mAR_s	mAR_m	mAR_l
原始损失	45.8	37.6	65.5	78.8
指导损失	47.4	39.6	65.0	77.5

表4.13和表4.14是用指导损失生成的监督和原始损失生成的监督的性能上界结果，可以看出，通过指导损失获得的监督信息远高于用原始损失获得的监督信息，$mmAP$ 比通过原始损失获得的监督信息计算的性能上界高 1.1，$mmAR$ 比通过原始损失获得的监督信息计算的性能上界高 1.6。这也说明，监督损失更能反应图片的性能。监督损失越低，图片在检测器中的检测性能越高。所以为了让图片得到适宜的降采样因子，应当通过衡量不同的降采样因子计算的监督损失。

4.3.3.3 降采样因子预测器的作用

为了证明降采样因子预测器的作用，实验将首先计算 DPNet 的计算量，用同等计算量（即同等降采样因子的比例）对降采样因子进行随机采样，为了公平起见，实验通过三次随机采样得到的检测结果和 DPNet 进行比较。

表 4.15 DPNet vs. 随机降采样因子

Table 4.15 DPNet vs. Random Factors

模型	GFLOPs	$mmAP$	mAP_{50}
Random-1	107.80	28.9	47.5
Random-2	107.80	29.0	47.7
Random-3	107.80	28.9	47.6
Random(mean)	107.80	28.93	47.7
DPNet	107.80	29.7	48.5

表4.15是 DPNet 随机降采样方法的对比，通过实验结果可以看出，在同等计算量的情况下，DPNet 的 $mmAP$ 比随机降采样因子要高接近 0.8，mAP_{50} 比随机

降采样因子方法高 0.8，这也说明，降采样因子预测器是起到了实际作用的，它能够根据图片内容有针对性地选择合适的降采样因子而传给骨干网络的下一个阶段，最后实现网络的自适应降采样因子选择，达到计算量与性能的权衡。

4.3.4 可视化结果

图4.1为 DPNet 在 TinyCOCO 验证集上部分图片的预测可视化结果。由于实际测试时的图片较为模糊，目标尺度过小，为了便于展示，本文将预测结果于对应的 COCO 数据集中的清晰图片上做展示。

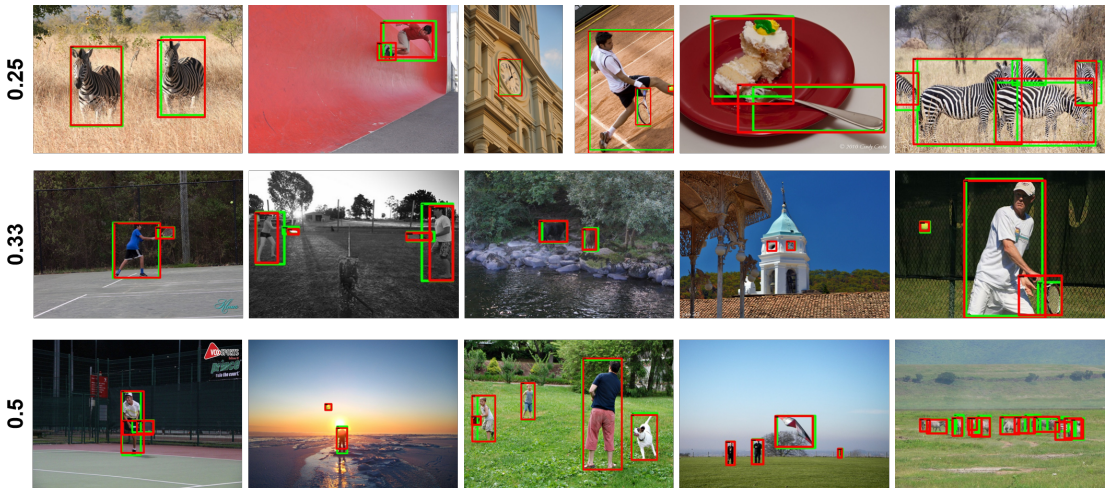


图 4.1 DPNet 在 TinyCOCO 验证集上的部分预测结果可视化

Figure 4.1 Image visualization results of DPNet on validation.

图4.1的左列数字表示降采样因子预测器的预测值，右列图片是预测结果，其中，红框为预测框，绿框为真值框。

4.4 本章小结

第四章介绍了研究结果及分析，4.1小节阐述了本文选择的评价指标与主要的计算过程；4.2小节介绍了本研究中实验的条件和具体的实验设置；4.3小节首先通过实验表明了本研究的目的和意义，然后展示了本研究所提出的 DPNet 在不同的骨干网络上实验的性能结果，接下来通过一系列消融实验阐明了本文中所设计的方法的有效性，每个部分都对结果做了详细阐述和分析。下一章将是本论文的总结与展望。

第 5 章　结论与展望

弱小目标检测 (Tiny Object Detection) 作为计算机视觉领域的子课题, 有着广泛的应用前景和科研价值。本研究在检测任务上提出了一种新的方法——动态池化网络，并且探究了该方法的意义，解决了该方法实现过程中的困难，针对性地提出了自适应标准化模块，在研究降采样因子预测器时设计了指导损失来获得预测器的监督。详尽的实验结果证明了本文提出的方法的有效性。本节将对研究的内容进行整体归纳，并规划未来的研究内容。

5.0.1　全文总结

本文首先介绍了计算机视觉这一领域, 然后着重介绍了弱小目标检测这一子任务, 并且说明了其存在的重要科研价值和应用前景。接下来阐述了放大输入图像对于小目标检测的具体优势，指出了其隐含的缺点和不足，同时提出了降采样因子来取代检测器骨干网络中的固定降采样操作，并借此简单介绍了文中设计的动态池化网络。

第二章对本研究涉及到几个方面: 检测算法、检测数据集、小目标检测和动态神经网络，文章对这四方面的前沿与经典工作进行了总结和阐述。

接下来论文从检测器常用的骨干网络的阐述出发，通过分析说明了动态池化网络的研究背景以及意义，介绍了动态池化网络的框架图，详细说明了混合降采样因子训练检测器的流程和专门设计的自适应规范化模块的结构以及降采样因子预测器的结构，还介绍了动态池化网络的详细训练流程和优化方法，为此说明介绍了在训练降采样因子预测器过程中为生成监督信息而设计的指导损失。

第四章基于设计的方法进行了一系列可选严谨的实验，包括方法的研究意义的实验，所提方法在不同的骨干网络上实现的结果都获得一致性提升，还有各个部分的消融实验也证明了设计的每个部分的有效性。

5.0.2　未来展望

科研工作的目的是可以运用到军用、民用、商用等各个领域发挥更大的价值, 但是从目前的深度学习的“黑盒”算法的特点来看, 科研成果距离实际落地应用还有不小的距离, 未来的工作也可以从缩短这一距离以及对算法的优化几个方

面展开。

对降采样因子预测的模块可以更灵活化，当前的降采样因子预测及改变的模块被固定到了某一位置，此后若有需要，应尝试设计更灵活的插入方法。

当前的降采样因子是作用于图片，无法兼顾到图片上的每一个目标，只能选择一个最为合适的权衡下的降采样因子，之后可以更进一步探究更详细的实例级别的信息，设法在实例级别上做到自适应调整。

从第四章的实验结果可以看出，预测器还有更大地潜力将性能提升，可以尝试将预测器的预测任务转化为别的任务，而不只是分类任务，也许可以将预测器在训练的过程中视为一个网络蒸馏的过程，用大的检测器去作为它的目标学习，这样可以更进一步利用到检测器的信息，加强检测器和预测器的联系。

可以尝试探究如何将网络训练改为端到端训练，当前流行和快捷的方式是端到端训练，这样也比较省时省力，如果可以将动态池化网络的训练方式改为端到端的方式，那么它的可用性将会更强，利用起来也更方便。

本文设计的方法旨在能够将检测器模型用于实际设备上，并针对不同计算资源的设备训练合适的模型，充分对模型计算量进行重分配，未来若能将该方法实际落地到应用场景中，动态池化网络将发挥它更大的价值。

参考文献

[1] Li Y, Chen Y, *et al.* N W. Scale-aware trident networks for object detection [C]//IEEE International Conference on Computer Vision (ICCV). 2019: 6053-6062.

[2] Deng C, Wang M, Liu L, et al. Extended feature pyramid network for small object detection [J]. IEEE Trans. Multim., 2022, 24: 1968-1979.

[3] Yu X, Gong Y, *et al.* N J. Scale match for tiny person detection [C]//IEEE Winter Conference on Applications of Computer Vision(WACV). 2020: 1246-1254.

[4] Han Y, Huang G, Song S, et al. Dynamic neural networks: A survey [J]. CoRR, 2021, abs/2102.04906.

[5] He K, Zhang X, Ren S. Deep residual learning for image recognition [C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2016: 770-778.

[6] 薛政钢. 基于多群体蚁群算法的多无人机协同搜索方法研究 [D]. 河南大学, 2018.

[7] Zhu P, Wen L, Bian X, et al. Vision meets drones: A challenge [J]. CoRR, 2018, abs/1804.07437.

[8] Cai Z, *et al.* N V. Cascade r-cnn: Delving into high quality object detection [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 6154-6162.

[9] Lin T, Dollár P, *et al.* R B G. Feature pyramid networks for object detection [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017: 936-944.

[10] Pang J, Chen K, Shi J, et al. Libra r-cnn: Towards balanced learning for object detection [C]// IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2019: 821-830.

[11] Lin T, Goyal P, Girshick R B, et al. Focal loss for dense object detection [C]//IEEE Trans. Pattern Anal. Mach. Intell. (PAMI). 2020: 318-327.

[12] Tian Z, Shen C, Chen H, et al. Fcos: Fully convolutional one-stage object detection [C]//IEEE International Conference on Computer Vision (ICCV). 2019: 9626-9635.

[13] Jiang N, Yu X, *et al.* X P. SM+: refined scale match for tiny person detection [C]//IEEE International Conference on Acoustics, Speech and Signal Processing,(ICASSP). 2021: 1815-1819.

[14] Lowe D G. Distinctive image features from scale-invariant keypoints [J]. Int. J. Comput. Vis., 2004, 60(2): 91-110.

[15] Dalal N, Triggs B. Histograms of oriented gradients for human detection [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2005: 886-893.

[16] Felzenszwalb P F, McAllester D A, Ramanan D. A discriminatively trained, multiscale,

deformable part model [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2008.

[17] Wang X, Han T X, Yan S. An HOG-LBP human detector with partial occlusion handling [C]// IEEE International Conference on Computer Vision (ICCV). 2009: 32-39.

[18] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [C]//Advances in Neural Information Processing Systems (NeurIPS). 2012: 1106-1114.

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

[20] 宋姚烨. 复杂交通环境下的车辆检测算法研究 [D]. 江苏大学, 2020.

[21] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition [C]//European Conference on Computer Vision (ECCV). 2014: 346-361.

[22] Girshick R B. Fast R-CNN [C]//IEEE International Conference on Computer Vision (ICCV). 2015: 1440-1448.

[23] Ren S, He K, Girshick R B, et al. Faster R-CNN: towards real-time object detection with region proposal networks [C]//Advances in Neural Information Processing Systems (NeurIPS). 2015: 91-99.

[24] He K, Gkioxari G, Dollár P, et al. Mask R-CNN [C]//IEEE International Conference on Computer Vision (ICCV). 2017: 2980-2988.

[25] Zhang S, Wen L, Bian X, et al. Single-shot refinement neural network for object detection [C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 4203-4212.

[26] Lu X, Li B, Yue Y, et al. Grid R-CNN [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019: 7363-7372.

[27] Redmon J, Divvala S K, Girshick R B, et al. You only look once: Unified, real-time object detection [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 779-788.

[28] Liu W, Anguelov D, *et al.* D E. SSD: single shot multibox detector [C]//European Conference on Computer Vision (ECCV). 2016: 21-37.

[29] Lin T, Goyal P, Girshick R B, et al. Focal loss for dense object detection [C]//IEEE International Conference on Computer Vision (ICCV). 2017: 2999-3007.

[30] Zhu C, He Y, Savvides M. Feature selective anchor-free module for single-shot object detection [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019: 840-849.

[31] Law H, Deng J. Cornernet: Detecting objects as paired keypoints [C]//European Conference on Computer Vision (ECCV). 2018: 765-781.

[32] Yang Z, Liu S, *et al.* H H. Reppoints: Point set representation for object detection [C]//IEEE International Conference on Computer Vision (ICCV). 2019: 9656-9665.

[33] Everingham M, Gool L V, Williams C K I, et al. The pascal visual object classes (VOC) challenge [J]. Int. J. Comput. Vis., 2010, 88(2): 303-338.

[34] Gupta A, Dollár P, Girshick R B. LVIS: A dataset for large vocabulary instance segmentation [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019: 5356-5364.

[35] Yang S, Luo P, Loy C C, et al. WIDER FACE: A face detection benchmark [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 5525-5533.

[36] Dollár P, Wojek C, Schiele B, et al. Pedestrian detection: An evaluation of the state of the art [J]. IEEE Trans. Pattern Anal. Mach. Intell. (PAMI), 2012: 743-761.

[37] Zhang S, Benenson R, Schiele B. Citypersons: A diverse dataset for pedestrian detection [C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017: 4457-4465.

[38] Shao S, Li Z, Zhang T, et al. Objects365: A large-scale, high-quality dataset for object detection [C]//IEEE International Conference on Computer Vision (ICCV). 2019: 8429-8438.

[39] Ess A, Leibe B, Schindler K, et al. A mobile vision system for robust multi-person tracking [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2008.

[40] Enzweiler M, Gavrila D M. Monocular pedestrian detection: Survey and experiments [J]. IEEE Trans. Pattern Anal. Mach. Intell. (PAMI), 2009: 2179-2195.

[41] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? the KITTI vision benchmark suite [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2012: 3354-3361.

[42] Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 3213-3223.

[43] Pang J, Li C, Shi J, et al. $\mathscr{R}^2$-cnn: Fast tiny object detection in large-scale remote sensing images [J]. IEEE Trans. Geosci. Remote. Sens., 2019, 57(8): 5512-5524.

[44] Singh B, Davis L S. An analysis of scale invariance in object detection SNIP [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 3578-3587.

[45] Singh B, Najibi M, Davis L S. SNIPER: efficient multi-scale training [C]//Advances in Neural Information Processing Systems (NeurIPS). 2018: 9333-9343.

[46] Noh J, Bae W, *et al.* W L. Better to follow, follow to be better: Towards precise supervision of feature super-resolution for small object detection [C]//IEEE International Conference on Computer Vision (ICCV). 2019: 9724-9733.

[47] Chen Y, Zhang P, *etal.* Z L. Stitcher: Feedback-driven data provider for object detection [J]. CoRR, 2020, abs/2004.12432.

[48] Liu Z, Gao G, Sun L, et al. Ipg-net: Image pyramid guidance network for small object detection [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020: 4422-4430.

[49] Bolukbasi T, Wang J, Dekel O, et al. Adaptive neural networks for efficient inference [C]// International Conference on Machine Learning (ICML). 2017: 527-536.

[50] Teerapittayanon S, McDanel B, Kung H T. Branchynet: Fast inference via early exiting from deep neural networks [C]//International Conference on Pattern Recognition (ICPR). 2016: 2464-2469.

[51] Huang G, Chen D, *et al.* T L. Multi-scale dense networks for resource efficient image classification [C]//International Conference on Learning Representations (ICLR). 2018.

[52] Graves A. Adaptive computation time for recurrent neural networks [J/OL]. CoRR, 2016, abs/1603.08983. http://arxiv.org/abs/1603.08983.

[53] Wang X, Yu F, Dou Z, et al. Skipnet: Learning dynamic routing in convolutional networks [C]//European Conference on Computer Vision (ECCV). 2018: 420-436.

[54] Wu Z, Nagarajan T, Kumar A, et al. Blockdrop: Dynamic inference paths in residual networks [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 8817-8826.

[55] Jacobs R A, Jordan M I, Nowlan S J, et al. Adaptive mixtures of local experts [J]. Neural Comput., 1991, 3(1): 79-87.

[56] Eigen D, Ranzato M, Sutskever I. Learning factored representations in a deep mixture of experts [C]//International Conference on Learning Representations (ICLR). 2014.

[57] Hua W, Zhou Y, Sa C D, et al. Channel gating neural networks [C]//Advances in Neural Information Processing Systems (NeurIPS). 2019: 1884-1894.

[58] Lin J, Rao Y, Lu J, et al. Runtime neural pruning [C]//Advances in Neural Information Processing Systems (NeurIPS). 2017: 2181-2191.

[59] Liu C, Wang Y, Han K, et al. Learning instance-wise sparsity for accelerating deep models [C]// Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019. 2019: 3001-3007.

[60] Wang Y, Shen J, Hu T, et al. Dual dynamic inference: Enabling more efficient, adaptive, and controllable deep inference [J]. IEEE J. Sel. Top. Signal Process., 14(4): 623-633.

[61] Xia W, Yin H, Dai X, et al. Fully dynamic inference with deep neural networks [J]. CoRR, 2020, abs/2007.15151.

[62] Zhu M, Han K, Wu E, et al. Dynamic resolution network [C]//Advances in Neural Information Processing Systems (NeurIPS). 2021: 27319-27330.

[63] Yang L, Han Y, *et al.* X C. Resolution adaptive networks for efficient inference [C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2020: 2366-2375.

[64] Xie S, Girshick R B, Dollár P, et al. Aggregated residual transformations for deep neural networks [C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2017: 5987-5995.

[65] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications [J]. CoRR, 2017, abs/1704.04861.

[66] Sandler M, Howard A G, Zhu M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks [C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2018: 4510-4520.

[67] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift [C]//International Conference on Machine Learning (ICML). 2015: 448-456.

[68] Perez E, Strub F, de Vries H, et al. Film: Visual reasoning with a general conditioning layer [C]//Association for the Advance of Artificial Intelligence (AAAI). 2018: 3942-3951.

[69] Li Y, Wang N, Shi J, et al. Revisiting batch normalization for practical domain adaptation [C]// International Conference on Learning Representations (ICLR) Workshop Trck Proceedings. 2017.

[70] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks [C]//International Conference on Learning Representations (ICLR). 2016.

[71] Lin T, Maire M, *et al.* S J B. Microsoft COCO: common objects in context [C]//European Conference on Computer Vision (ECCV). 2014: 740-755.

致 谢

感谢韩振军老师对我的指导和帮忙，本论文从选题、方案选择到实际撰写，韩老师都为我提出了许多实质性有用的意见和建议。同时要感谢韩老师在我读研三年期间对我的谆谆教诲。在我科研过程中，韩老师总是帮我分析思路，开拓视角，提供充足的设备和计算资源，为我顺利科研提供了强有力的基础；在我遇到困难的时候，韩老师总是给予我很大的支持和鼓励。韩老师严谨求实的治学态度，踏实坚韧的工作精神，将使我终身收益。除此之外，韩老师对我们在生活方面也十分照顾，关心我的身心健康，也在我科研之余交流我未来选择规划方面的问题，让我收益颇多。

感谢师兄们一直以来的帮助，余学辉师兄在我研究生期间提供了很多宝贵的帮助，在我对代码框架不熟悉时耐心地教我熟悉每个部分，讲解了很多编程时的技巧，同时他也为我的研究方向提供了有意义的指导意见，经常同我探讨方法的不足和可改进之处，为我讲述当前学界的前沿研究，给我开拓了思路，在我遇到瓶颈和困难时，他总能沉下心来与我一同分析当前的错误，并给我提供建议，他在科研和生活中不骄不躁的精神深深影响了我，属实难能可贵。蒋楠师兄的思维灵活且宽阔，他总能产生很多新奇的点子，论文的阅读数量也十分可观，在我研一时期为我入门深度学习提供了很多建议，指导我参加比赛，为我在后续的研究中打下了编程基础，同时，他高效的工作方式也成为了我的榜样，让我一直牢记于心。宫宇琦师兄是一个开朗但严谨的人，我从他身上学到了很多东西，他在我研一时期督促我阅读论文，让我在自己研究的领域建立了理论基础，研二时他为我的择业和科研进程的把握都提供了具体的建议，让我在读研时期减少了很多走弯路的可能，同时，他对待事情乐观又谨慎的态度也是我所钦佩的。感谢三位师兄对我一如既往的关照和支持。

感谢同届的王岿然同学和韩许盟同学，我们虽然研究的方向不同，但三年间一直是彼此的好友，也有过共同为了某一目标奋进努力及相互帮助的时期，对我来说十分珍贵。感谢实验室的师弟师妹们，吴狄和陈鹏飞在我撰写论文时为我提供了很多帮助，感谢他们不辞辛劳地陪我熬夜，和我共同探讨论文细节以及图片细节，为我的实验给予帮助，支持我走过了研究生生涯中很困难的一段时期，感

谢黄智勋、杨登杰、曹光明、陈皓睿、陆旭然一直以来的帮助和陪伴。感谢与我一同在国科大就读的陈雨豪和许严同学，从高中时期一直到研究生时期，他们都是我学习的榜样，也是为我排忧解难的好朋友。

最后要感谢我的父母和家人，感谢父母对我从小到大的辛勤付出，他们从不对我的成绩苛责，只希望我能健康平安地长大，我在这样愉快的家庭氛围中成长并得到他们一如既往地支持。在研究生就读期间，我的家庭总是我温暖的港湾，父母亲总是时时刻刻牵挂我，关心我，在我面对挫折时鼓励我，支持我，使我一次又一次为自己的未来坚定信念。

在接下来的人生之路中，我会带着所有人的希望和期羡继续坚定出发。

作者简历及攻读学位期间发表的学术论文与研究成果

作者简历：

彭潇珂，云南省丘北县人，出生日期为 1997 年 01 月 17 日。

2015 年 09 月——2019 年 06 月，在中山大学电子与通信工程学院获得学士学位。

2019 年 09 月——2022 年 06 月，在中国科学院大学电子电气与通信工程学院攻读硕士学位。

获奖情况

2021 年 06 月三好学生

已发表（或正式接受）的学术论文：

[1] Gong Y, Yu X, Ding Y, **Peng X**, Ding Y, Han Z. Effective Fusion Factor in FPN for Tiny Object Detection[J]. 2021. IEEE Winter Conference on Applications of Computer Vision（WACV）. IEEE, 2021.

[2] Jiang N, Yu X, **Peng X**, Gong Y and Han Z. SM+: Refined Scale Match for Tiny Person Detection[C]. In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2021.

[3] Jiang N†, Wang K†, **Peng X**†, Yu X, Wang Q, Xing J, Li G, Zhao J, Guo G, Han Z. Anti-UAV: A Large Multi-Modal Benchmark for UAV Tracking[J]. IEEE Transactions on Multimedia. († 表示共同一作.)

[4] **Peng X**, Wu D, Chen P, et al. DPNet: Dynamic Pooling Network for Accurate and Efficient Size-Aware Tiny Object Detection[C]. Submitted to IEEE International Conference on Computer Vision. Submitted to ACM International Conference on Multimedia.

申请或已获得的专利:

[1] 中国科学院大学. 一种基于尺度匹配的弱小人体目标检测方法：中国，201910918836.2[P].2019-09-26.（已授权）

[2] 中国科学院大学. 一种基于无监督深度孪生网络的视频去重方法：中国，202010214485.X[P]. 2020-03-24.（已授权）

[3] 中国科学院大学. 基于精确尺度匹配的弱小人体目标检测方法：中国，202010746942.X[P]. 2020-07-29.（已授权）

[4] 中国科学院大学. 基于多源信息融合的弱小目标检测方法：中国，202010215165.6[P]. 2020-03-24.（已授权）

[5] 中国科学院大学. 基于 FPN 的融合因子的弱小目标检测方法：中国，202010752490.6[P].2020-07-30.（已授权）