软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2021,32(4):1201−1227 [doi: 10.13328/j.cnki.jos.006166]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

多尺度目标检测的深度学习研究综述*

陈科圻[1,2,3], 朱志亮[2,3,4], 邓小明[2,3], 马翠霞[1,2,3], 王宏安[1,2,3]

[1](中国科学院大学 计算机科学与技术学院,北京 100190)

[2](计算机科学国家重点实验室(中国科学院 软件研究所),北京 100190)

[3](人机交互北京市重点实验室(中国科学院 软件研究所),北京 100190)

[4](华东交通大学 软件学院,江西 南昌 330013)

通讯作者: 马翠霞, E-mail: cuixia@iscas.ac.cn

摘 要: 目标检测一直以来都是计算机视觉领域的研究热点之一,其任务是返回给定图像中的单个或多个特定目标的类别与矩形包围框坐标.随着神经网络研究的飞速进展,R-CNN 检测器的诞生标志着目标检测正式进入深度学习时代,速度和精度相较于传统算法均有了极大的提升.但是,目标检测的尺度问题对于深度学习算法而言也始终是一个难题,即检测器对于尺度极大或极小目标的检测精度会显著下降,因此,近年来有不少学者在研究如何才能更好地实现多尺度目标检测.虽然已有一系列的综述文章从算法流程、网络结构、训练方式和数据集等方面对基于深度学习的目标检测算法进行了总结与分析,但对多尺度目标检测的归纳和整理却鲜有人涉足.因此,首先对基于深度学习的目标检测的两个主要算法流派的奠基过程进行了回顾,包括以 R-CNN 系列为代表的两阶段算法和以 YOLO、SSD 为代表的一阶段算法;然后,以多尺度目标检测的实现为核心,重点诠释了图像金字塔、构建网络内的特征金字塔等典型策略;最后,对多尺度目标检测的现状进行总结,并针对未来的研究方向进行展望.

关键词: 目标检测;深度学习;尺度问题;多尺度特征

中图法分类号: TP393

中文引用格式: 陈科圻,朱志亮,邓小明,马翠霞,王宏安.多尺度目标检测的深度学习研究综述.软件学报,2021,32(4):1201−1227. http://www.jos.org.cn/1000-9825/6166.htm

英文引用格式: Chen KQ, Zhu ZL, Deng XM, Ma CX, Wang HA. Deep learning for multi-scale object detection: A survey. Ruan Jian Xue Bao/Journal of Software, 2021,32(4):1201−1227 (in Chinese). http://www.jos.org.cn/1000-9825/6166.htm

Deep Learning for Multi-scale Object Detection: A Survey

CHEN Ke-Qi[1,2,3], ZHU Zhi-Liang[2,3,4], DENG Xiao-Ming[2,3], MA Cui-Xia[1,2,3], WANG Hong-An[1,2,3]

[1](School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100190, China)

[2](State Key Laboratory of Computer Science (Institute of Software, Chinese Academy of Sciences), Beijing 100190, China)

[3](Beijing Key Laboratory of Human-computer Interaction (Institute of Software, Chinese Academy of Sciences), Beijing 100190, China)

[4](School of Software, East China Jiaotong University, Nanchang 330013, China)

Abstract: Object detection is a classic computer vision task which aims to detect multiple objects of certain classes within a given image by bounding-box-level localization. With the rapid development of neural network technology and the birth of R-CNN detector as a milestone, a series of deep-learning-based object detectors have been developed in recent years, showing the overwhelming speed and accuracy advantage against traditional algorithms. However, how to precisely detect objects in large scale variance, also known as the scale problem, still remains a great challenge even for the deep learning methods, while many scholars have made several contributions to

* 基金项目: 国家重点研发计划(2016YFB1001200); 国家自然科学基金(61872346)

Foundation item: National Key Research and Development Program of China (2016YFB1001200); National Natural Science Foundation of China (61872346)

收稿时间: 2020-08-10; 修改时间: 2020-09-20; 采用时间: 2020-10-28; jos 在线出版时间: 2020-12-02

it over the last few years. Although there are already dozens of surveys focusing on the summarization of deep-learning-based object detectors in several aspects including algorithm procedure, network structure, training and datasets, very few of them concentrate on the methods of multi-scale object detection. Therefore, this paper firstly review the foundation of the deep-learning-based detectors in two main streams, including the two-stage detectors like R-CNN and one-stage detectors like YOLO and SSD. Then, the effective approaches are discussed to address the scale problems including most commonly used image pyramids, in-network feature pyramids, etc. At last, the current situations of the multi-scale object detection are concluded and the future research directions are looked ahead.

Key words: object detection; deep learning; scale problem; multi-scale feature

目标检测是一个重要的计算机视觉任务.它由图像分类任务发展而来,区别在于不再仅仅只对一张图像中的单一类型目标进行分类,而是要同时完成一张图像里可能存在的多个目标的分类和定位,其中分类是指给目标分配类别标签,定位是指确定目标的外围矩形框的顶点坐标.因此,目标检测任务更具有挑战性,也有着更广阔的应用前景,比如自动驾驶、人脸识别、行人检测、医疗检测等等.同时,目标检测也可以作为图像分割、图像描述、目标跟踪、动作识别等更复杂的计算机视觉任务的研究基础.

目标检测算法主要分为 3 个步骤:图像特征提取、候选区域生成与候选区域分类.其中,图像特征提取是整个检测流程的基石.传统算法普遍是基于人工设计的特征算子来描述图像,例如 SIFT 特征[1]、HOG 特征[2]等等.这些特征算子普遍是基于底层视觉特征来设计的,因此很难获取复杂图像里的语义信息.2012 年,Krizhevsky 等人[3]提出的 AlexNet 在 ILSVRC 挑战赛[4]的图像分类任务上以显著优势夺得了冠军,让人们看到了卷积神经网络强大的特征表示能力.自此,基于深度学习的研究热潮拉开了帷幕.在之后的几年里,VGGNet[5]、GoogLeNet[6]、ResNet[7]等更强大的分类网络相继问世.由于它们都能够提取出非常抽象的特征,因此除了完成图像分类任务以外,还普遍被用作更复杂的计算机视觉任务的骨架网络,其中就包括目标检测.2014 年,Girshick 等人[8]提出的 R-CNN 算法在 PSACAL VOC 检测数据集[9]上以绝对优势击败了传统的 DPM 算法[10],为目标检测开启了一个新的里程碑.自此,深度学习算法在目标检测的研究领域里占据了绝对的主导地位,并一直持续至今,近年有很多综述文献对此进行了详细的调研[11−13].

基于深度学习的目标检测算法主要分为两个流派:(1) 以 R-CNN 系列为代表的两阶段算法;(2) 以 YOLO[14]、SSD[15]为代表的一阶段算法.具体来说,两阶段算法首先在图像上生成候选区域,然后对每一个候选区域依次进行分类与边界回归;而一阶段算法则是直接在整张图像上完成所有目标的定位和分类,略过了生成候选区域这一步骤.两种流派各有优势,通常来说,前者精度更高,后者速度更快.对于现阶段的目标检测任务来说,无论采用哪种流派的算法,都不可避免地会面临着尺度问题的挑战:不同图像之间、甚至是同一张图像里,需要被检测出的目标的大小相对于整张图像的比例的差异是非常大的.例如,从图 1 可以看到,图 1(a)~图 1(c)这 3 张图中的大本钟、狗、人群的尺度是从大到小的,而图 1(d)所示的两张图则均包含了多种尺度的目标.对于检测任务常用的 MS COCO 数据集[16],若根据目标掩膜相对于图像的像素比例对所有实例的尺度进行统计和排序后会发现:数据集中有 10%的目标的尺度小于 0.020 7,同样有 10%的目标的尺度大于 0.345,尺度跨度极大.在该数据集的评价标准下,现有的检测器在检测不同尺度的目标时,普遍存在着精度不均衡的现象,即小目标的准确率通常只有中、大目标的准确率的一半左右.这种尺度差异带来的挑战性,严重限制了现有检测器的整体表现.因此,如何更好地实现多尺度目标检测,近年来一直是目标检测领域的研究热点之一.

目标检测包含了目标的定位和分类这两个子任务,而尺度问题的根源就在于,卷积神经网络在不断加深的过程中,表达抽象特征的能力越来越强,但浅层的空间信息也相对丢失.这就导致深层特征图无法提供细粒度的空间信息,对目标进行精确定位.同时,小目标的语义信息也在下采样的过程中逐渐丢失.因此,解决尺度问题的一个通用的思路就是构建多尺度的特征表达.目前,常用的构建多尺度特征的方法包括:(1) 采用图像金字塔,将图像以不同的分辨率依次进行目标检测;(2) 在神经网络内部通过对不同深度的特征图进行跨层连接,构建特征金字塔并进行目标检测;(3) 在神经网络内部设计感受野不同的并行支路,构建空间金字塔并进行目标检测.除了构建多尺度的特征表达以外,也有学者从算法流程中更细节的层面研究缩小不同尺度目标检测精度差距的策略,包括锚点、交并比、动态卷积、边界框损失函数等等.

本文围绕基于深度学习的多尺度目标检测,首先在第 1 节中对背景知识进行介绍,包括从两阶段和一阶段这两类算法的角度回顾主流检测器的奠基过程,以及尺度问题的提出.第 2 节介绍基于图像金字塔的多尺度目标检测.第 3 节介绍基于网络内特征金字塔的多尺度目标检测,包括跨层连接和并行支路两种构建金字塔的方式.第 4 节介绍多尺度目标检测的锚点、交并比、动态卷积、边界框损失函数等策略.第 5 节对多尺度目标检测未来可能的研究方向和趋势进行展望.第 6 节对全文进行总结.

(a) 大目标 (b) 中目标 (c) 小目标

(d) 多尺寸目标

Fig.1 The scale variance of the objects within different images or one image in object detection task

图 1 目标检测任务中不同图像或同一图像内的目标存在的尺度差异

1 背景知识

1.1 基于深度学习的目标检测主流算法

本节将对基于深度学习的目标检测算法中的两阶段算法和一阶段算法这两大主要流派的发展历史进行简要回顾.两阶段算法首先通过启发式方法或者卷积神经网络生成一系列可能存在潜在目标的候选区域,然后根据候选区域的特征对每一个区域进行分类和边界回归.一阶段算法则省略了候选区域生成的步骤,仅使用一个卷积神经网络直接完成整张图像上所有目标的定位与分类.两种流派各有优势:两阶段算法相对而言精度更高,尤其体现在定位上;而一阶段算法的速度普遍更快,更容易满足实际应用场景中的实时性需求.

1.1.1 两阶段算法

传统目标检测算法的流程主要包括候选框生成、特征提取和目标分类这 3 个步骤,而两阶段算法正是在传统算法的基础上一步步演化而来的.2014 年,Girshick 等人[8]提出的 R-CNN 算法,在 PASCAL VOC 数据集[9]上以绝对优势击败了经典的 DPM 算法[10],开启了基于深度学习的两阶段目标检测算法的先河.R-CNN 算法包括 3 个模块:(1) 采用选择性搜索[17]生成可能包含潜在目标的候选区域;(2) 将所有候选区域采样至某一固定分辨率后,逐一输入卷积神经网络,提取出固定长度的特征向量;(3) 采用多个支持向量机对所有特征向量进行分类.

最后,作者还额外增加了一个矩形框回归的步骤:根据已知类别和提取出的特征向量,对矩形框进行回归修正,从而进一步提高定位精度.相较于传统算法,R-CNN 的最大创新点在于不再需要人工设计特征算子,而是引入卷积神经网络自动学习如何更好地提取特征,实验结果也表明这样做是更有效的.但是,R-CNN 算法本身存在很多缺陷:(1) 虽然在提取特征向量时使用了 CNN,但生成候选区域采用的选择性搜索算法仍然还是基于底层视觉特征,因此候选框质量不高;(2) 算法的 3 个模块是相互独立的,导致训练过程繁琐,无法实现端到端的训练,且不能获得全局最优解;(3) 在提取特征向量时,每个候选区域都会被单独地从原图上裁剪下来,再依次输入神经网络,这样做既占用了大量磁盘空间,也带来了很多重复性计算,导致训练速度和推断速度都非常缓慢.

为了解决 R-CNN 算法的这些缺陷,后续出现了一系列算法对其进行改进.He 等人[18]意识到,卷积层本身并不限制输入图像的分辨率,R-CNN 之所以在提取特征向量前需要统一候选区域的尺寸,是因为 CNN 最后的全连接层只能处理固定尺寸的输入.于是,为了避免重复运算,他们提出的 SPP-Net 不再是将候选区域依次通入 CNN,而是直接计算整张图的特征图,然后划分出每一个候选区域的特征.在全连接层之前,为了统一特征向量的长度,他们新增了一个 SPP 层,通过池化操作将任意输入都转化为固定长度的输出.由此可以看出,相较于 R-CNN,SPP-Net 最大的贡献在于显著加速了训练和推断的过程.但是,SPP-Net 的精度与 R-CNN 并无明显差别,而且它的算法流程依旧是独立的多个模块,保存特征向量依旧需要大量存储空间.于是,Girshick 在此基础上,又提出了 Fast R-CNN[19].Fast R-CNN 吸纳了 SPP-Net 的思想,对整张图进行一次性的特征计算,新提出的 RoI 池化层相当于 SPP 层的简化版.除此以外,Fast R-CNN 为了简化流程,不再使用支持向量机进行分类,也不再使用额外的回归器,而是设计了多任务损失函数,直接训练 CNN 在两个新的网络分支上分别进行分类和回归.这也正是 Fast R-CNN 最大的创新:将特征提取、分类、回归整合为了一步,这样就不再需要中途保存特征向量,从而解决了存储空间的问题;而且在训练时能够进行整体的优化,因此取得了更高的精度.

Fast R-CNN 虽然成功地将分类和回归也整合进了神经网络,但距离真正的端到端训练还差一步:候选框的生成依旧是完全独立的.选择性搜索等传统算法是基于图像的底层视觉特征直接生成候选区域,无法根据具体的数据集进行学习.而且,选择性搜索非常耗时,在 CPU 上处理一张图像需要 2s.即便是当时能够最好地权衡候选框的生成质量与速度的 EdgeBoxes 算法[20],处理一张图像也需要 0.2s,与 Fast R-CNN 的神经网络部分的耗时差不多[21].因此,Ren 等人[21]再次对 Fast R-CNN 进行改进,提出了 Faster R-CNN 算法.该算法最大的创新点在于设计了 RPN 这样一个候选框生成网络.RPN 有两大创新:(1) RPN 的输入是已有的 Fast R-CNN 的骨架网络所提取的整张图像的特征图,这种共享特征的设计既充分利用了 CNN 的特征提取能力,又节省了运算;(2) 提出了锚点(anchor)概念,RPN 基于预先设定好尺寸的锚点进行分类(前景或背景)和回归,既确保了多尺度的候选框的生成,也让模型更易于收敛.RPN 生成候选区域之后,算法的剩余部分就和 Fast R-CNN 一致了.正因为有了 RPN 取代选择性搜索算法,Faster R-CNN 最终在 GPU 上的检测速度达到了 5FPS,打破了 PASCAL VOC 数据集的记录.同时,它还是第一个真正实现了端到端训练的检测算法,标志着两阶段检测器的正式成型.

自 Faster R-CNN 问世之后,新诞生的两阶段检测器几乎都以它为雏形.Dai 等人[22]提出的 R-FCN 为了进一步提高 Faster R-CNN 的效率,去除了各分支独立的计算耗时的全连接层,设计了位置敏感得分图和位置敏感 RoI 池化层来保留空间信息,显著提高了推断速度与精度.Lin 等人[23]考虑到网络深层特征有较强的语义信息,而浅层特征有较强的空间信息,于是提出了将深层特征图通过多次上采样和浅层特征图逐一结合的 FPN 架构,基于多层融合后的特征图进行输出,能够更好地检测到不同尺度的目标,是多尺度目标检测的里程碑.He 等人[24]提出的 Mask R-CNN 在 Faster R-CNN 的基础上将 RoI 池化层替换成了 RoI 对齐层,使得特征图和原图像素能对齐得更精准,并新增了一个掩膜分支,用于实例分割.令人惊讶的是,该算法不仅在实例分割任务上取得了优秀表现,对分类、回归、掩膜分支同时进行多任务训练也提高了目标检测任务的性能.Qin 等人[25]则提出了轻量级的二阶段检测器 ThunderNet:通过为检测任务定制轻量级骨架网络 SNet、对 RPN 和检测头进行压缩以及引入 CEM、SAM 等模块,让模型在速度和精度方面超越了不少一阶段检测器.

1.1.2 一阶段算法

从 Faster R-CNN 开始的二阶段算法虽然已经实现了端到端训练的完整流程,但是与真正满足实时性需求

仍有相当大的差距.因此,以 YOLO 算法[14]为代表的一阶段检测器便登上了舞台.这一类算法不再单独设计生成候选区域的初始阶段,而是在整张图像上一次性完成所有目标的定位与分类.Sermanent 等人[26]于 2013 年提出的 OverFeat 是最早的一阶段检测器.虽然它的精度不如同期 R-CNN,但其思想很有前瞻性:(1) 采用卷积层替代全连接层实现全卷积神经网络,适应不同分辨率的图像作为输入,相当于用卷积来快速实现滑动窗口算法;(2) 采用同一个卷积神经网络作为共享的骨架网络,通过更改网络头部分别实现分类、定位和检测任务,这使得 OverFeat 比 R-CNN 的检测速度快了 9 倍[8].2015 年,Redmon 等人[14]提出的 YOLO 算法则真正地实现了实时性目标检测.其核心思想是将目标检测视为一个回归任务,算法流程十分简洁:将输入图像划分为 7×7 的网格,每一个网格负责预测中心点处于该网格内的目标,回归中心点相对于网格的位置、目标的长宽和类别.YOLO 的损失函数由定位损失、置信度损失、分类损失这 3 部分组成,其中,置信度是指是否存在目标.可以看到,YOLO 是一种端到端的算法,没有候选框这一概念,输入一张图片,在检测到前景的同时就回归得到了需要的属性.从实验结果来看,YOLO 的检测速度能够达到 45FPS,Fast YOLO 甚至能到 155FPS,比二阶段检测器快了一个数量级.除此以外,YOLO 在检测时考虑了更多的背景信息,因此将背景误判为前景的概率比 Fast R-CNN 要低很多[14].当然,YOLO 也存在一些明显的缺陷:(1) 每一个网格只检测两个目标,且规定为同一类别,导致算法难以处理密集目标的检测;(2) 精度比 Fast R-CNN 要差,尤其体现在定位上,主要原因在于,后者经过了从整体到局部的两次矩形框回归,而 YOLO 只经过了一次;(3) 由于全连接层的存在,输入图像的分辨率是固定的;(4) 只在单张特征图上检测目标,导致算法难以驾驭多尺度目标的检测.

YOLO 诞生之后,更多的一阶段检测器也相继问世.Liu 等人[15]提出的 SSD 算法继承了 YOLO 的核心思想,而主要的不同之处在于:(1) 训练网络在多个不同深度的特征层上预测不同尺度的目标,最后进行整合;(2) 引入 Faster R-CNN[21]的锚点概念,使模型更容易收敛,保证不同感受野的特征图适应不同尺度的目标检测;(3) 使用全卷积神经网络,适应不同分辨率的图像输入;(4) 损失函数由定位损失和分类损失组成,没有 YOLO 的前景置信度这一概念,因为它在分类时直接将背景也视为一个类别,与其他类别同时进行预测.此外,SSD 在特征图上铺设了密集的锚点,而有效匹配目标的锚点个数是很有意义的,若直接采用所有样本进行训练,会存在严重的正负样本不平衡问题.于是,SSD 采用了难例挖掘的手段来缓解这一问题.从实验结果来看,SSD 的检测速度能和 YOLO 媲美,而精度能够匹敌 Faster R-CNN.不过,虽然 SSD 在多层特征图进行预测,但是相对于 Faster R-CNN,小目标的检测结果并未能够得到明显的改善.其主要原因可能在于,浅层的特征层虽然有着较小的感受野,但特征表示能力相对于深层特征要弱很多.

对原始的 YOLO 进行全面升级之后,Redmon 等人[27]推出了 YOLOv2.YOLOv2 做出了一系列改进:(1) 对所有的卷积层引入批量标准化;(2) 统一预训练与实际训练的图像分辨率;(3) 去除全连接层,引入锚点作为预测目标尺寸的参照物,锚点的尺寸通过 *k*-means 聚类来确定;(4) 设计了 passthrough 层,将不同层的特征图拼接在了一起,基于更丰富的特征进行预测;(5) 采用多尺度训练,使模型更鲁棒;(6) 设计了 Darknet-19 作为骨架网络,加快了速度.可以看到,YOLOv2 吸取了很多深度学习的技巧,最终在速度、精度上均得到提高.此外,Redmon 等人还同时提出了 YOLO 9000,采用树的形式将 ImageNet 分类数据集和 COCO 检测数据集的目标类别进行合并,通过联合训练分类任务和检测任务,让模型最终能够检测超过 9 000 种目标.YOLO 9000 的思想是想要消除分类和检测数据集在数据量上的鸿沟,这对于检测任务的拓展性有很大帮助.

新诞生的这一系列一阶段检测器虽然普遍有着绝对的速度优势,但与顶尖的二阶段检测器相比仍然存在着不可忽视的精度差距.Lin 等人[28]认为,两类算法最本质的区别在于,后者通过对候选框的筛选,保证了第 2 阶段训练样本的高质量和类别的均衡,而前者必须在图像上每一个滑动窗口处进行预测,换言之,即存在严重的正负样本不均衡和难易样本不均衡问题.因此,他们为一阶段检测器设计了新的损失函数 Focal Loss.Focal Loss 在交叉熵损失函数的基础上引入了两个新的参数,一个用于降低负样本的权重,另一个用于降低简单样本的权重,让模型在训练时能够避免被一阶段算法存在的大量负样本、简单样本转移注意力.实验测试中,作者采用 ResNet 和特征金字塔网络架构设计了简单的一阶段检测器 RetinaNet,并应用 Focal Loss 进行训练,最终在 MS COCO 测试集上展现出了超越 Faster R-CNN 的精度的能力,尤其体现在小样本的检测上.

YOLOv2 之后,Redmon 等人再次对其进行升级,提出了 YOLOv3[29].YOLOv3 主要有 3 个改进点:(1) 采用多个逻辑回归分类器取代 softmax 分类器,使模型能够适用于类别间存在交集的分类任务;(2) 引入特征金字塔网络架构,对最深层特征图进行两次上采样,分别与浅层特征相融合,最后在 3 个特征层上设置不同的锚点,预测不同尺度的目标;(3) 学习残差网络的思想,设计了 Darknet-53 作为新的骨架网络,在精度上可与 Resnet-101、ResNet152 相匹敌,而且速度更快.YOLOv3 在当时实现了最好的速度与精度的权衡,也是目前工业界目标检测的首选算法之一.

近年来,在“舍弃锚点(anchor-free)”的潮流之下,除了对传统的锚点策略进行反思与改进的算法之外[30−32],还有另外一系列基于关键点检测的一阶段目标检测算法涌现出来.Law 等人提出的 CornerNet[33]通过检测矩形框的左上和右下成对的角点来确定目标.Zhou 等人提出的 CenterNet[34]则更进一步,只检测目标的中心点位置,然后通过回归不同的属性(目标的长宽、深度、方向等),将目标检测、3D 目标检测、姿态估计等多项视觉任务进行了统一.将目标检测问题转换为关键点检测问题之后,就可以使用 Hourglass[35]等下采样系数更小的关键点检测网络以保留更多的空间信息.由此可见,在不同的计算机视觉任务之间进行算法迁移,同样是一个重要的研究方向.

1.2 目标检测的尺度问题

为了对目标的尺度进行量化,通常以目标实例所占面积(即掩膜所占的像素数量)除以所在图像的面积并开方得到的结果作为该目标实例的相对尺度(介于 0~1 之间),简称尺度.因此,不同图像中的目标的相对尺度存在很大差异,或同一张图像中的多个目标的尺寸存在较大差异,这一状况即被称为尺度问题,一直以来都是影响目标检测任务精度的最核心的挑战之一,即便是进入深度学习时代也同样如此.MS COCO 数据集[16]是目前目标检测领域最常用的基准数据集之一,很适用于对算法的多尺度目标检测能力进行综合评估,主要基于以下几点原因.

(1) 数据集包含了 80 种不同类别的目标,覆盖范围广,场景跨度大;

(2) 根据尺度对数据集中的所有目标实例进行排序后,得到的尺度分布曲线如图 2 所示,可见,最小的 10%目标的尺度均小于 0.020 7,最大的 10%目标的尺度都超过了 0.345,尺度跨度极大,因此,COCO 数据集非常考验模型的多尺度目标检测能力;

(3) 在数据集的评价指标中,将面积小于 32×32 的实例视为小目标,大于 96×96 的实例视为大目标,剩下的实例视为中等目标.因此,在评价模型时除了给出整体的准确率、召回率之外,数据集还会分别计算并给出小、中、大目标 3 种情况下的准确率和召回率,这有利于直观地看出模型在面对不同尺度目标时的检测能力.

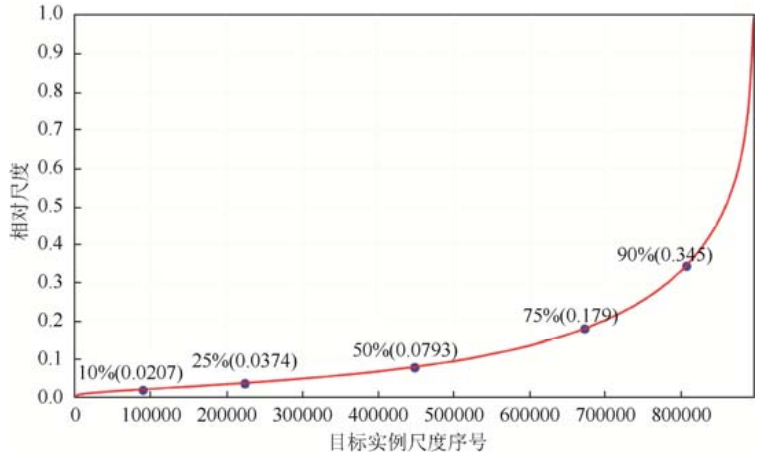


Fig.2 The scale distribution curve of the instances among MS COCO detection dataset

图 2 MS COCO 检测数据集的目标实例的尺度分布曲线

表 1 中列举了一些目标检测算法在 MS COCO 测试集上的检测结果,“++”符号表示模型在推断时使用了图像金字塔.其中,AP 是指当 IoU 阈值分别为 0.50:0.05:0.95 时的平均准确率,AP50 和 AP75 分别是 IoU 阈值为 0.50 和 0.75 的准确率,APS、APM、APL 分别指小、中、大目标的 AP.从表 1 中的数据可以看出,早期的 SSD、YOLOv2、FPN 等检测器的小目标的检测精度都不到中、大目标的精度的一半.近两年的检测器在尺度问题上有所改善,但小目标的精度仍然与中、大目标的精度有着显而易见的差距,这严重影响了整体精度的提高.因此,如何让检测器能够更好地应对不同尺度的目标(尤其是小目标),仍是当今目标检测研究的一个重要难题.

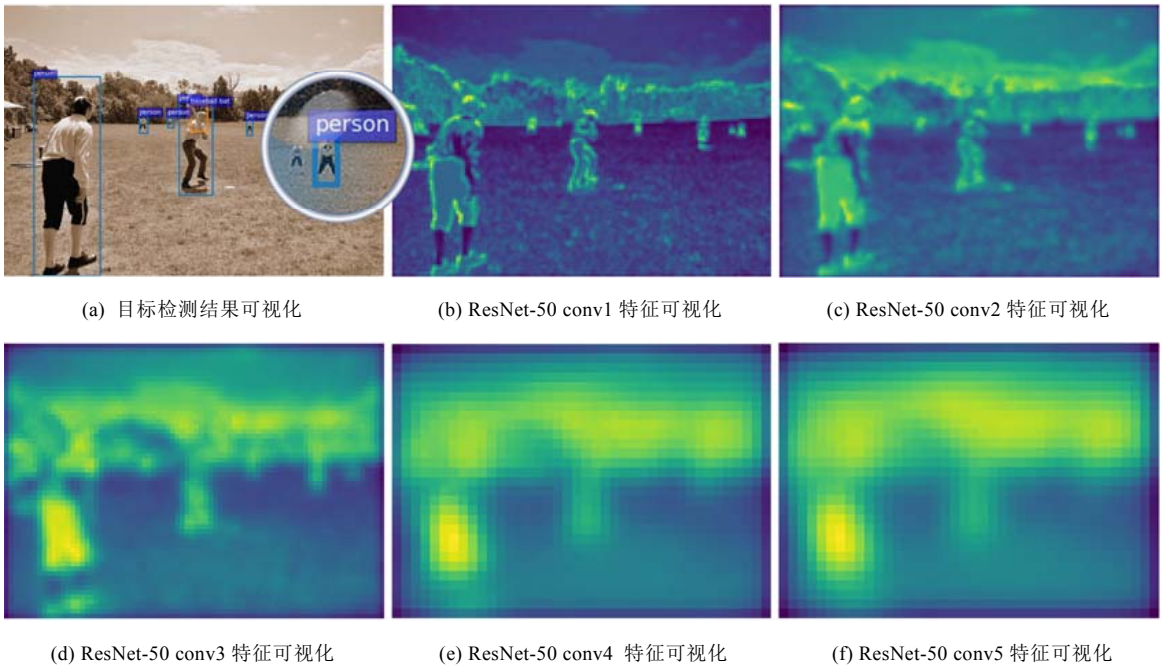
Table 1 Detection performance on the MS COCO TEST-DEV dataset

表 1 目标检测算法在 MS COCO 测试集上的检测性能

算法名称	骨架网络	年份	AP	AP50	AP75	APS	APM	APL
Faster R-CNN[21]	VGGNet-16[5]	2015	21.9	42.7	–	–	–	–
SSD512*[15]	VGGNet-16	2016	28.8	48.5	30.3	10.9	31.8	43.5
Faster R-CNN+++[7]	ResNet-101[7]	2016	34.9	55.7	37.4	15.6	38.7	50.9
R-FCN[22]	ResNet-101	2016	29.9	51.9	-	10.8	32.8	45.0
Faster R-CNN w FPN[23]	ResNet-101	2017	36.2	59.1	39.0	18.2	39.0	48.2
YOLOv2[27]	DarkNet-19	2017	21.6	44.0	19.2	5.0	22.4	35.5
DSSD513[36]	ResNet-101	2017	33.2	53.3	35.2	13.0	35.4	51.1
Mask R-CNN[24]	ResNet-101	2017	38.2	60.3	41.7	20.1	41.1	50.2
RetinaNet500[28]	ResNet-101	2017	34.4	53.1	36.8	14.7	38.5	49.1
RetinaNet800[28]	ResNet-101	2017	39.1	59.1	42.3	21.8	42.7	50.2
Cascade R-CNN[37]	ResNet-101	2018	42.8	62.1	46.3	23.7	45.5	55.2
PANet[38]	ResNeXt-101[39]	2018	47.4	67.2	51.8	30.1	51.7	60.0
YOLOv3[29]	DarkNet-53	2018	33.0	57.9	34.4	18.3	35.4	41.9
Faster R-CNN w SNIP++[40]	ResNet-101-Deformable[41]	2018	44.4	66.2	44.9	27.3	47.4	56.9
Faster R-CNN w SNIPER++[42]	ResNet-101-Deformable	2018	46.1	67.0	51.6	29.6	48.9	58.1
RFB Net512-E[43]	VGGNet-16	2018	34.4	55.7	36.4	17.6	37.0	47.6
PFPNet-R512[44]	VGGNet-16	2018	35.2	57.6	37.9	18.7	38.6	45.9
Faster R-CNN w FPN	DetNet-59[45]	2018	40.3	62.1	43.8	23.6	42.6	50.0
CornerNet[33]	Hourglass-104[35]	2018	40.6	56.4	43.2	19.1	42.8	54.3
SOD-MTGAN[46]	ResNet-101	2018	41.4	63.2	45.4	24.7	44.2	52.6
STDN513[47]	DenseNet-169	2018	31.8	51.0	33.6	14.4	36.1	43.4
DES512[48]	VGGNet-16	2018	32.8	53.2	34.6	13.9	36.0	47.6
DCNv2[49]	ResNet-101-DeformableV2	2019	44.8	66.3	48.8	24.4	48.1	59.6
Grid R-CNN w FPN[50]	ResNet-101	2019	41.5	60.9	44.5	23.3	44.9	53.1
TridentNet[51]	ResNet-101	2019	42.7	63.6	46.5	23.9	46.6	56.6
TridentNet*++[51]	ResNet-101-Deformable	2019	48.4	69.7	53.5	31.8	51.3	60.3
GA-Faster-RCNN w FPN[31]	ResNet-50	2019	39.8	59.2	43.5	21.8	42.6	50.7
FSAF[30]	ResNet-101	2019	40.9	61.5	44.0	24.0	44.2	51.3
FCOS w FPN[32]	ResNet-101	2019	41.5	60.7	45.0	24.4	44.8	51.6
CenterNet[34]	Hourglass-104	2019	42.1	61.1	45.9	24.1	45.5	52.8
YOLOv3@800 w ASFF*[52]	DarkNet-53	2019	43.9	64.1	49.2	27.0	46.6	53.4
Double-Head-Ext[53]	ResNet-101	2020	42.3	62.8	46.3	23.9	44.9	54.3
Faster R-CNN w AugFPN[54]	ResNet-101	2020	41.5	63.9	45.1	23.8	44.7	52.8
ATSS[55]	ResNet-101-Deformable	2020	46.3	64.7	50.4	27.7	49.8	58.4
TSD[56]	ResNet-101	2020	43.2	64.0	46.9	24.0	46.3	55.8
D2Det w FPN[57]	ResNet-101	2020	45.4	64.0	49.5	25.8	48.7	58.1
Dynamic R-CNN[58]	ResNet-101	2020	42.0	60.7	45.9	22.7	44.3	54.3
YOLOv4[59]	CPSDarkNet-53	2020	43.5	65.7	47.3	26.7	46.7	53.3

检测器在面对尺度跨度较大的数据集时会表现不佳的根本原因在于,卷积神经网络在不断加深的过程中,表达抽象特征的能力越来越强,浅层的空间信息也相对丢失.以基于 ResNet-50 的 Faster R-CNN 为例,在检测包含多尺度目标的图像时,可以看到图 3(a)中远处较小的人没有被检测到.为了能直观地进行分析,图 3 对骨架网络 ResNet-50 逐层提取出的特征进行了可视化,可以观察到,浅层网络提取出的特征包含了边缘等空间信息,随着网络的加深,特征逐渐抽象化,空间信息也不断丢失,图 3(a)中没有检测到的那个人的特征早已经无法分辨.此外,由于目标检测任务包含了目标分类和目标定位这两个子任务,因此在检测尺度较大、细节特征丰富的目标时,需要更强的语义信息作为分类依据;在检测尺度较小、偏差容忍度较小的目标时,则需要更细粒度的空间信息以实现精确定位.因此,要解决尺度问题,最常见的思路是构建多尺度的特征表达,常用的方法包括:(1) 输入图像金字塔提取不同尺度的特征;(2) 在网络内部融合不同深度的特征层,构建特征金字塔;(3) 在网络内部设计并行支路,构建空间金字塔.除此以外,也有对锚点策略进行反思和对小目标进行特征重建等其他方法来缓解尺

度问题.下文将从这些方面对已有算法进行概述.本文关于实验结果的陈述,若不明确强调数据集,则默认是在MS COCO 数据集上进行测试得到的结果.



(a) 目标检测结果可视化 (b) ResNet-50 conv1 特征可视化 (c) ResNet-50 conv2 特征可视化

(d) ResNet-50 conv3 特征可视化 (e) ResNet-50 conv4 特征可视化 (f) ResNet-50 conv5 特征可视化

Fig.3 The visualization of detection results and backbone network features of ResNet-50-based Faster R-CNN

图 3 Faster R-CNN 目标检测结果及骨架网络 ResNet-50 特征可视化

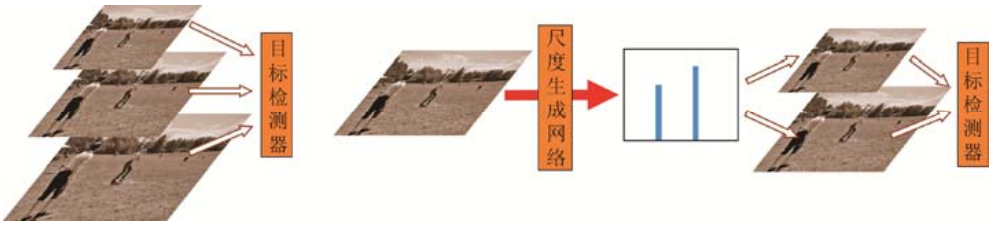
2 基于图像金字塔的多尺度目标检测

同一张图像,在低分辨率下能看到整体的轮廓,在高分辨率下能看清更多的细节,这正是图像金字塔的基本原理.早在目标检测进入深度学习时代之前,图像金字塔就已成为一种通用的提高检测精度的手段,比如用同样大小的滑动窗口在不同尺度的图像上进行特征提取.而神经网络的本质也是特征感知,因此图像金字塔同样适用于神经网络,如下文中图 4(a)所示.在训练阶段,随机输入不同尺度的图像,能够强迫神经网络适应不同尺度的目标检测;在测试阶段,对同一张图像以不同的尺度进行多次检测,最后采用非极大值抑制算法整合所有结果,能够使检测器覆盖尽可能大的尺度范围内的目标.实验结果表明,图像金字塔的引入的确能够在一定程度上提升整体精度,但其弊端也十分明显:高分辨率的图像输入既会增大内存开销,也会增加计算耗时.这不仅会导致训练时难以使用较大的批尺寸,影响了模型精度,同时成倍增加的推断时间还会进一步抬高将算法投入实际应用的门槛.因此,最原始的图像金字塔的实用价值可以说是十分有限的.

2.1 基于尺度生成网络的图像金字塔

Hao 等人[60]在将图像金字塔运用于人脸检测时注意到一个问题:在进行多尺度检测时,金字塔的很多层实际上是没有检测到有效目标的,即存在着明显的资源浪费.其原因在于,每一张图像的目标的尺度分布都存在着显著差别:有的图像可能只有一种尺度的目标,因此实际只需要对金字塔的某一层进行检测;有的图像可能只有中等目标和大目标,因此金字塔里分辨率最高的那一层其实是不需要的,而那恰好是计算开销最大的一层.为了提高检测效率,他们认为:在正式进行目标检测之前,若能先判断图像内目标的尺度分布情况,就能去除图像金字塔中冗余的层,而且在已知目标尺度的情况下还可以对后续检测作进一步的优化.因此,他们设计了一个尺度生成网络,将原本的目标检测任务拆分成尺度估计和单一尺度的目标检测这两步,如图 4(b)所示.尺度生成网络

顾名思义,是用于估计图像里的被检测目标的尺度分布.该网络基于图像级别的监督信号进行训练,输出尺度直方图向量,经过均值滤波和一维的非极大值抑制操作后就能够得到离散的目标尺度分布.由于已知目标尺度,因此后续的检测器只需检测单一尺度的目标,所以可以将 RPN 的锚点的尺寸数缩减为 1,这样就能在不影响精度的前提下进一步提高检测速度.最后,将图像依次采样至目标尺度所对应的分辨率,再轮流进行检测,最后对所有结果进行汇总,就完成了多尺度目标的检测.从实验结果来看,该算法针对 FDDB[61]、AFW[62]和 MALF[63]数据集无论是在速度还是在精度上均超越了原版 RPN.



(a) 图像金字塔 (b) 基于尺度生成网络的图像金字塔

(c) 基于尺度归一化的图像金字塔 (d) 基于注意力机制的图像金字塔

Fig.4 Overview of the image-pyramid-based multi-scale object detection

图 4 基于图像金字塔的多尺度目标检测总览

从本质上讲,尺度生成网络其实是提供了构建图像金字塔的参考意见,让金字塔的层数和每一层的分辨率都更适应于具体图像,有效提高了算法的检测效率.值得深究的一点是,为什么卷积神经网络固定的感受野不适用于多尺度目标检测,但却可以估计出图像中目标的尺度分布呢?作者基于尺度生成网络的响应图,给出的解释是,对于人脸检测,即便是感受野受限,比如只能看清眼睛,也能根据眼睛的尺度相对估计出整张脸的尺度.不过,人脸检测有其特殊性,在面对目标类型更丰富的通用目标检测任务时,该算法的思路是否仍然适用,需要进一步的实验验证.

2.2 基于尺度归一化的图像金字塔

Singh 等人[40]就 MS COCO 数据集中大量的小目标带来的挑战,采用不同的训练策略对 Faster R-CNN 算法做了详细的对照实验,最后测试小目标的检测精度.常规训练策略是以 800×1200 的分辨率进行训练,测试时则采用 1400×2000 的分辨率.以该策略为参照,他们在实验中发现:若将训练集上采样至 1400×2000,最后测试时小目标的精度有所提升但十分微弱,可能是由于本来很大的目标经过上采样后变得更大,干扰了模型对小目标的学习;若同样对训练集上采样,但只让模型检测小目标,超出小目标尺度范围的标签则无视,模型精度却严重下降,这很有可能是因为大量的训练数据的丢失带来了更严重的负面效果;若采用随机多尺度训练策略,精度却几乎没有变化,原因或许是上采样会导致大的目标可能更大、下采样会导致小的目标可能更小,于是带来了更加极端的尺度变化,不利于模型学习.综合这几组实验,他们得出的结论是:极端尺度的目标不利于训练,所有的训练样本都应该参与训练.因此,他们最终提出了名为尺度归一化图像金字塔(简称 SNIP)的训练策略:采用图像金字塔训练模型,但是每一层都只提供合适的尺度范围内的监督信号,如图 4(c)所示.这样做的根本目的是让模型

专注于检测某一尺度范围内的目标,同时又通过金字塔的方式保证所有的训练数据都能够被学习.最后,在验证模型时同样采用图像金字塔.该策略可同时应用于 Faster R-CNN 的两个阶段,并对所有尺度的目标的检测精度带来全方位的提升.可以说,SNIP 本质上是基于 CNN 的固有缺陷对传统的多尺度训练策略的一个改进,将图像金字塔的优势发挥到了极致.不过,该训练策略并未能够解决图像金字塔的内存与时间开销问题.

之后,Singh 等人将 SNIP 升级为 SNIPER[42].为了能够解决图像金字塔在训练时的内存限制问题,SNIPER 不再是对完整的图像进行训练,而是从金字塔的每一层中裁剪出分辨率固定为 512×512 的碎片作为训练单元.其中,在不同层上以碎片大小为网格单元,选择囊括了该尺度下有效目标的网格作为碎片,即为训练时的正样本.而为了防止检测器将背景误判为目标,Singh 等人也将包含了若干假正例的碎片作为负样本,共同参与训练.由于碎片的分辨率较小,从而有效解决了图像金字塔的内存问题,在训练时可以使用更大的批尺寸,这样既加快了训练速度,也提高了模型的检测精度.不过,在实际应用模型检测目标时,仍然必须通入完整的图像金字塔,因此,推断的计算耗时问题还有待解决.

2.3 基于注意力机制的图像金字塔

有一系列文献基于注意力机制的思想,通过引入放大操作,重点关注图的某个区域,以自适应的方式实现多尺度目标检测.最早在深度学习目标检测中引入放大操作的是 Lu 等人提出的 AZ-Net[64].他们认为,RPN 网络的锚点策略本质上是一个固定了滑动窗口大小的穷举算法,效率既不高,对多尺度的目标也不具备适用性.因此,他们设计了一种自适应搜索的候选区域生成算法 AZ-Net.算法以整张图像作为搜索起点,提供邻接区域预测和放大指示器两种输出,前者是指与该搜索区域尺度接近的一系列候选区域,后者是用于指示当前搜索区域内是否存在更小的目标.若存在,则将整张图像分为左上、左下、右上、右下、中间这 5 个区域,依次作为新的搜索起点,直到所有区域都不再包含小目标为止.在 PASCAL VOC 数据集上的实验结果表明,该算法生成的候选区域比 RPN 网络生成的候选区域数量更少但质量更高.不过,精度优势并不明显.其原因有两点:一是算法的自适应优势在单个数据集上体现得不够明显,二是算法的递归搜索操作始终是在原始图像经过 CNN 后提取出的特征图上进行,没有引入更细粒度的特征.不过,该算法仍旧提供了一个很好的解决尺度问题的思路.

Gao 等人[65]延续了 AZ-Net 的搜索的思想,通过引入具有决策能力的强化学习,设计了一个由粗到精的策略来检测高分辨率图像中的目标:首先用一个粗糙的 Fast R-CNN 对下采样后的低分辨率图像进行检测,生成准确率提升概率图,然后利用强化学习找到有可能包含小目标的区域,采用更精细的检测器对高分辨率的这一区域进行目标检测,同时将该区域作为新的算法输入,再次通入粗糙检测器,如此循环,直到不再包含小目标.实验结果表明,在几乎未损失精度的前提下,该算法在 Caltech 行人检测数据集[66]上的像素处理数量减少了 50%、推断时间缩短了 25%,在 YFCC100M 数据集[67]上的像素处理数量减少了 70%、推断时间缩短了 50%.Uzkent 等人[68]延续了 Gao 等人的做法,同样是引入强化学习选择图像中需进一步查看的区域,不过其区别在于,算法还会判断该区域是由大目标主导还是小目标主导,然后分别通过两种不同尺度的检测器进行检测,其目的在于进一步节省计算量.在 xView 遥感图像数据集[69]上所进行的实验结果表明,算法相对于直接在高分辨率原图上进行检测的检测器而言,效率提升了 50%,精度却几乎没有下降.总的来说,这些算法都是源于注意力机制的思想,将多尺度目标检测视为由粗到细、从整体到细节的递归过程(基本流程如图 4(d)所示),并且拓宽了目标检测算法的边界,引出了高分辨率图像、深度强化学习等新的方向.除此以外,这些算法同样可以看作是对图像金字塔的优化:从金字塔的最顶端开始检测,并利用强化学习判断金字塔的下一层中的哪一部分区域存在潜在目标,如此循环,直到下一层不再包括目标为止.所以,算法相当于利用强化学习的决策能力做引导,去除了图像金字塔的冗余部分,解决了 SNIPER 策略中仍然存在的推断时计算耗时这一严重的问题.

3 基于网络内特征金字塔的多尺度目标检测

早期以 R-CNN[8]为代表的检测器直接在神经网络的最后一层特征图上进行预测,由于细粒度空间特征的缺失,对小目标的检测效果不佳,因此需寻求多尺度的特征表示.图像金字塔虽能基于不同分辨率的输入提取不同尺度的特征,但也会带来严重的内存和时间开销问题,不具备适用性.因此,如果能在卷积神经网络内部构建

多尺度的特征表示,就能够在只输入一次图像的情况下近似地得到图像金字塔所能提取的多尺度特征,且计算代价要小得多.现阶段主要通过以下两种方式构建网络内的特征金字塔:(1) 基于跨层连接融合网络内不同深度的特征图,得到不同尺度的特征表示;(2) 基于感受野不同的并行支路,构建空间金字塔.

3.1 基于跨层连接构建特征金字塔

3.1.1 特征金字塔网络

考虑到卷积神经网络层层相叠的结构,越深的特征图,其感受野越大,因此,网络内不同深度的特征图就形成了天然的多尺度表达,于是SSD算法[15]和MS-CNN算法[70]即被提出,可以直接在这些不同尺度的特征图上分别检测目标并最后进行整合,其中,浅层特征图负责检测小目标,深层特征图负责检测大目标.但是,从实验结果来看,小目标的检测精度却并未得到明显改善.究其根本原因在于,这些特征层因为深度各不相同,特征表示能力也各不相同,存在着显著的语义鸿沟.浅层特征层虽然保留了更为细粒度的空间信息,但特征表示能力太弱,缺少有效的语义信息,所以检测效果差.因此,直接在网络内不同深度的特征图上预测不同尺度的目标是不合适的,需要首先构建出每一层都具有足够特征信息的特征金字塔.

针对 SSD 算法存在的缺陷,Lin 等人提出了著名的特征金字塔网络 FPN[23],其基本思想在于结合浅层特征图的细粒度空间信息和深层特征图的语义信息对多尺度的目标进行检测,网络结构如下文图 6(a)所示.算法在RPN网络的基础上,额外增加了一条由上至下的侧路:从最深层特征图开始,经过1×1卷积与上采样之后,与浅层特征层对齐,然后通过对应元素相加的方式融合得到新的特征图,以此类推,将新得到的特征图再进行 1×1 卷积与上采样,就能与更加浅层的特征图相融合.最终,FPN 网络内便成功构建了一个每一层都具备多层特征信息的 5 层特征金字塔.然后,在特征金字塔的每一层上分别预测相应尺度的目标,整合之后便得到了最终的检测结果.原版 RPN 在特征图上每一点处都有 3 个尺度的锚点,FPN 由于特征金字塔每一层只需要预测单一尺度范围的目标,因此所有层均只设置一种尺度的锚点.实验结果表明,该网络结构虽在一定程度上削弱了模型检测大目标的能力,却显著提高了小目标检测的精度,整体性能甚至超越了采用图像金字塔进行推断的 Faster R-CNN,且能够保证 6FPS 的检测速度.由此可见,FPN 在代价较小的前提下,有效提升了检测网络对于多尺度目标的适应性.该网络结构如今已成为目标检测领域在处理多尺度问题时的通用网络架构,而且还被推广到了其他的计算机视觉任务上(例如实例分割).与 FPN 同期诞生的 DSSD 算法[36]也是基于类似的思想融合不同深度的特征,但区别在于 DSSD 是通过反卷积操作提升特征图分辨率,而非双线性插值.

为了对 FPN 提出的特征融合的效果有一个直观的认识,下文图 5 中以热度图的形式展示了采用 FPN 架构的 YOLOv3 网络在处理包含较小尺度的目标的部分图像时,浅层特征在融合深层特征前后的可视化对比情况.可以明显看出,图 5(a)中整张图都是密集的响应,语义信息强和弱的区域没有明显区分性,这也反映了为什么 SSD 算法直接利用浅层特征预测小目标的表现不佳;图 5(b)中,在深层特征的影响下,热度明显聚焦到了图中的小目标上,受背景干扰的影响小了很多,这体现了 FPN 架构对于小目标检测的优势.

3.1.2 改进特征金字塔网络的特征融合

FPN 的核心思想在于融合网络内部的不同深度的特征信息,但是,由上至下逐层融合的结构是值得商榷的,因此出现了一系列对此进行讨论和改进的算法.Liu 等人[38]认为:(1) FPN 虽然将深层语义特征用于构建金字塔的每一层,但是由于网络加深的过程中浅层特征不断丢失,导致金字塔顶端的特征层缺少空间信息,这不利于精确的目标定位;(2) 根据目标的尺度将其分配到金字塔的某一层进行预测,这个策略并非最优,因为没有考虑到其他特征层可能存在的有用信息.因此,他们提出了 PANet,特征金字塔部分的结构如图 6(b)所示.可以看出,PANet 从 FPN 已经构建的特征金字塔的最底层开始,又增加了一条自底向上的特征再融合的侧路,重新构建了一个强化了空间信息的金字塔.然后,将 RoI 对齐操作应用于金字塔的每一层,再将对齐后的特征层通过取最大值来进行融合,最终在融合的特征图上进行检测,就保证了每一个目标的预测都充分利用了所有特征层的信息.实验跟踪了 PANet 的每一个候选区域的融合后特征所对应的原特征金字塔的层数,可以发现,大约有 70%的特征都来自于其他尺度的特征层,只有30%来自于尺度最接近的某一特征层,这证明了特征融合的必要性.最终,该算法取得了当年 MS COCO 实例分割挑战的第 1 名和目标检测挑战的第 2 名.

(a) YOLOv3 浅层特征融合深层特征前的可视化

(b) YOLOv3 浅层特征融合深层特征后的可视化

Fig.5 The comparison of the visualizations of the shallow features of YOLOv3 detector under circumstances whether concatenating deep features or not

图 5 对比 YOLOv3 浅层特征在融合深层特征前后的可视化

Kong 等人[71]认为 FPN 的特征融合本质上相当于特征层的线性组合,难以胜任高度非线性的识别任务.因此,为了让网络能够更灵活地学习每一特征层的关键信息,他们提出了一种新的构建特征金字塔的方式:将不同深度的特征层拼接在一起后,采用 SENet[72]的 SE 模块添加全局注意力,再通过一个残差模块实现局部重构,最后完成新的特征层的构建.如此重复,便构造出了特征金字塔,结构如图 6(c)所示.全局注意力和局部重构这两个轻量级模块可以被任意嵌入到不同网络中,实验中分别将其运用在了 Faster R-CNN 和 SSD 上,均提高了模型的整体性能.

Pang 等人[73]认为金字塔的每一层应该相对平衡地兼顾所有特征层的信息.但是,无论是 FPN 还是 PANet,在构建金字塔时都是通过从上至下或自底向上的侧路逐层传递特征,这个过程势必会导致信息的不断流失,于是金字塔每一层的信息都主要来源于相邻的特征图,因为较远的特征图的信息已经被稀释.所以,他们提出的 Libra R-CNN 将特征金字塔的多层特征统一为中间层尺度后,计算它们的平均特征,然后通过 non-local 模块[74]对特征进行加强,最后将其叠加于原本的特征金字塔的每一层,如图 6(d)所示.实验中,将该架构应用于 FPN 和 PANet 后,模型在 MS COCO 数据集上都提升了 0.9 的精度.

Liu 等人[52]同样对 FPN 进行了反思,认为每一特征层之间存在的不一致性限制了一阶段检测器的性能.例如,某一目标若被分配给了某一特征层进行预测,那么该目标在其他特征层上的相应区域相当于被视为了背景,而这显然是不合理的.部分学者为此将邻接特征层的相关区域的梯度置零[30,31],但这种矫枉过正的做法可能会让邻接特征层的检测质量下降.因此,他们提出了自适应空间特征融合算法(简称 ASFF):将原特征金字塔的所有特征图采样至金字塔中的某一层的尺度,按照一定的权重进行线性组合得到新的特征图,该权重由模型学习得到.有了所有尺度的特征图后,就得到了新的特征金字塔,如图 6(e)所示.Liu 等人从梯度计算的层面证明了该方案构建的金字塔的特征一致性,在实验中以增强版 YOLOv3 作为基准线,证明了 ASFF 模块要优于简单的特征求和或特征连接,并且没有带来显著的计算开销.

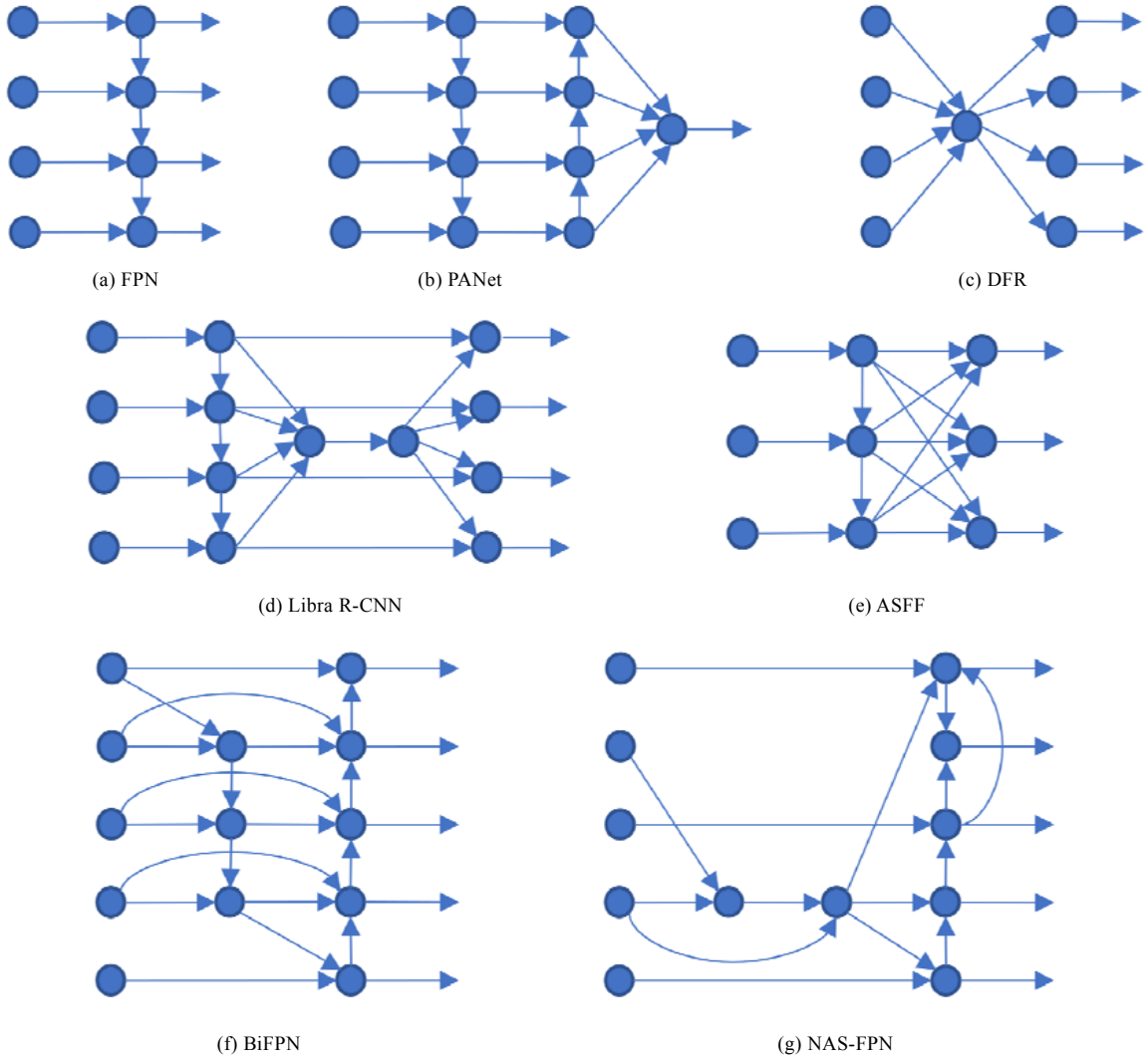


Fig.6 Overview of different ways to construct the in-network feature pyramids through cross-layer connections

图 6 基于跨层连接构建网络内特征金字塔的多种方式

Guo 等人[54]针对 FPN 存在的 3 个缺陷,依次提出解决方案将其升级为了 AugFPN:(1) 不同特征层的语义信息是不一致的,FPN 将它们直接通过 1×1 卷积后相加,势必会削弱多尺度特征的表达能力,因此 AugFPN 在训练时会根据融合前的特征直接进行检测并计算损失,再与网络本身的损失进行加权求和,提供监督信号;(2) FPN 采用从上至下的特征融合强化特征金字塔的语义信息,但是 1×1 的卷积过程会导致最高层特征信息的损失,因此,AugFPN 对原本的最高层特征图进行空间金字塔池化操作,对所有分支进行自适应的特征融合后,再和 1×1 降维后的最高层特征图融合,弥补丢失的信息;(3) FPN 中每一个实例都是根据尺度启发式地选择特征层,但其他层也可能有重要的特征信息,因此,AugFPN 对每一个实例会提取出它在金字塔每一层上的特征,再让网络学习权重参数对这些特征进行求和,最后预测目标.AugFPN 应用在不同骨架网络的 RetinaNet 和 Faster R-CNN 上后,都明显提升了模型的精度.

Tan 等人[75]提出的 EfficientDet 在 PANet 的特征金字塔架构的基础上做出了几点改变:(1) 将只有一个输入的节点移除,理由是未经过特征融合,对多尺度特征的贡献较小;(2) 在同一尺度的输入特征图和输出特征图之

间增加了一条连接,以融合更丰富的特征;(3) 将整个特征金字塔架构进行多次堆叠,使其具有更强大的特征表示能力.最终,得到了 BiFPN 架构,如图 6(f)所示.BiFPN 同样对特征进行加权融合,让网络能学习到不同输入特征的重要性.EfficientDet 以基于神经网络架构搜索的 EfficientNet[76]作为骨架网络,在 MS COCO 数据集上取得了当时最好的精度.

PANet、Libra R-CNN 等方案都是人工设计网络内特征金字塔的构造方式,而在 2019 年,Ghaisi 等人[77]则采用神经网络架构搜索来寻找更好的构造方案:以递归神经网络作为控制器,用强化学习对其进行训练,让它决定每次选择哪两个特征层以怎样的方式进行融合,并以某一分辨率输出,直到填满特征金字塔的每一层.算法最终搜索得到的 NAS-FPN 的结构如图 6(g)所示,在 MS COCO 数据集上相较于原版 FPN 精度提升了 2%~4%不等.Wang 等人[78]同样采用神经网络架构搜索,在 FCOS 的基础上搜索 FPN 结构,与 NAS-FPN 相比,增加了可变卷积,实验中模型性能比原版 FCOS 提升了 1%~2%不等.

3.1.3 改进特征金字塔网络的骨架网络

上述文献都是针对 FPN 提出的特征融合的方式而做出改变,Li 等人[45]则是对 FPN 的骨架网络本身进行了改进.由于多数检测器都采用分类网络作为骨架网络(例如 ResNet),预训练也是在分类数据集上完成,他们认为这带来了两个问题:(1) FPN 等检测器引入了未参与预训练的额外的网络阶段;(2) 骨架网络的感受野和下采样系数均较大,这虽然有利于图像分类,但空间信息的缺失不利于大目标的精确定位,下采样过程中语义信息的丢失不利于小目标的识别,即便是引入了 FPN 架构也没有解决本质问题.为此,他们专门针对检测任务的需求设计了新的骨架网络 DetNet-59,与 ResNet-50 相比,有 3 点主要的区别:(1) 网络与 FPN 有着相同的阶段数量,因此所有阶段都可以参与预训练;(2) 从第 4 阶段开始,DetNet 的下采样系数固定为 16,通道数固定为 256;(3) 在残差模块中引入空洞卷积增加感受野.从实验结果来看,DetNet 的参数量介于 ResNet-50 和 ResNet-101 之间,但在检测任务上的性能表现要优于它们.具体到不同尺度的目标,可以看到,DetNet 尤其擅长定位大目标和寻找小目标,符合作者预期.

3.2 基于并行支路构建特征金字塔

3.2.1 构建空间金字塔

要构建多尺度的特征表达,除了使用图像金字塔或在网络内融合不同深度的特征层构建特征金字塔以外,还有一种方案是在网络内设计参数不同的并行支路,每条支路基于各自的感受野提取不同空间尺度下的特征图,进而构建出了空间金字塔.空间金字塔这一概念最早源于 Lazebnik 等人[79]提出的空间金字塔对齐策略(简称 SPM),是词袋模型[80]的延伸:根据不同的尺度将图像划分成若干子块,然后分别统计每一子块的特征,最后汇总所有特征对图像进行完整表达.在深度学习领域,类似的思路可以追溯到 GoogLenet[6]提出的 Inception 模块,模块内包含了 4 个分支,其中,前 3 个分支分别用 1×1、3×3 和 5×5 的卷积核进行卷积操作,第 4 条分支进行最大池化,最后将所有分支的输出融合,如图 7(a)所示.虽然具体的实现方法有很大差异,但 Inception 模块和 SPM 的思想是一致的,都是为了提取图像在不同空间尺度下的特征.SPP-Net[18]的 SPP 模块同样是采用 SPM 的多尺度分块的方法,对每一分块进行池化操作,即可将任意大小的特征图转换为固定长度的特征向量.总而言之,构建空间金字塔同样是解决目标检测的尺度问题的一个可行方案.

受 SPP-Net 的 SPP 模块的启发,Chen 等人[81]在 DeepLabV2 里设计了类似的 ASPP 模块提取多尺度特征.但不同于 SPP 模块和 Inception 模块的是,ASPP 模块第一次采用空洞卷积[82]构建空间金字塔:4 条支路的卷积核的大小虽均为 3×3,但空洞卷积系数分别为 6、12、18、24,因此感受野也各不相同.在 DeepLabV3[83]中,作者们又发现,随着空洞卷积系数的增大,滤波器的有效权重逐渐变小,最后会退化为 1×1 卷积.因此,他们将 ASPP 模块的空洞卷积系数为 24 的支路修改为了 1×1 卷积,同时新增了一条全局平均池化的支路,如图 7(c)所示.虽然 DeepLab 系列针对的是语义分割任务,但思路同样适用于目标卷积.Liu 等人提出的 RFBNet[43]中设计了包含 3 条支路的 RFB 模块.该模块为了尽可能地模拟人类视觉系统的感受野结构,融合了 Inception 模块和 ASPP 模块的特点,如图 7(b)所示:3 条支路首先分别经过 1×1、3×3 和 5×5 的卷积核滤波,然后再分别经过空洞卷积系数为 1、3、5 的 3×3 空洞卷积,最后融合输出特征图.将 RFB 模块运用于 SSD 网络后,在保证检测速度的前提下,精

度也有较大提升,并明显改善了原 SSD 网络在面对小目标时的不佳表现.

Zhao 等人[84]为了将全局信息和局部信息相结合,设计了类似于 SPP 模块的金字塔池化模块,模块内包含了 4 条分别进行 1×1、2×2、3×3、6×6 池化的分支提取多尺度信息,在语义分割任务上效果有明显提升.Kim 等人提出的 PFPNet[44]同样是出于融合不同尺度的上下文信息的思想,在一阶段检测器里引入了包含 3 条支路的 SPP 模块,不过,每个分支池化得到的特征图还经过了作者设计的 MSCA 模块,分别与另外两个分支的输出特征进行了融合,如图 7(e)所示:将另外两个分支的特征图进行上下采样,然后和主干支路进行特征拼接.最后,在 3 条支路的输出特征图上分别进行目标检测,采用非极大值抑制算法汇总结果.从 MS COCO 数据集的实验结果来看,PFPNet 比使用 FPN 架构的 YOLOv3 还要略胜一筹.

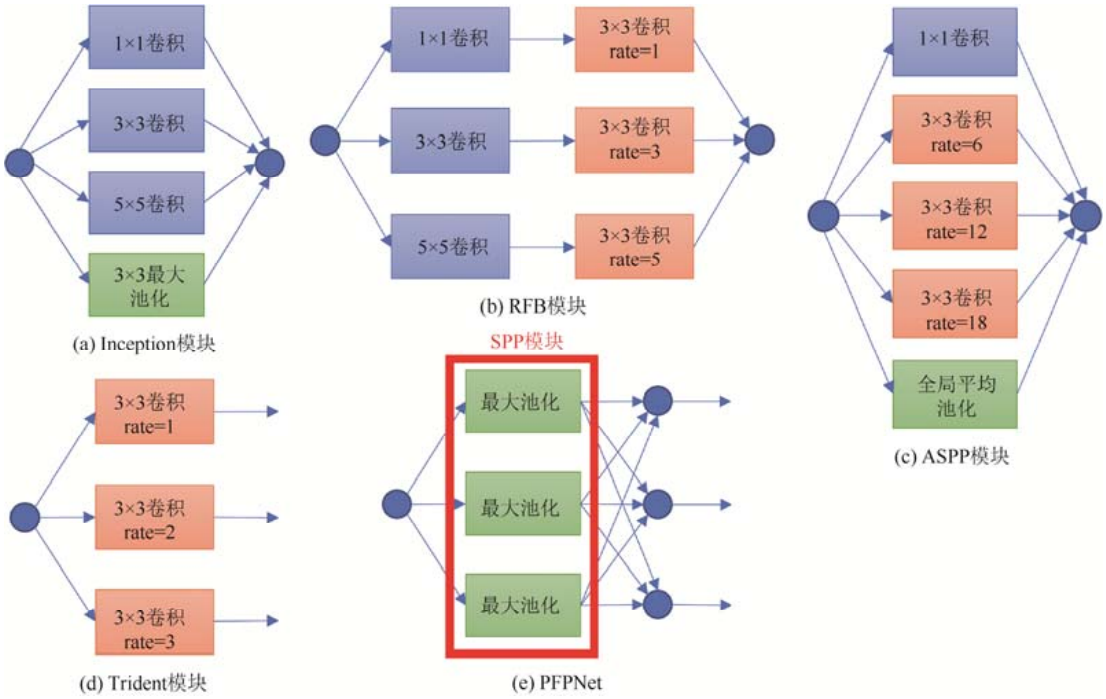


Fig.7 Overview of different ways to construct the in-network feature pyramids through parallel branches

图 7 基于并行支路构建网络内特征金字塔的多种方式

3.2.2 尺度不变特征

SSD[15]、FPN[23]等一系列在网络内部的不同特征图上检测不同尺度目标的算法,本质上是以更小的计算代价近似地构建图像金字塔最终得到的多尺度特征表达.但是,Li 等人[51]指出,网络内部的多尺度特征图的深度各不相同,感知特征的能力也大相径庭,简单的特征融合并不能完全弥补较小的目标所缺失的语义信息,因此势必会牺牲不同尺度目标的特征一致性.而图像金字塔就不存在这个问题,因为任何尺度的输入图像都经过了同种程度的特征提取,所以经过精心设计的 SNIPER 算法[42]取得了更出色的实验结果,当然,计算开销始终是图像金字塔算法绕不过去的坎.如何才能结合图像金字塔和 FPN 的优势呢?Li 等人认为,图像金字塔和构建网络内特征金字塔的共同思路在于,模型在检测不同尺度的目标时需要使用不同大小的感受野才能有更好的表现.因此,他们首次设计了以感受野作为唯一变量的实验:利用空洞卷积,只改变特定卷积层的空洞卷积系数,观察模型在 MS COCO 数据集上的表现.实验结果印证了他们的观点:小、中、大目标分别在空洞卷积系数为 1、2、3 的模型中有着最佳精度.基于这一结果,他们提出了 Trident 模块,基本结构如图 7(d)所示.该模块包括了 3 个核心设计:(1) 模块内有 3 条支路,每条支路的卷积层的空洞卷积系数不同,换言之,即感受野不同;(2) 3 条支路虽然感受野不同,但是权值是共享的;(3) 训练时,采用类似 SNIP[40]的策略,在不同的分支上根据感受野的不同,设置不同

尺度的目标的监督信号.将 Trident 模块运用于 Faster R-CNN 上得到 TridentNet,推断时将 3 个分支的输出进行汇总,最终在 MS COCO 数据集上取得了超越 SNIPER 的结果.可以看出,与 RFB 模块或 PFPNet 相比,Trident 模块最大的特点就在于巧妙地利用了空洞卷积的特性,设计了 3 条支路权重共享这一妙招,可以说是体现了模块的精髓:同样的目标在不同的尺度下,有些特征会发生变化,也有些特征不会随着尺度而改变.权重共享这一设计就相当于在强迫神经网络学习尺度不变的特征,在不同的感受野下检测不同尺度的同一目标,能够得到同样的结果.

尽管 TridentNet 的计算开销赶不上图像金字塔,但多条支路的输出仍然是比较耗时的.为了保证算法的实用性,他们额外设计了训练时在所有分支训练所有目标、但推断时只使用中间分支的 TridentNet Fast,相当于网络内部的多尺度数据增强,在没有引入任何额外计算量的前提下精度相较于基准线提升了 2.7,与原版 TridentNet 的差距也只有 0.6.虽然 TridentNet 的本质在于学习尺度不变的特征,但是,为什么舍弃了 SNIP 训练策略的 TridentNet Fast 也能够取得接近 TridentNet 的实验结果呢?作者给出的猜测是由于权重共享,更具体的原因还有待进一步研究.

4 多尺度目标检测的其他策略

无论是使用图像金字塔,还是在网络内构建特征金字塔,都是利用基于多尺度的特征来解决目标检测的尺度问题.除了这一思路以外,也有学者从检测算法流程中更细节的层面去解决尺度问题,包括锚点、动态卷积、基于生成对抗网络重建特征等.本节将对这些策略一一进行概述.

4.1 锚 点

4.1.1 锚点分布的影响

早期的目标检测为了检测到不同尺度的目标,除了采用固定大小的滑动窗口在图像金字塔上逐层滑动以外,还可以采用不同大小的滑动窗口轮流在同一张图上滑动.Ren 等人[21]提出的 RPN 网络引入的锚点这一概念,同样相当于在骨架网络提取的特征图上设置了 9 个不同大小的滑动窗口(3 种尺度和 3 种长宽比)作为检测的先验信息,以确保网络能够尽可能地覆盖更大尺度范围内的目标.虽然模型对于小目标的检测精度并不理想,但多尺度的锚点策略还是成为了后来多数检测器的标配,结合特征金字塔甚至能够进一步扩大锚点的尺度范围.例如,FPN 就在每一输出特征层都布置了不同尺度的锚点.

也正是考虑到锚点的影响,Ming 等人[85]指出,除了特征的融合以外,FPN 架构有效的根源,可能并非特征金字塔网络架构带来的多尺度的特征表达,而是每一层特征图上不同的锚点分布,它们的数量、尺度分布加起来远远超过了最早的 Faster R-CNN,在过往的分析中忽视了这一因素.因此,Ming 等人基于 WIDER FACE 人脸检测数据集[86],在 FPN 架构的基础上设计了多组实验进行对比,结果见表 2.其中,原版 FPN 为基准线,FPN-finest-stride 表示只利用 FPN 的最后一层特征图进行检测,但是设置和原本的 FPN 所有特征层上同样的数量以及分布的锚点,结果精度相对于原版 FPN 几乎没有下降.这说明,至少对于人脸检测而言,在 FPN 的多层特征图上进行检测可能并不是必要的,锚点的设置才是关键.于是,他们设计了表 2 中的 FPN-finest,即同样是在最后一层特征图上进行预测,但将所有锚点的步长都统一为最小尺度的锚点的步长,结果是:简单和普通目标的精度有所上升,但困难目标的精度却明显下降.鉴于此,对不同模型的锚点分布进行统计后,他们得出新的结论:锚点的尺度分布不均衡和正负分布不均衡对结果有着重要影响.因此,他们提出了锚点的分组采样策略,即表 2 中的 FPN-finest-sampling:将锚点按照尺度分为不同的组别,在计算损失函数时,不再使用所有的数据,而是在每一个组内进行随机采样,实现不同尺度的锚点的平衡.从表 2 中可以看到,即便 FPN-finest-sampling 只利用了 FPN 的最后一层特征图进行检测,应用了分组采样策略后却超越了包括原版 FPN 的所有模型.虽然人脸检测的结论不确定是否能够直接推广到其他数据类型的多尺度检测,但是作者们对于 FPN 结构的反思和对于锚点分布这一因素的探讨,的确值得深入研究.

Table 2 Quantitative comparison of different anchor strategies based on Faster R-CNN's face detection performance on WIDER FACE validation set

表 2 基于 Faster R-CNN 在 WIDER FACE 验证集上的人脸检测性能量化对比不同的锚点策略

不同方法	预测特征层	锚点步长	是否分组采样	All	Easy	Medium	Hard
FPN	P2~P5	{4,8,16,32}	否	82.1	90.9	91.3	87.6
FPN-finest-stride	P2	{4,8,16,32}	否	81.6	90.4	91.0	87.1
FPN-finest	P2	4	否	80.2	94.1	93.0	86.6
FPN-finest-sampling	P2	4	是	82.8	94.7	93.8	88.7

4.1.2 锚点的功与过

多尺度的锚点设置虽然已成为多数检测器面对尺度问题的标配,但是近年来越来越多的学者意识到锚点策略所存在的天然缺陷.例如,以 FPN 为例,Zhu 等人[30]指出训练时会规定目标的尺度大小,这一点决定了哪一特征层负责对它进行检测,然后又根据目标和锚点的交并比决定哪一种尺度的锚点负责回归它的坐标,这些都是人为经验决定的启发式规则,实际上并不是最优的.因此,为了改善这一问题,Zhu 等人基于 RetinaNet 提出了 FSAF 模块.该模块在特征金字塔的每一层都新增了一条无锚点(anchor-free)分支,直接预测当前位置相对于目标矩形框的 4 个边界的距离.两种分支参与联合训练,其中,不基于锚点进行预测的分支会通过在线特征选择的方式进行学习:根据每一条支路的输出为每一个目标选择确定最优(即损失最小)的特征层,并且只为相应特征层提供监督信号.在推断的时候,可以结合基于锚点的分支和舍弃锚点的分支进行预测.从结果来看,单独的舍弃锚点的分支,其预测结果即已超越了原 RetinaNet,而若与基于锚点的分支共同预测,那么就能更好地加以互补,因为从实验结果来看,不基于锚点的分支能够检测出尺度更加极端的目标.在 MS COCO 数据集的测试中,FASF 比 RetinaNet 的整体准确率提升了 1.8%,小目标准确率提升了 2.2%.

同样地,Wang 等人[31]也指出锚点的缺陷:(1) 锚点的尺寸需要预先定义,如果定义得不好会明显降低模型的性能;(2) 为了保证足够的召回率,往往需要大量的锚点,然而其中的大部分锚点都是对检测结果没有帮助的.于是,他们提出,根据图像的特征首先预测出合适的锚点的中心点位置和长宽.然后,根据锚点的尺度采用可变卷积对原始的特征图进行修正,最后基于新的特征图和预测的锚点进行目标检测.实验中,该算法生成的候选区域的小目标召回率比 RPN 高了 9.2%,锚点的数量却减少了 90%,应用在 Faster R-CNN 上能够提升 2.7%的 mAP.Tian 等人[32]为了排除锚点带来的负面影响,提出了 FCOS 算法,类似语义分割那样逐像素点地进行目标检测.他们规定,图像中每个落入了矩形框内的点都属于正样本,并回归该点到矩形框 4 条边界的距离.如果同一个点落入了多个目标的矩形框中,则选择面积较小的框进行标记.同时,为了避免大量远离目标中心的位置产生的低质量矩形框,他们还添加了一个中心度分支,以预测偏离中心点的程度,取值在 0~1 之间.将中心度与分类结果相乘,得到最终结果,就能降低远离中心的矩形框的权重.算法既能直接运用于一阶段检测器,也能用在二阶段检测器的 RPN 阶段.在同样的骨架网络下,FCOS 比 RetinaNet 小目标的准确率高了 2.6%.不过,Zhang 等人[55]在对只使用一个锚点的 RetinaNet 和 FCOS 算法进行仔细的对比研究后认为,基于锚点的算法和舍弃锚点的算法的本质区别在于正负样本的选取,锚点的尺寸反而影响不大.因此,他们提出了自适应训练样本选取策略(简称 ATSS),运用在 RetinaNet 和 FCOS 上都能有稳定的性能提升.Ke 等人[87]认为,现有锚点策略仅通过锚点和目标交并比的大小来分配锚点,导致了分类与定位任务的割裂.因此,他们提出了多锚点学习策略,为每一个目标筛选出交并比较高的多个锚点组成锚点袋,并结合分类和定位的分数来评估锚点袋中的正样本锚点,整体参与迭代训练,以使最终选择的锚点在检测任务上有更好的表现.

4.2 交并比阈值

在目标检测的训练过程中,我们通常是基于预测矩形框和真实标签的交并比来确定正负样本,譬如交并比大于 0.5 的为正样本,小于 0.3 的为负样本.但是,这样的阈值设定主要是基于经验,并不一定是最优选择.而且,采用固定的交并比阈值对于多尺度目标检测来说更加不合适,因为相等的坐标偏差会对小目标的交并比造成更大的影响,对于大目标的影响则微弱得多.为了尝试解决这一问题,Cai 等人[37]提出了 Cascade R-CNN 算法,将 3 个 R-CNN 网络分别设置 0.5、0.6、0.7 的交并比阈值,然后级联在一起.这样做的依据是,如果直接在单个网络

上将交并比阈值提高,会使得正样本数量快速减少,导致网络精度显著下降.因此,Cai 等人便想到了以级联的方式逐步提升生成的矩形框的质量,将前一个检测网络的输出作为后一个检测网络的输入,就能够不断地适应更高的交并比阈值,并且每一个网络都可以检测特定交并比范围内的目标.该算法提出的级联结构对于精度的提升是显著的,不过也明显增加了训练时间和推断时间.同样是考虑到固定的交并比阈值并不合理,Zhang 等人[58]提出的 Dynamic R-CNN 算法则是基于候选框的整体质量,根据一定百分比动态地控制交并比阈值.虽然与 Cascade R-CNN 的思想相似,但该算法没有级联多个检测器,只调整了训练过程,不影响推断过程,实用性更强.

4.3 动态卷积

传统的卷积神经网络存在着一个固有缺陷:卷积核的大小是固定的,池化层的尺度也是固定的,这就导致了网络内所有特征层的感受野始终是固定的,不利于感知不同尺度的目标.因此,便有了一系列方法尝试着将卷积操作动态化.例如,空洞卷积[82]的提出,就是让卷积层能够在参数量不变的情况下,感受野随着空洞卷积系数单调变化,这也使得神经网络能够更方便地捕获多尺度的特征.而 Dai 等人提出的可变卷积[41]则更进一步,对卷积计算的每一个采样点的位置都增加了一个偏置,这样就可以让卷积核呈现出各式各样的形状,空洞卷积相当于是可变卷积的一种特例.同样地,池化层也可以增加偏置,进而被改造为可变池化.从实验的可视化结果来看,可变卷积的确能够帮助神经网络更好地适应不同形状和尺度的目标.不过,Zhu 等人[49]也发现可变卷积因为偏置不可控,引入了过多的可能造成负面影响的上下文信息.因此,他们对可变卷积进行了升级,让其不仅能学习偏置,还能学习到每个采样点的权重,相当于局部的注意力机制.此外,文献[49]还让模型去模仿 R-CNN 算法提取出的特征,以进一步消除冗余的上下文信息的影响,提高了模型性能.总体来说,可变卷积这一设计显著增加了卷积神经网络的自由度,可以很好地与其他检测器兼容.Chen 等人[88]则提出了动态卷积,即根据输入集成多个并行的卷积核,再基于注意力机制对其进行融合,因此具有适应性更强的特征表达能力.该方法应用于轻量级的 MobileNetV3-small[89]后,在 ImageNet 分类数据集上取得了 4%的 Top-1 正确率提升,但是参数量也变为了原模型的 3~4 倍左右,因此,目前而言还难以推广到成熟的检测网络中.

4.4 边界框损失函数

L1 和 L2 范数是经典的回归损失函数,在目标检测任务中可以用于对边界框进行回归.但是 L1 损失函数的收敛速度较慢且解不稳定,L2 损失函数对离群点敏感而不够鲁棒.因此,Girshick[19]提出了平滑 L1 损失函数,结合了两者的特点:相比 L1 损失函数,在靠近真实值时,梯度值足够小,收敛更快;相比 L2 损失函数,离群点的梯度更小,更鲁棒.不过,这 3 种损失函数有两个共同的缺点:(1) 都是对矩形框的顶点坐标和长宽的偏移进行惩罚,无法直接反映预测框与真实框的相似程度;(2) 都不具备尺度不变性.为了解决这一问题,Yu 等人[90]提出了交并比损失函数,将矩形框视为一个整体,直接对比例形式的交并比求对数来指导边界回归,因此,该损失函数就具备了尺度不变性,相比 L2 损失函数,在处理多尺度的目标时有着明显的效果提升.

Rezatofighi 等人[91]对交并比损失函数进行了更加深入的分析,指出了它的不足:(1) 如果预测框与真实框没有相交,那么交并比始终为 0,无法根据两个框的真实距离进行学习;(2) 交并比无法区分矩形框重合的角度,进而无法精确反映重合程度.为此,他们提出了广义交并比(GIoU),在交并比的基础上新增了一个基于预测框和真实框的最小闭包面积的惩罚项.由于该惩罚项受预测框和真实框的距离和角度的影响,因此在保留尺度不变性的前提下解决了上述两个问题.实验中,GIoU 取代 L2 损失函数后,让 YOLOv3 的 AP 和 AP75 分别提升了 1.9%和 2.9%,取代平滑 L1 损失函数后,让 Faster R-CNN 的 AP 和 AP75 分别提升了 0.9%和 1.2%.

Zheng 等人[92]在实验中发现,GIoU 在训练过程中会倾向于先增大预测框与真实框产生交集,然后公式中的交并比项发挥作用,使交集最大化.一旦预测框将真实框完全包围,GIoU 损失函数会退化为普通的交并比损失函数,无法区分相对位置关系.因此他们认为,过于依赖交并比项使得 GIoU 的收敛速度太慢,甚至对于很多先进的检测算法无法很好地收敛.所以他们提出了距离交并比(DIoU)损失函数和完全交并比(CIoU)损失函数.DIoU 在交并比损失函数的基础上增加了预测框和真实框的中心距离的惩罚项,CIoU 在 DIoU 的基础上新增了长宽比相似性的惩罚项.以基于交并比损失函数的 Faster R-CNN 为基准线,他们所做的对比实验的结果见表 3.可以

看到,DIoU 在所有指标上均超越了 GIoU,但 CIoU 的小目标准确率却出现了下滑,说明长宽比这一惩罚项对于小目标的收敛带来了更多的负面效应.

Table 3 Quantitative comparison of different loss functions based on Faster R-CNN's detection performance on the MS COCO TEST-DEV dataset

表 **3** 基于 Faster R-CNN 在 MS COCO 测试集上的检测性能量化对比不同的损失函数

损失函数	AP	AP75	APS	APM	APL
交并比(IoU)损失函数	37.93	40.79	21.58	40.82	50.14
广义交并比(GIoU)损失函数	38.02	41.11	21.45	41.06	50.21
距离交并比(DIoU)损失函数	38.09	41.11	**21.66**	41.18	50.32
完全交并比(CIoU)损失函数	**38.65**	**41.96**	21.32	**41.83**	**51.51**

4.5 解耦分类与定位

目标检测任务包含了目标分类和目标定位两部分,Faster R-CNN 等传统算法在第 2 阶段普遍通过共享的全连接层对候选区域进行特征提取,最后再在两个分支上分别进行分类和回归.但是,这种做法的合理性值得商榷.Song 等人[53]基于热度图分析指出,分类任务的敏感区域为目标的显著性区域,而定位任务的敏感区域则是目标的边界区域,两者在空间上无法对齐.显然,对于多尺度目标检测,随着目标的尺度增大,分类与定位任务在空间上的不对齐问题也会愈加严重.同样地,Wu 等人[56]则从全连接层和卷积层的特性出发,认为前者的空间敏感性使它更适合于进行分类,后者的权重共享的特点使它提取出的特征的空间相关性更强,更适合回归边界,实验结果证明了这一观点.为了解决分类与回归问题潜在的冲突,最直观的思路就是将两个任务进行解耦.

Lu 等人[50]提出 Grid R-CNN,首次将回归分支分离出来,对候选区域特征使用全卷积网络得到概率热度图,再预测边界框的网格点,实现定位.实验结果表明,该方案能够显著提升较高的交并比阈值下的准确率.Wu 等人[56]的解决方案是 Double-Head,在 RoI 对齐层后分为两条支路,一条经过两层全连接层后进行分类,另一条经过两层卷积层后进行回归.而 Song 等人[53]提出的 TSD 模块则对候选区域特征也进行了解耦,即分类和回归分支对原始的候选区域进行了不同的特征变换,使其更适应于不同的任务.Cao 等人[57]考虑到 Grid R-CNN 在固定区域内寻找关键点的策略难以应对尺度较大的目标,因此提出了密集局部回归策略,使用全卷积网络预测候选区域的多个前景子区域的偏移量并求平均.从表 1 中的实验对比可以看到,该算法与 Grid R-CNN 相比在大目标的精度上提升了 5%.

4.6 小目标特征重建

在 MS-CNN 算法[70]中,为了能够更好地检测尺度较小的目标,网络中设计了反卷积层来对特征图进行上采样,有效地减少了内存占用和计算耗时.Zhou 等人提出 STOD 算法[47],以 DenseNet-169[93]作为骨架网络,设计了尺度变换模块,将最后的多个通道的特征图通过平铺展开的方式构造成为分辨率更高、通道数更少的特征图,用来检测小目标.Zhang 等人提出的 DES 算法[48]为了能够加强 SSD 的浅层特征在检测小目标时缺失的语义信息,设计了一个分割模块的分支进行语义分割,将分割得到的特征图作为权重叠加到浅层特征图上,相当于一种注意力机制.从可视化的结果来看,浅层特征图上的无关特征得到了有效的抑制.

除此以外,生成对抗网络的出现,同样为小目标的特征重建提供了新的思路.Li 等人[94]设计了一个 Perceptual GAN 模型,其中的生成器负责学习将小目标的特征重建为与大目标的特征相接近的超分辨率特征,判别器既负责判断生成的特征是否是真实的大目标特征,同时还会将定位精度反馈给生成器,计算感知损失函数,其目的在于判断生成的超分辨率特征是否有助于提升检测精度.算法最终在 Tsinghua-Tencent 100K[95]和 Caltech 数据集[66]上取得了当时的最佳结果.Bai 等人[46]提出了一个端到端的多任务生成对抗网络 MTGAN,其中生成器用于对模糊的低分辨率图像进行超分辨率重建,判别器除了判断生成的超分辨率图像的真假以外,还会完成目标的分类和定位,即检测器的任务.事实上,该算法的一大优势在于能够与任何已有的检测器相结合.在训练时,判别器计算得到的分类和定位的损失也会反向传播给生成器,以便使其能够重建出更多对检测有益的细节.作者们认为,相较于传统的双线性插值,超分辨率网络生成的图像更真实、质量更高,更有利于检测任务.

该算法在 MS COCO 数据集上取得了超越 Mask R-CNN 的结果,而且精度的提升尤其体现在小目标上.

4.7 数据增强

此外,数据增强同样是缓解尺度问题的可行方案,比如 YOLOv2 算法[27]的随机多尺度训练策略.此外,Kisantal 等人[96]以 Mask R-CNN 作为基准线,针对 MS COCO 数据集小目标检测精度较差的问题提出了两种数据增强的手段:(1) 采用过采样策略,解决数据集中包含小目标图片较少的问题;(2) 在同一张图片里,对小目标的分割掩膜进行复制粘贴,使锚点策略能够匹配到更多的小目标正样本中,进而增加小目标在损失函数中的权重.该思路的本质是通过改变训练数据的目标尺度分布,让模型更倾向于感知小目标.从实验结果来看,大目标的检测精度略有下降,小目标的检测精度有所提升.

在目标检测任务中,为了提高检测器的整体性能,通常会采用额外的数据集对模型进行预训练,然后再在正式的数据集上进行微调,亦或是直接让额外的数据集参与联合训练.但是,Yu 等人[97]指出,如果两个数据集的目标尺度差异较大的话,会对模型的性能造成负面影响.因此,他们提出了尺度匹配算法:通过在原数据集里随机采样来确定期望的目标绝对值尺度分布,然后计算该尺度相对于额外数据集的某一张图像的尺度比例,根据比例对整张图像进行放大或缩小,如此循环,直到处理完整个额外数据集.当然,该算法还存在一个缺陷:数据集在经过尺度匹配之后,目标的尺度顺序可能会发生改变,原本的小目标可能被缩小,大目标被放大.因此,作者们也采用了直方图均衡化的思想,设计了单调尺度匹配算法,以保证尺度匹配过程中不会改变尺度顺序.最后的实验结果表明,若在单调尺度匹配后的 MS COCO 数据集上进行预训练,相对于直接在原始的 MS COCO 数据集上进行预训练,得到的模型在包含大量极端小目标的 TinyPerson 数据集[97]上有 5%左右的相对精度提升.可见,该数据增强策略不失为一种构建不同尺度数据集间的桥梁的可行方案.

Chen 等人[98]提出的图像拼接(stitcher)也是一种能够显著提升小目标精度的数据增强手段.类似的思路最早起源于 Yun 等人[99]提出的 CutMix 数据增强策略,即将一张经过裁剪的图像以补丁的形式粘贴到其他图像上作为新的训练数据,如图 8(a)所示.

(a) CutMix 数据增强 (b) Mosaic 数据增强

Fig.8 CutMix and Mosaic data augmentation

图 8 CutMix 与 Mosaic 数据增强

这样做能够提高训练效率、增强模型的鲁棒性.Bochkovskiy 等人[59]在 YOLOv4 中将 CutMix 改进为 Mosaic 数据增强,不再只是将一张图粘贴到另一张图上,而是将 4 张图拼接在一起,如图 8(b)所示,这让目标的上下文信息变得更加复杂.图像拼接属于 Mosaic 数据增强的特殊形式,虽然该想法不是 Chen 等人的首创,但他们从多尺度目标检测的视角赋予了图像拼接新的意义.他们观察到数据集中小目标的分布极不均匀,导致损失函数的计算中小目标的占比非常低,没有提供给网络充足的监督信号.因此,受 SNIP 和 SNIPER 裁剪策略的启发,他们反其道而行之,将图像缩小并拼接在一起,从而能够将中、大目标的尺度缩小,进而更好地权衡训练数据的尺度分布.此外,他们还提出了相应的训练策略:以损失函数中小目标的比重作为图像拼接的反馈信号,若上一次迭代

中小目标的损失比重过低,则下一次迭代就会采用拼接图像来训练.图像拼接的实验对比见表 4,可以看到,模型的性能得到了全方位的提升,其中小目标的精度上涨最为明显.此外,在训练时间加倍的情况下,引入图像拼接的模型能够持续获得稳定的性能提升,而不会轻易过拟合,这应该得益于显著增强的数据多样性.相比 SNIPER,图像拼接数据增强除了实现更简单、不需要经过繁琐的正负碎片制作以外,更难能可贵的是,推断时不需要付出任何额外代价.

Table 4 Quantitative evaluation of Stitcher augmentation and corresponding training strategies based on Faster R-CNN's detection performance on the MS COCO TEST-DEV dataset

表 4 基于 Faster R-CNN 在 MS COCO 测试集上的检测性能量化评估图像拼接数据增强及其训练策略

网络架构	是否使用 Stitcher	训练时间	AP	AP50	AP75	APS	APM	APL
ResNet-50 Faster R-CNN w FPN	否	1 倍	36.7	58.4	39.6	21.1	39.8	48.1
		2 倍	37.7	59.2	41.0	21.6	40.6	49.6
		4 倍	37.3	58.1	40.1	20.3	39.6	50.1
		6 倍	35.6	55.9	38.4	19.8	37.7	47.6
	是	1 倍	38.6	60.5	41.8	24.4	41.9	49.3
		6 倍	40.4	62.5	44.2	26.1	43.1	51.5

5 多尺度目标检测的研究展望

多尺度目标检测一直以来是一个研究难点.结合目前已有的方案,本节总结了一些值得深入探讨的问题,可作为未来研究的方向.

(1) "碎片式"图像金字塔.SNIP[40]策略让 CNN 能够尽最大可能地发挥图像金字塔的潜力,但是训练时的内存问题和推断时的速度问题限制了它的推广.SNIPER[42]通过裁剪图像金字塔的碎片作为训练样本,解决了训练问题,但是推断问题没有得到改善,因为图像碎片的生成是基于已有的目标标签.若要解决这个问题,则需要通过某种不依赖于标签的方法,构建出"碎片式"图像金字塔,即每一层都只保留了同一尺度范围内的目标.用强化学习来选择放大区域或许是一个可行的方案.

(2) 高分辨率图像的多尺度目标检测.在对高分辨率图像进行目标检测时,往往并不缺少小目标的细节信息,而是难以实现精度与计算资源的权衡.由于受到内存、检测速度需求等限制,Faster R-CNN[21]、YOLO[14]等算法都会先将高分辨率图像下采样至某一分辨率,再通入网络进行检测,这就导致了信息的丢失.若采用滑动窗口法实现地毯式检测,整体速度又太慢.Gao 等人[65]提出的用强化学习引导细粒度检测的策略,对于日常设备拍摄的高分辨率图像是有一定效益的.但是,对于细粒度信息更密集的图像是否仍有效(例如无人机航拍)以及能否设计出更简洁的算法,都还有待于进一步加以研究.

(3) 特征金字塔架构下的特征融合.自特征金字塔网络(FPN)[23]诞生之后,有很多新算法在它的基础上改进了特征金字塔的构建方式.但是,PANet[38]、Libra R-CNN[73]和 ASFF[52]等算法都是在 FPN 已构建好的特征金字塔的基础上再次融合特征,构建新的金字塔.这不得不让人提出疑问,究竟是新提出的特征融合方式发挥了作用,还是反复地堆叠特征融合操作才是精度提升的主要原因,例如像 EfficientDet[75]那样多次堆叠 BiFPN.要证明前者,需要在实验中尝试抛弃 FPN.例如,直接将 ASFF 模块运用于原本的特征层上,再看精度的提升是否强于 FPN.但是如果答案是后者,则需考虑:特征融合对目标检测任务的精度提升的上限在何处.

(4) 神经网络架构搜索.近年来兴起的神经网络架构搜索(NAS)方法在目标检测任务上展现出了相当大的潜力:通过 NAS 得到的特征金字塔网络结构[77,78]和骨架网络[100],与以往人工设计的模型相比都展现出了性能上的优势.但是,一方面,只有针对完整的目标检测算法流程进行搜索,才可能得到真正意义上突破现有算法流程的结构,目前的 NAS 仍只是对局部网络架构进行搜索改进,譬如 FPN 式的特征融合对于尺度问题并不见得是最合适的选择.另一方面,过于复杂的拓扑结构并不一定具有普适性,反而更可能在数据集上出现过拟合.因此,如何更好地运用 NAS,仍是一个值得探讨的方向.

(5) 卷积神经网络能否理解尺度概念.特征金字塔网络的出发点是在不同尺度的特征图上检测不同尺度的目标,但是,这也意味着网络实际上是在把不同尺度的目标当作不同的目标在检测,即便它们可能真的就是同一

个目标.因此,卷积神经网络可能并没有真正理解尺度这一概念,只是在依靠庞大的参数量来强行记忆.这可能也正是引入 FPN 架构的算法在检测大目标时精度普遍会有所下降的原因.TridentNet[51]最特别的地方就在于,通过权重共享和尺度归一化的训练策略尝试着让 CNN 用相同的网络参数在不同的感受野下分辨不同尺度的目标,这一设计的本质就是在让 CNN 学习尺度这一概念.但是,TridentNet 和完全舍弃了尺度归一化训练策略的 TridentNet Fast 的性能差距却非常小,说明这一思路的潜力可能还没有被完全挖掘出来.

(6) 锚点的存在价值.早期的 YOLO 和 Densebox[101]算法都没有锚点,而 RPN 网络的提出最早引出了锚点这一概念,相当于为回归任务提供了一个先验知识,该思想被之后的多数检测器所采纳,与 FPN 架构相结合后显著扩大了目标检测的尺度范围.但是,近两年来,却有越来越多的学者重新回归到舍弃固有锚点的思路上进行研究,因为他们认为锚点存在着引入额外超参数、难以回归极端尺度的目标等问题.但是,从形式上看,很多舍弃锚点的算法其实相当于在每一个像素点处都有一个锚点,没有太本质的区别.而从性能上看,这些新提出的算法目前也没有明显超越基于锚点的算法.此外,Zhang 等人[55]提出的正负样本的选取才是这两类算法的本质区别的观点,也提醒我们需要重新审视锚点对于多尺度目标检测的真正影响.

(7) 多尺度目标的交并比.现有的多数目标检测算法在训练过程中,仍然是基于固定的交并比阈值来确定正负样本.尽管已有部分学者[37,58]指出了这样做并非最优解,并尝试通过对阈值进行动态调整来改进算法.但是,这样的改进始终是将所有的目标视为了一个整体来进行同等处理,实际上不同尺度的目标对于交并比的敏感度是不同的.小目标由于自身尺度较小,候选框的交并比通常会更低,在相同的阈值下更难以得到足够的正样本.因此,若能够针对不同尺度的目标设计不同的交并比阈值,亦或是重新定义更加公正的正负样本的选择依据,或许能够进一步提高多尺度目标检测的性能.

(8) 数据集的尺度不均衡.特征金字塔网络等算法很多时候会给我们带来错觉,认为模型在分配了更多资源提升小目标的检测精度后,大目标的精度也会有所下降,是学习资源受限的必然趋势.但是,针对锚点设置而提出的分组采样策略[85],从每个尺度的锚点中随机采样来计算监督信号,保证尺度均衡,训练得到的模型性能却全方位地超越了其他所有的对照组,包括大尺度目标占据主导的模型.图像拼接数据增强策略[98]也出现了类似的现象,训练数据里中、小目标所占的比重增加,但最终得到的模型在检测大目标时精度同样有所提升.这些实验结果提供了一种可能:原本的模型在大目标上有可能已经过拟合了,而小目标则处于欠拟合的状态,因此相对均衡的尺度能够给模型带来全方位的性能提升.

6 总 结

本文以基于深度学习的目标检测为背景,首先对主流算法的成型历史进行了简要回顾,包括 R-CNN 等两阶段检测算法和 YOLO 等一阶段检测算法.然后,本文总结了近几年来提出的众多检测算法在 MS COCO 检测数据集上的表现,并以相应评价指标为依据,指出了目标检测所面临的尺度问题这一巨大挑战,并分析了其根本原因在于目标定位所需要的浅层空间信息和目标分类所需要的深层语义信息的矛盾.

以解决尺度问题为导向,本文对现有的多尺度目标检测策略进行了汇总和归纳.其中,构建多尺度特征表达是最典型且宏观的策略,具体可以分为图像金字塔和网络内特征金字塔.前者将多尺度的图像通入网络,能够稳定提升检测精度,但显著增加的内存开销和计算耗时是主要问题.后者则只需输入原图,根据特征金字塔构建方式的不同可分为跨层连接和并行支路,计算代价比图像金字塔更小.除此以外,本文也从锚点、交并比阈值、动态卷积、边界框损失函数等更细节的层面分析了能够改善尺度问题的策略,更透彻地了解了检测算法流程的许多细节设计的意义.

最后,本文基于上述分析,对多尺度目标检测的研究方向进行了展望.譬如,能否构建“碎片式”图像金字塔解决计算耗时问题、堆叠特征融合操作的上限在何处、卷积神经网络能否理解尺度这一概念以及数据集中是否存在不同尺度目标的过拟合和欠拟合问题等等.这些疑问都值得继续深入探讨.

References:

[1] Lowe DG. Distinctive image features from scale-invariant keypoints. Int'l Journal of Computer Vision, 2004,60(2): 91–110.

[2] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Proc. of the Computer Vision and Pattern Recognition. 2005,1:886–893.

[3] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Proc. of the Neural Information Processing Systems. 2012. 1097–1105.

[4] Deng J, Dong W, Socher R, Li LJ, Li K, Li FF. Imagenet: A large-scale hierarchical image database. In: Proc. of the Computer Vision and Pattern Recognition. 2009. 248–255.

[5] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv Preprint arXiv: 1409.1556, 2014.

[6] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: Proc. of the Computer Vision and Pattern Recognition. 2015. 1–9.

[7] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proc. of the Computer Vision and Pattern Recognition. 2016. 770–778.

[8] Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proc. of the Computer Vision and Pattern Recognition. 2014. 580–587.

[9] Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A. The pascal visual object classes (VoC) challenge. Int'l Journal of Computer Vision, 2010,88(2):303–338.

[10] Felzenszwalb PF, Girshick RB, McAllester D, Ramanan D. Object detection with discriminatively trained part-based models. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2009,32(9):1627–1645.

[11] Jiao L, Zhang F, Liu F, Yang S, Li L, Feng Z, Qu R. A survey of deep learning-based object detection. IEEE Access, 2019,7: 128837–128868.

[12] Wu X, Sahoo D, Hoi SCH. Recent advances in deep learning for object detection. Neurocomputing, 2020.

[13] Liu L, Ouyang W, Wang X, Fieguth P, Chen J, Liu X, Pietikäinen M. Deep learning for generic object detection: A survey. Int'l Journal of Computer Vision, 2020,128(2):261–318.

[14] Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2016. 779–788.

[15] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC. SSD: Single shot multibox detector. In: Proc. of the European Conf. on Computer Vision. 2016. 21–37.

[16] Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft coco: Common objects in context. In: Proc. of the European Conf. on Computer Vision. 2014. 740–755.

[17] Uijlings JRR, Van De Sande KEA, Gevers T, Smeulders AWM. Selective search for object recognition. Int'l Journal of Computer Vision, 2013,104(2):154–171.

[18] He K, Zhang X, Ren S, Sun J. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2015,37(9):1904–1916.

[19] Girshick R. Fast R-CNN. In: Proc. of the Int'l Conf. on Computer Vision. 2015. 1440–1448.

[20] Zitnick CL, Dollár P. Edge boxes: Locating object proposals from edges. In: Proc. of the European Conf. on Computer Vision. 2014. 391–405.

[21] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. In: Proc. of the Neural Information Processing Systems. 2015. 91–99.

[22] Dai J, Li Y, He K, Sun J. R-FCN: Object detection via region-based fully convolutional networks. In: Proc. of the Neural Information Processing Systems. 2016. 379–387.

[23] Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2017. 2117–2125.

[24] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proc. of the Int'l Conf. on Computer Vision. 2017. 2961–2969.

[25] Qin Z, Li Z, Zhang Z, Bao Y, Yu G, Peng Y, Sun J. ThunderNet: Towards real-time generic object detection on mobile devices. In: Proc. of the Int'l Conf. on Computer Vision. 2019. 6718–6727.

[26] Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv Preprint arXiv: 1312.6229, 2013.

[27] Redmon J, Farhadi A. YOLO9000: Better, faster, stronger. In: Proc. of the Computer Vision and Pattern Recognition. 2017. 7263–7271.

[28] Lin TY, Goyal P, Girshick R, He K, Dollar P. Focal loss for dense object detection. In: Proc. of the Int'l Conf. on Computer Vision. 2017. 2980–2988.

[29] Redmon J, Farhadi A. Yolov3: An incremental improvement. arXiv Preprint arXiv: 1804.02767, 2018.

[30] Zhu C, He Y, Savvides M. Feature selective anchor-free module for single-shot object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 840–849.

[31] Wang J, Chen K, Yang S, Loy CC, Lin D. Region proposal by guided anchoring. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 2965–2974.

[32] Tian Z, Shen C, Chen H, He T. FCOS: Fully convolutional one-stage object detection. In: Proc. of the Int'l Conf. on Computer Vision. 2019. 9627–9636.

[33] Law H, Deng J. Cornernet: Detecting objects as paired keypoints. In: Proc. of the European Conf. on Computer Vision. 2018. 734–750.

[34] Zhou X, Wang D, Krähenbühl P. Objects as points. arXiv Preprint arXiv: 1904.07850, 2019.

[35] Newell A, Yang K, Deng J. Stacked hourglass networks for human pose estimation. In: Proc. of the European Conf. on Computer Vision. 2016. 483–499.

[36] Fu CY, Liu W, Ranga A, Tyagi A, Berg AC. DSSD: Deconvolutional single shot detector. arXiv Preprint arXiv: 1701.06659, 2017.

[37] Cai Z, Vasconcelos N. Cascade R-CNN: Delving into high quality object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 6154–6162.

[38] Liu S, Qi L, Qin H, Shi J, Jia J. Path aggregation network for instance segmentation. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 8759–8768.

[39] Xie S, Girshick R, Dollár P, Tu Z, He K. Aggregated residual transformations for deep neural networks. In: Proc. of the Computer Vision and Pattern Recognition. 2017. 1492–1500.

[40] Singh B, Davis LS. An analysis of scale invariance in object detection snip. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 3578–3587.

[41] Dai J, Qi H, Xiong Y, Li Y, Zhang G, Hu H, Wei Y. Deformable convolutional networks. In: Proc. of the Int'l Conference on Computer Vision. 2017. 764–773.

[42] Singh B, Najibi M, Davis LS. SNIPER: Efficient multi-scale training. In: Proc. of the Neural Information Processing Systems. 2018. 9310–9320.

[43] Liu S, Huang D. Receptive field block net for accurate and fast object detection. In: Proc. of the European Conf. on Computer Vision. 2018. 385–400.

[44] Kim SW, Kook HK, Sun JY, Kang MC, Ko SJ. Parallel feature pyramid network for object detection. In: Proc. of the European Conf. on Computer Vision. 2018. 234–250.

[45] Li Z, Peng C, Yu G, Zhang X, Deng Y, Sun J. Detnet: A backbone network for object detection. arXiv Preprint arXiv: 1804.06215, 2018.

[46] Bai Y, Zhang Y, Ding M, Ghanem B. SOD-MTGAN: Small object detection via multi-task generative adversarial network. In: Proc. of the European Conf. on Computer Vision. 2018. 206–221.

[47] Zhou P, Ni B, Geng C, Hu J, Xu Y. Scale-transferrable object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 528–537.

[48] Zhang Z, Qiao S, Xie C, Shen W, Wang Bo, Yuille AL. Single-shot object detection with enriched semantics. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 5813–5821.

[49] Zhu X, Hu H, Lin S, Dai J. Deformable convnets v2: More deformable, better results. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 9308–9316.

[50] Lu X, Li B, Yue Y, Li Q, Yan J. Grid R-CNN. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 7363–7372.

[51] Li Y, Chen Y, Wang N, Zhang Z. Scale-aware trident networks for object detection. In: Proc. of the Int'l Conf. on Computer Vision. 2019. 6054–6063.

[52] Liu S, Huang D, Wang Y. Learning spatial fusion for single-shot object detection. arXiv Preprint arXiv: 1911.09516, 2019.

[53] Song G, Liu Y, Wang X. Revisiting the sibling head in object detector. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 11563–11572.

[54] Guo C, Fan B, Zhang Q, Xiang S, Pan C. AUGFPN: Improving multi-scale feature learning for object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 12595–12604.

[55] Zhang S, Chi C, Yao Y, Lei Z, Li SZ. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 9759–9768.

[56] Wu Y, Chen Y, Yuan L, Liu Z, Wang L, Li H, Fu Y. Rethinking classification and localization for object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 10186–10195.

[57] Cao J, Cholakkal H, Anwer RM, Khan FS, Peng Y, Shao L. D2Det: Towards high quality object detection and instance segmentation. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 11485–11494.

[58] Zhang H, Chang H, Ma B, Wang N, Chen X. Dynamic R-CNN: Towards high quality object detection via dynamic training. arXiv Preprint arXiv: 2004.06002, 2020.

[59] Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv Preprint arXiv: 2004. 10934, 2020.

[60] Hao Z, Liu Y, Qin H, Yan J, Li X, Hu X. Scale-aware face detection. In: Proc. of the Computer Vision and Pattern Recognition. 2017. 6186–6195.

[61] Jain V, Learned-Miller E. FDDB: A benchmark for face detection in unconstrained settings. UMass Amherst Technical Report, 2010,2(4).

[62] Zhu X, Ramanan D. Face detection, pose estimation, and landmark localization in the wild. In: Proc. of the Computer Vision and Pattern Recognition. 2012. 2879–2886.

[63] Yang B, Yan J, Lei Z, Li SZ. Fine-grained evaluation on face detection in the wild. In: Proc. of the Int'l Conf. and Workshops on Automatic Face and Gesture Recognition. 2015,1:1–7.

[64] Lu Y, Javidi T, Lazebnik S. Adaptive object detection using adjacency and zoom prediction. In: Proc. of the Computer Vision and Pattern Recognition. 2016. 2351–2359.

[65] Gao M, Yu R, Li A, Morariu VI, Davis LS. Dynamic zoom-in network for fast object detection in large images. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 6926–6935.

[66] Dollár P, Wojek C, Schiele B, Perona P. Pedestrian detection: A benchmark. In: Proc. of the Computer Vision and Pattern Recognition. IEEE, 2009. 304–311.

[67] Kalkowski S, Schulze C, Dengel A, Borth D. Real-time analysis and visualization of the YFCC100M dataset. In Proc. of the Workshop on Community-Organized Multimodal Mining: Opportunities for Novel Solutions. 2015. 25–30.

[68] Uzkent B, Yeh C, Ermon S. Efficient object detection in large images using deep reinforcement learning. In: Proc. of the Winter Conf. on Applications of Computer Vision. 2020. 1824–1833.

[69] Lam D, Kuzma R, McGee K, Dooley S, Laielli M, Klaric M, Bulatov Y, McCord B. xview: Objects in context in overhead imagery. arXiv Preprint arXiv: 1802.07856, 2018.

[70] Cai Z, Fan Q, Feris RS, Vasconcelos N. A unified multi-scale deep convolutional neural network for fast object detection. In: Proc. of the European Conf on Computer Vision. 2016. 354–370.

[71] Kong T, Sun F, Tan C, Liu H, Huang W. Deep feature pyramid reconfiguration for object detection. In: Proc. of the European Conf. on Computer Vision. 2018. 169–185.

[72] Hu J, Shen L, Sun G. Squeeze-and-excitation networks. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 7132–7141.

[73] Pang J, Chen K, Shi J, Feng H, Ouyang W, Lin D. Libra R-CNN: Towards balanced learning for object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 821–830.

[74] Wang X, Girshick R, Gupta A, He K. Non-local neural networks. In: Proc. of the Computer Vision and Pattern Recognition. 2018. 7794–7803.

[75] Tan M, Pang R, Le QV. Efficientdet: Scalable and efficient object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 10781–10790.

[76] Tan M, Le QV. Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv Preprint arXiv: 1905.11946, 2019.

[77] Ghiasi G, Lin TY, Le QV. NAS-FPN: Learning scalable feature pyramid architecture for object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 7036–7045.

[78] Wang N, Gao Y, Chen H, Wang P, Tian Z, Shen C, Zhang Y. NAS-FCOS: Fast neural architecture search for object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 11943–11951.

[79] Lazebnik S, Schmid C, Ponce J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Proc. of the Computer Vision and Pattern Recognition. 2006,2:2169–2178.

[80] Sivic J, Zisserman A. Video Google: A text retrieval approach to object matching in videos. In: Proc. of the Int'l Conf. on Computer Vision. 2003. 1470–1478.

[81] Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv Preprint arXiv: 1412.7062, 2014.

[82] Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions. arXiv Preprint arXiv: 1511.07122, 2015.

[83] Chen LC, Papandreou G, Schroff F, Adam H. Rethinking atrous convolution for semantic image segmentation. arXiv Preprint arXiv: 1706.05587, 2017.

[84] Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network. In: Proc. of the Computer Vision and Pattern Recognition. 2017. 2881–2890.

[85] Ming X, Wei F, Zhang T, Chen D, Wen F. Group sampling for scale invariant face detection. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 3446–3456.

[86] Yang S, Luo P, Loy CC, Tang X. Wider face: A face detection benchmark. In: Proc. of the Computer Vision and Pattern Recognition. 2016. 5525–5533.

[87] Ke W, Zhang T, Huang Z, Ye Q, Liu J, Huang D. Multiple anchor learning for visual object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 10206–10215.

[88] Chen Y, Dai X, Liu M, Chen D, Yuan L, Liu Z. Dynamic convolution: Attention over convolution kernels. In: Proc. of the Computer Vision and Pattern Recognition. 2020. 11030–11039.

[89] Howard A, Sandler M, Chu G, Chen LC, Chen B, Tan M, Wang W, Zhu Y, Pang R, Vasudevan V, Le QV, Adam H. Searching for mobilenetv3. In: Proc. of the Int'l Conf. on Computer Vision. 2019. 1314–1324.

[90] Yu J, Jiang Y, Wang Z, Cao Z, Huang T. Unitbox: An advanced object detection network. In: Proc. of the ACM Int'l Conf. on Multimedia. 2016. 516–520.

[91] Rezatofighi H, Tsoi N, Gwak JY, Sadeghian A, Reid I, Savarese S. Generalized intersection over union: A metric and a loss for bounding box regression. In: Proc. of the Computer Vision and Pattern Recognition. 2019. 658–666.

[92] Zheng Z, Wang P, Liu W, Li J, Ye R, Ren D. Distance-IoU loss: Faster and better learning for bounding box regression. In: Proc. of the American Association for Artificial Intelligence. 2020. 12993–13000.

[93] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proc. of the Computer Vision and Pattern Recognition. 2017. 4700–4708.

[94] Li J, Liang X, Wei Y, Xu T, Feng J, Yan S. Perceptual generative adversarial networks for small object detection. In: Proc. of the Computer Vision and Pattern Recognition. 2017. 1222–1230.

[95] Zhu Z, Liang D, Zhang S, Huang X, Li B, Hu S. Traffic-sign detection and classification in the wild. In: Proc. of the Computer Vision and Pattern Recognition. 2016. 2110–2118.

[96] Kisantal M, Wojna Z, Murawski J, Naruniec J, Cho K. Augmentation for small object detection. arXiv Preprint arXiv: 1902.07296, 2019.

[97] Yu X, Gong Y, Jiang N, Ye Q, Han Z. Scale match for tiny person detection. In: Proc. of the Winter Conf. on Applications of Computer Vision. 2020. 1257–1265.

[98] Chen Y, Zhang P, Li Z, Li Y, Zhang X, Meng G, Xiang S, Sun J, Jia J. Stitcher: Feedback-driven data provider for object detection. arXiv Preprint arXiv: 2004.12432, 2020.

[99] Yun S, Han D, Oh SJ, Chun S, Choe J, Yoo Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proc. of the Int'l Conf. on Computer Vision. 2019. 6023–6032.

[100] Chen Y, Yang T, Zhang X, Meng G, Xiao X, Sun J. DetNAS: Backbone search for object detection. In: Proc. of the Neural Information Processing Systems. 2019. 6638–6648.

[101] Huang L, Yang Y, Deng Y, Yu Y. Densebox: Unifying landmark localization with end to end object detection. arXiv Preprint arXiv: 1509.04874, 2015.

陈科圻(1997－),男,硕士生,主要研究领域为计算机视觉.

朱志亮(1988－),男,博士,讲师,主要研究领域为图像智能感知与增强,人机交互.

邓小明(1980－),男,博士,副研究员,CCF 高级会员,主要研究领域为计算机视觉,人机交互.

马翠霞(1975－),女,博士,研究员,博士生导师,CCF 高级会员,主要研究领域为人机交互,媒体大数据可视分析.

王宏安(1963－),男,博士,研究员,博士生导师,CCF 高级会员,主要研究领域为自然人机交互,实时智能计算.