计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2021,30(3):1−13 [doi: 10.15888/j.cnki.csa.007839]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于深度学习的目标检测技术综述①

陆　峰[1], 刘华海[2], 黄长缨[2], 杨　艳[1], 谢　禹[4], 刘财喜[3,4]

[1](上海城投环境 (集团) 有限公司, 上海 200331)
[2](上海环境物流有限公司, 上海 200333)
[3](百工汇智 (上海) 工业科技有限公司, 上海 201209)
[4](上海宝信软件股份有限公司, 上海 201203)
通讯作者: 刘财喜, E-mail: caixi.999@163.com

摘　要: 目标检测是计算机视觉领域中的研究热点. 近年来, 目标检测的深度学习算法有突飞猛进的发展. 基于深度学习的目标检测算法大致可分为基于候选区域和基于回归两大类. 基于候选区域的目标检测算法精度高, 但是结构复杂, 检测速度较慢. 而基于回归的目标检测算法结构简单、检测速度快, 在实时目标检测领域有较高的应用价值, 然而检测精度相对略低. 本文总结了基于深度学习的目标检测主流算法, 并分析了相关算法的优缺点和应用场景. 最后根据深度学习的目标检测算法中存在的困难和挑战, 对未来的发展趋势做了思考和展望.

关键词: 深度学习; 目标检测; 计算机视觉; 算法; 结构

引用格式: 陆峰,刘华海,黄长缨,杨艳,谢禹,刘财喜.基于深度学习的目标检测技术综述.计算机系统应用,2021,30(3):1–13. http://www.c-s-a.org.cn/1003-3254/7839.html

Overview on Deep Learning-Based Object Detection

LU Feng[1], LIU Hua-Hai[2], HUANG Chang-Ying[2], YANG Yan[1], XIE Yu[4], LIU Cai-Xi[3,4]

[1](Shanghai Chengtou Environment (Group) Co. Ltd., Shanghai 200331, China)
[2](Shanghai Environmental Logistics Co. Ltd., Shanghai 200333, China)
[3](Bai-Tech (Shanghai) Industrial Technology Co. Ltd., Shanghai 201209, China)
[4](Shanghai Baosight Software Co. Ltd., Shanghai 201203, China)

Abstract: Object detection is a research hotspot in the field of computer vision. In recent years, the deep learning algorithms contributing to object detection has developed by leaps and bounds. Objection detection algorithms based on deep learning can be roughly divided into two categories depending on candidate regions and regression, respectively. The object detection algorithms based on candidate regions have high accuracy, but complex structure and low speed of detection. The object detection algorithms based on regression, contrarily, have simple structure, high speed of detection, and thus more applications in the field of real-time object detection, but its detection is with low accuracy. This paper summarizes the mainstream algorithms of object detection based on deep learning and analyzes the advantages and disadvantages of different algorithms and their applications. Finally, this paper predicts the prospects of deep learning-based object detection algorithms according to the existing challenges.

Key words: deep learning; object detection; computer vision; algorithm; structure

① 基金项目: 国资委企业技术创新和能级提升资本金支持项目 (2017017)
Foundation item: Support Project for Enterprise Technological Innovation and Level up of State-Owned Asset Supervision and Administration Commission of the State Council (2017017)
收稿时间: 2020-07-18; 修改时间: 2020-08-13, 2020-08-25; 采用时间: 2020-09-01; csa 在线出版时间: 2021-03-03

1 引言

目标检测是一种与计算机视觉和图像处理有关的计算机技术, 用于检测数字图像和视频中特定类别的语义对象 (例如人、建筑物或汽车等), 其在视频安防, 自动驾驶, 交通监控, 无人机场景分析和机器人视觉等领域有广阔的应用前景[1,2]. 近年来, 由于卷积神经网络的发展和硬件算力提升, 基于深度学习的目标检测取得了突破性的进展. 目前, 深度学习算法已在计算机视觉的整个领域得到广泛采用, 包括通用目标检测和特定领域目标检测. 大多数最先进的目标检测算法都将深度学习网络用作其骨干网和检测网络, 分别从输入图像 (或视频), 分类和定位中提取特征.

本文将对基于深度学习的主流目标检测算法进行总结和比较. 第 2 节对卷积神经网络的发展做简要概述; 第 3 节对主流卷积神经网络进行分析和比较, 总结算法性能以及优缺点; 第 4、5 节分别对基于候选区域和基于回归的目标检测算法深入分析, 包括网络结构以及创新和改进; 第 6 节对典型的目标检测算法进行比较和总结; 第 7 节对目标检测算法的未来研究方向进行了思考和展望.

2 神经网络的发展

深度学习模型可以看作是为具有深度结构的神经网络. 神经网络的历史可以追溯到 1940 年代[3], 最初的目的是模拟人的大脑系统, 以有原则的方式解决一般的学习问题. 随着 Hinton 等[4] 提出的反向传播算法, 神经网络算法逐渐变得流行起来. 但是, 由于缺乏大规模的训练数据、过度拟合、有限的计算能力以及与其他机器学习工具相比性能的不足等缺点, 到 2000 年, 各学者对神经网络算法的研究趋于冷淡. 自 2006 年以来, 由于语音识别技术的突破, 重新燃起了人们对于深度学习研究的热情[5,6]. 对深度学习的重新重视可以归因于以下几点:

(1) 大规模的带注释的训练数据的出现, 以充分展现其非常大的学习能力.

(2) 快速开发高性能并行计算系统, 例如 GPU 集群.

(3) 网络结构和训练策略设计方面的重大进步. 在自动编码器的指导下进行无监督的预训练, 可以提供良好的初始化. 随着 dropout 技术和数据扩充, 训练中的过度拟合问题得到缓解. 使用批量归一化后, 深层次的神经网络的训练变得简单有效. 同时, 为了提高神经网络的泛化性能, 提出各种不同结构的神经网络. 例如 AlexNet[7]、VGG[8]、GoogLeNet[9] 和 ResNet[10] 等.

卷积神经网络 CNN 是深度学习的最具代表性的模型[11]. CNN 的每一层称为特征图, 输入层的特征图是不同颜色通道 (例如 RGB) 像素强度的 3D 矩阵. 任何内部层的特征图都是感应的多通道图像, 其“像素”可以视为特定特征. 每个神经元都与前一层的一部分相邻神经元相连. 可以在特征图上执行不同类型的转换[12], 例如滤波和池化, 滤波运算将滤波器矩阵 (学习的权重) 与神经元感受野的值进行卷积, 并采用非线性函数 (例如 Sigmoid, ReLU) 以获得最终响应. 池化操作, 诸如最大池化、平均池化和 L2 池化操作[13] 是将接收域的响应汇总为一个值, 以生成更可靠的特征描述. 通过卷积和池化之间的交织, 能够构造初始要素的层次性结构, 最后添加几个全连接层以适应不同的视觉任务. 根据涉及的任务, 添加不同的激活函数, 以获得每个输出神经元的响应. 通过随机梯度下降方法在目标函数 (例如均方误差或交叉熵损失) 上优化整个网络.

与传统方法相比, CNN 的优势可总结如下:

(1) 通过分层多级结构[14,15] 可以学习到从像素到高级语义特征的多级表示, 从而获得输入数据的隐藏信息.

(2) 与传统的浅层模型相比, 更深的网络结构成倍的增加了表达能力.

(3) CNN 的架构为共同优化几个相关任务提供了可能 (例如, Fast R-CNN 将分类和边界框回归结合为一种多任务学习方式)

图 1 为目标检测算法的发展史, 时间轴下方展示了基于深度学习的分类网络的发展历程. 其中标记框内为 one-stage 算法. 可以看出图像分类算法贯穿目标检测算法的始终, 而 two-stage 算法在前期占据主导地位, one-stage 目标检测算法在后期蓬勃发展. 这是因为图像分类算法和 two-stage 目标检测算法中回归分析方法对于 one-stage 目标检测算法的发展都有重要的促进作用. 表 1 比较了不同目标检测算法在 COCO 数据集上的性能, 可以发现随着目标检测技术的发展, 目标检测算法精度和检测速度均得到了大幅提升[16–26].

3 典型的卷积神经网络

3.1 LeNet

手写字体识别模型 LeNet[27] 诞生于 1998 年, 是最早的卷积神经网络之一. 它利用卷积、参数共享、池

化等操作提取特征, 避免了大量的计算成本, 最后再使用全连接神经网络进行分类识别.

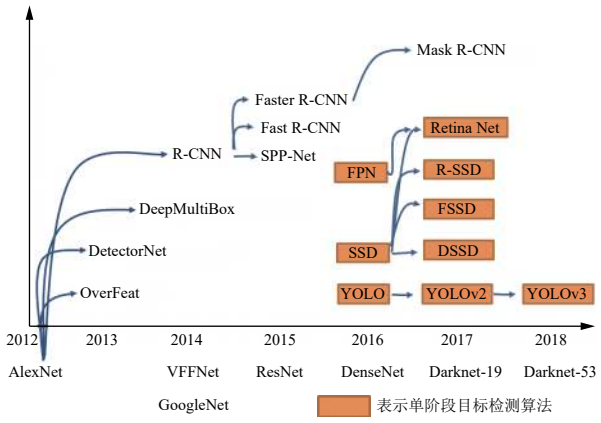

图 1 目标检测算法发展史

LeNet 网络结构除去输入和输出层后, 它有 6 层网络组成, 其中包括 3 个卷积层 (C), 2 个下采样层 (S) 和 1 个全连接层 (F). 其中卷积层采用 5×5 的卷积核, 而下采样层分别采用的平均值池化 (S2) 和最大值池 (S4).

LeNet 特点如下: (1) 使用卷积来提取特征, 组成基本的结构单元: 卷积-池化-非线性激活; (2) 加入非线性激活, 采用 tanh 和 Sigmoid; (3) 池化层使用平均值池化; (4) 分类器使用高斯分类.

3.2 AlexNet

2012 年 Krizhevsky 等[7] 提出的 AlexNet 以 16.4% 的显著优势问鼎 ILSVRC 的冠军, 它第一次采用 ReLU, dropout, GPU 加速等技巧, 参数数量为 6000 万个, 模型大小 240 MB 左右. 其网络结构如图 2 所示, 共 8 层网络结构, 其中 5 个卷积层和 3 个全连接层. 第一个卷积层的卷积为步长为 4, 大小为 11×11; 第二个卷积层的卷积核步长为 1, 大小为 5×5; 其余卷积层的大小均为 3×3, 步长为 1.

表 1 不同目标检测算法性能对比

类型	名称	骨干网络	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Two-stage methods	Fast R-CNN[16]	VGG-16	19.7	35.9	—	—	—	—
	Faster R-CNN[17]	ResNet-101-C4	34.9	55.7	37.4	15.6	38.7	50.9
	Mask R-CNN[18]	ResNet-101-FPN	38.2	60.3	41.7	20.1	41.1	50.2
	Cascade R-CNN[19]	ResNet-101-FPN	42.8	67.3	51.1	29.3	48.8	57.1
	TridentNet[20]	ResNet-101-Deformable	**48.4**	**69.7**	**53.5**	**31.8**	**51.3**	**60.3**
One-stage methods	YOLOv2[21]	DarkNet-19	21.6	44.0	19.2	5.0	22.4	35.5
	SSD513[22]	ResNet-101-SSD	31.2	50.4	33.3	10.2	34.5	49.8
	YOLOv3[23]	DarkNet-53	33.0	57.9	34.4	18.3	35.4	41.9
	RetinaNet[24]	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
	CornerNet[25]	Hourglass-104	42.1	57.8	45.3	20.8	44.8	56.7
	CenterNet[26]	Hourglass-104	**45.1**	**63.9**	**49.3**	**26.6**	**47.1**	**57.7**

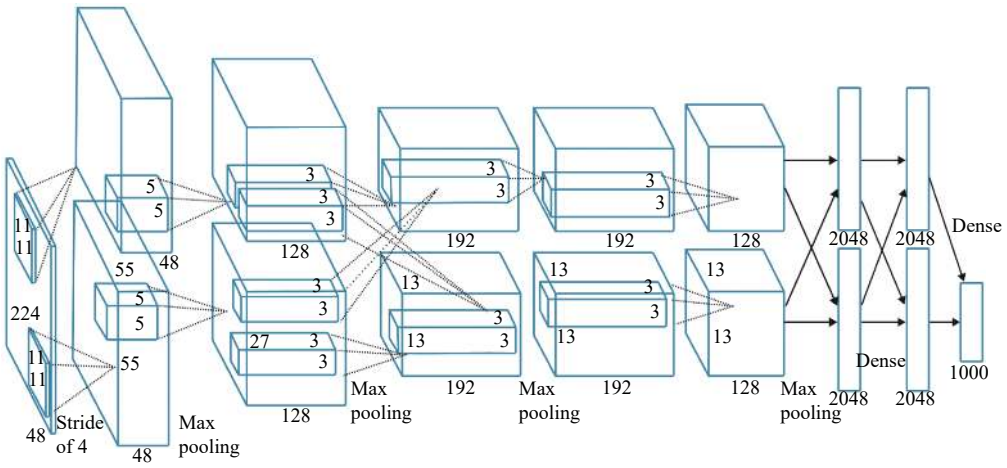

图 2 AlexNet 网络结构图

AlexNet 将 CNN 的基本原理应用到了更深的网络中. 主要应用的新技术如下:

(1) 使用 ReLU 作为 CNN 的激活函数, 成功解决了 Sigmoid 在网络较深时的梯度弥散问题.

(2) 训练时使用 dropout 随机忽略一部分神经元, 以避免模型过拟合.

(3) 池化层使用重叠的最大池化, 避免平均池化的模糊化效果.

(4) 提出 LRN 层, 对局部神经元的活动创建竞争机制, 使得其中响应比较大的值变得相对更大, 并抑制其他反馈较小的神经元, 增强了模型的泛化能力.

(5) 使用 CUDA 加速深度卷积网络的训练, 利用 GPU 强大的并行计算能力, 处理神经网络训练时大量的矩阵运算.

(6) 数据增强. 随机对图片进行镜像, 旋转, 随机噪声等数据增强操作, 大大降低过拟合现象.

3.3 VGGNet

VGGNet[8] 网络结构如图 3 所示, 相比 AlexNet 具有较深的深度, 网络表达能力进一步增强, 同时在 ImageNet 上测试的精度进一步提高, VGGNet 网络优点:

(1) 结构非常简洁, 整个网络都使用了同样大小的卷积核尺寸 (3×3) 和最大池化尺寸 (2×2).

(2) 验证了几个小滤波器 (3×3) 卷积层组合比一个大滤波器 (5×5 或 7×7) 卷积层好. 减少了参数同时得到更多非线性映射, 增加网络表达能力.

(3) 验证了深层次的网络可以获得高级语义特征, 通过不断加深网络结构可以提升网络性能.

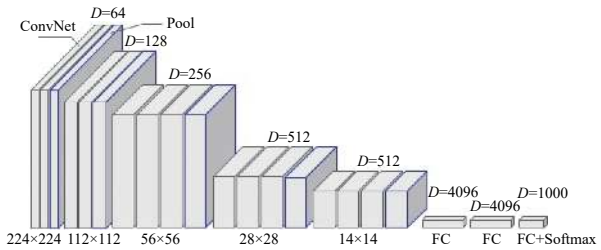

图 3 VGGNet 网络结构图

3.4 GoogLeNet

GoogLeNet[9] 是一个 22 层的深度网络, 在主干卷积环节有 3 个 inception 模块, 模块之间用 3×3 的最大池化层隔开. Inception 模块如图 4 所示, 该结构分 4 条线路并行, 将 CNN 中常用的卷积 (1×1, 3×3, 5×5)、池化操作 (3×3) 堆叠在一起. 采用不同大小的卷积核获得不同大小的感受野, 最后拼接融合不同尺度的特征, 增加了网络对尺度的适应性. 为了解决计算过量的问题, 在后 3 条路线上增加了 1×1 卷积核来进行降维. 同时为了缓解梯度消失的问题, GoogLeNet 增加了两个辅助分类器. 这两个辅助分类器被添加到网络的中间层, 它们和主分类器共享同一套训练数据.

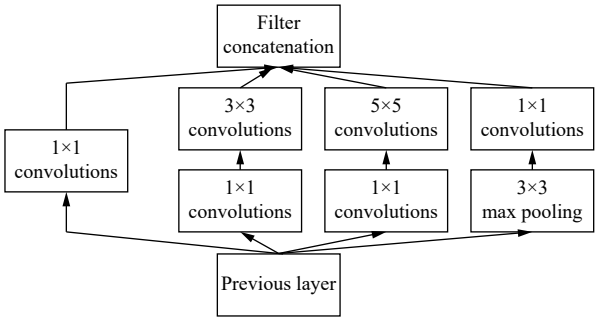

图 4 Inception 网络结构图

GoogleNet 特点:

(1) 采用 1×1 卷积核, 性价比高, 用很少的计算量就可以增加一层的特征变换和非线性变换;

(2) 提出 batch normalization, 把每层神经元的输入值分布回归到均值 0 方差 1 的正太分布, 使其落入激活函数的敏感区, 避免梯度消失, 加快收敛;

(3) 引入 Inception 模块, 4 个分支结合的结构, 每个分支采用 1×1 的卷积核;

(4) 去除了最后的全连接层, 大大减少计算量.

3.5 ResNet

越深的网络有越高等级特征, 拥有强大表达能力. 然而神经网络随着层数的增加, 网络退化严重, 即深层次的网络反而不如稍浅层次的网络性能, 这并非是过拟合导致的. 何凯明等提出的 ResNet[10] 网络很好的解决了这个问题, ResNet 模型的核心是通过建立前面层与后面层之间的“短路连接”, 在浅层网络的基础上叠加 $y=x$ 层, 就是恒等映射 (identity mapping), 可以让网络深度增加而不退化, 有助于训练过程中梯度的反向传播, 从而能训练出更深的 CNN 网络, 实现更高的准确度.

图 5 为残差模块的示意图, 残差函数 $F(x)=H(x)-x$, 如果 $F(x)=0$, 即为恒等映射, 这样学习训练过程相当于自主确定了多少层次的网络是最优的.

ResNet 特点:

(1) 通过残差模块将网络深度进一步提高, 解决了不断深化神经网络而使准确率饱和的问题. 通过活用 1×1 卷积降低了特征图维度, 控制了参数数量.

(2) 存在明显层级, 特征图个数层层递进, 保证输出特征的表达能力.

(3) 使用较少池化层, 大量采用下采样, 提高传播效率.

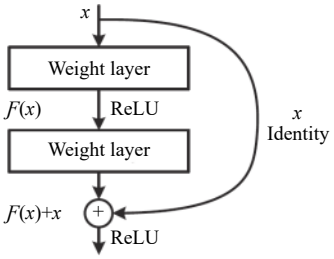

图 5 残差网络结构图

3.6 DenseNet

DenseNet[28] 模型的基本思路与 ResNet 一致, 通过短路连接来融合前后几层的信息, 但是 DenseNet 建立前面所有层与后面层的密集连接, 同时通过特征上的连接来实现特征重用. 这些特点让 DenseNet 在参数和计算成本更少的情形下实现比 ResNet 更优的性能, DenseNet 也因此斩获 CVPR 2017 的最佳论文奖.

DenseNet 网络结构主要结构分稠密块 (Dense blocks) 和过渡层 (transition layers). 稠密块结构能够互相连接所有的层, 具体就是每个层都会接受其前面所有层作为其额外的输入. 对于一个 L 层的网络, DesNet 共包含 $L(L+1)/2$ 个连接, 可以实现特征重用, 提升效率, 并且缓解了深层网络的梯度消失问题. 过渡层为相邻 2 个稠密块的中间部分, 稠密块是连接两个相邻的稠密块, 并且通过池化使特征图大小降低.

DenseNet 的优势主要体现在以下几个方面:

(1) 由于密集连接方式, DenseNet 提升了梯度的反向传播, 使得网络更容易训练. 由于每层可以直达最后的误差信号, 实现了隐式的深度监督;

(2) 参数更小且计算更高效, 由于 DenseNet 是通过拼接特征来实现短路连接, 实现特征重用;

(3) 由于特征复用, 最后的分类器使用了低级特征.

4 基于候选区域的目标检测算法

基于候选区域的目标检测算法, 即 two-stage 目标检测算法, 该方法先提取到目标的候选区域, 然后再由神经网络做分类和回归. 本节将就 two-stage 目标检测主流算法做简要介绍.

4.1 R-CNN

Girshck 等针对卷积神经网络如何实现目标定位的问题, 提出了将于 Region proposal 和 CNN 结合的算法 R-CNN[14], 开创了神经网络实现目标检测的先河, 其计算流程如图 6 所示, 可分为 4 步, 首先利用选择搜索算法提取候选区域, 接着将候选区域缩放到固定大小, 然后进入卷积神经网络提取特征, 随后将提取的特征向量送入 SVM 分类器得到候选区域目标的类别信息, 送入全连接网络进行回归得到位置信息.

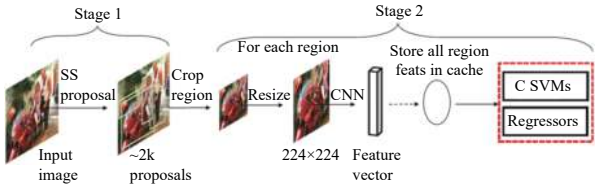

图 6 R-CNN 网络结构示意图

R-CNN 缺点:

(1) 输入的图像大小会进行缩放, 导致图像失真;

(2) 用选择性搜索算法提取约 2000 个候选区域, 每个候选区域都要送入卷积神经网络提取特征, 计算量大耗时多;

(3) 训练测试复杂, 候选区域提取、特征获、分类和回归都是单独运行, 中间数据也是单独保存卷积提取的特征需单独存储, 占用大量硬盘空间.

4.2 SPP-Net

2014 年 He 等针对 R-CNN 的缺陷进行了改进, 开发出了 SPP-Net (Spatial Pyramid Pooling Network)[29]. SPP-Net 将整个图片送入卷积提取特征, 避免了候选区域分别送入卷积层提取特征造成的重复计算. 再次, 在卷积层和全连接层之间增添了空间金字塔池化层 (Spatial Pyramid Pooling, SPP), 可以对不同大小特征图进行池化操作并生成特定大小的特征图, 避免了 R-CNN 对图像进行缩放导致图像失真.

SPP-Net 缺点:

(1) 与 R-CNN 设计相同, 训练各个阶段都是单独运行, 且中间数据必须保存.

(2) 训练数据的图像尺寸大小不一致, 使候选框感受野过大, 不可以使用反向传播有效的更新权重.

(3) 网络微调只更新了全连接层, 不能实现端到端的检测且检测精度还需进一步提升.

4.3 Fast R-CNN

借鉴空间金字塔池化层的思路, Ross 等 2015 年提出 Fast R-CNN[16], 用感兴趣池化层 (Region of Interest, RoI) 代替了空间金字塔池化层, 它去掉了 SPP 的多尺度池化, 直接用网格将每个候选区域均匀分若干个区域块, 同时对每个块进行最大值池化, 从而将特征图上大小不一的候选区域转变为大小统一的特征向量. 针对在训练期间为多阶段和特征重复计算造成的时间代价以及数据存储过量的问题, 将神经网络与 SVM 分类相结合, 由全连接层同时负责分类和回归任务, 实现了多任务端到端训练, 使检测精度和速度同时得到了提高. Fast R-CNN 网络流程如图 7 所示, 主要分 3 部分, 首先将图像送入卷积网络提取特征, 然后通过感兴趣池化层将候选区域池化为同一长度的特征向量, 最后通过全连接层进行分类和回归. Fast R-CNN 在 PASCAL VOC 数据集中检测时间为每张 0.32 s, 远小于 R-CNN 的 45 s 和 SPP-Net 的 2.3 s.

虽然 Fast R-CNN 实现了多任务端到端的训练, 然而通过选择性搜索算法提取候选区域耗费了较多时间, 训练和预测所需时间仍较长, 不能实现实时性检测.

4.4 Faster R-CNN

Ren 等在 Fast R-CNN 的基础上提出 Faster R-CNN[17] 算法, 在卷积层后添加了区域提取网络 RPN (Region Proposal Network), 代替了 Fast R-CNN 的选择性搜索算法. RPN 核心思想是使用 CNN 卷积神经网络直接产生候选区域, 锚框机制和边框回归可以得到多尺度多长宽比的候选区域. RPN 网络也是全卷积网络 (Fully-Convolutional Network, FCN), 可以针对生成检测建议框的任务端到端训练, 能够同时预测出目标物体的边界和分数.

Faster R-CNN 整个流程如图 8 所示, 先对图像进行卷积提取特征, 然后进入 RPN 层得到候选区域, 最后全连接层进行分类和回归. 整个流程从图像特征提取、候选区域获得到分类和回归都在神经网络中进行, 且整个网络流程都能共享卷积神经网络提取的特征信息, 提高了算法的速度和准确率, 从而实现了两阶段模型的深度. Faster R-CNN 在 PASCAL VOC 2007 和 2012 上的 mAP 分别为 73.2% 和 70.4%, 检测速度达到 5 fps.

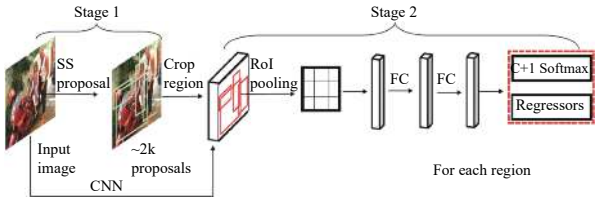

图 7 Fast R-CNN 网络流程图

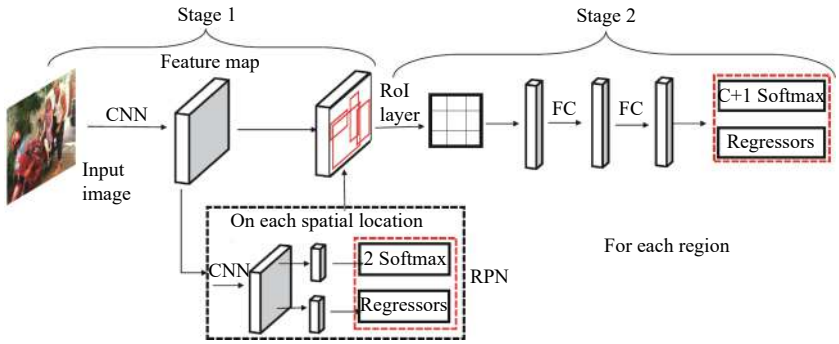

图 8 Faster R-CNN 网络流程图

Faster R-CNN 虽然大幅提高了算法精度和速度, 但仍存在一些缺点: (1) 获取候选区域, 再对每个候选区域分类计算量还是比较大; (2) 虽然速度有了提高, 但还是没达到实时性检测的要求.

4.5 Mask R-CNN

2017 年 He 等在 Faster R-CNN 的基础上再次改进, 提出了 Mask R-CNN[18] 算法, 通过添加 Mask 分支, 能够同时实现目标检测和语义分割任务. 由于 Faster R-CNN 在下采样和感兴趣池化层都采取了取整运算, 对检测任务产生了影响, 特别是对于像素级检测的任务. 通过 RoI align 层替换 RoI Pooling 层, 使用双线性插值来确定非整数位置的像素, 使得每个感受野取得的特

征能更好对齐原图感受野区域. 此外 Mask R-CNN 采用基础网络 ResNet+FPN (Feature Pyramid Network) 来提取图像特征, 如图 9 所示, 在 COCO 数据集上的检测准确率从 Fast R-CNN 的 19.7% 提高至 39.8%.

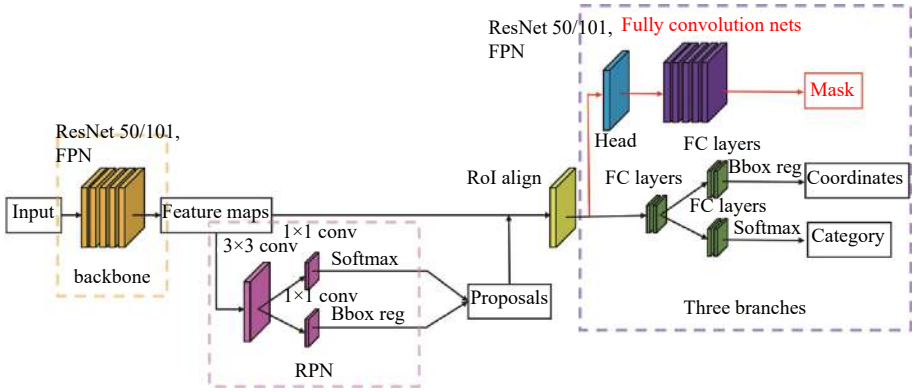

图 9 Mask R-CNN 网络流程图

Mask R-CNN 网络的优点在于: 它在 Faster R-CNN 网络的基础上增加了目标掩码作为输出量, 掩码是将一个对象的空间布局进行编码, 从而使得对目标的空间布局提取更精细. 其缺点在于: 分割分支增加了计算量, 导致 Mask R-CNN 比 Faster R-CNN 检测速度慢一些.

4.6 Cascade R-CNN

R-CNN 系列算法在检测目标时均存在 IoU 阈值选取问题, 阈值选取越高就越容易得到高质量的样本, 但一味选取高阈值会引发两个问题: (1) 样本减少引发的过拟合; (2) 在推理阶段出现检测器最优的阈值与输入候选的 IOU 值发生不匹配. 因此, 单一检测器很难抉择一个合适的 IOU, 无论高低都是有很大弊端.

针对此类问题 Cai 等[19] 提出了一种级联检测器, 他们利用前一个检测器输出作为下一个检测器的输入, 同时相应的提高训练时的 IoU 阈值, 使得 IoU 阈值与预选框的 IoU 值较为接近, 训练的 3 个检测器最后输出结果精度更高, 从而在推理阶段合理的提高了检测器优选 IoU 阈值.

Cascade R-CNN 算法核心由一系列的检测模型组成, 每个检测模型都基于不同 IoU 阈值的正负样本训练得到, 通过重采样改变了不同阶段的输入假设分布, 并且保证了每个阶段有足够固定数量的正例样本数. 通过调整阈值的方式重采样, 不断改变候选框的分布, 在保证样本数不减少的情况下训练出高质量的检测器, 从而不会出现过拟合, 且在更深层的训练阶段可以得到更高的 IoU 阈值.

4.7 TridentNet

目标检测算法对图像多尺度特征的提取主要是采用骨干网络, 影响骨干网络进行多尺度特征提取的因素有: 网络下采样率、网络深度和感受野. R-CNN 系列算法中大部分学者研究了下采样率和网络深度对算法精度的影响, 而对于目标检测中感受野的影响很少有人研究. Li 等[20] 首次提出了 TridentNet 算法, 通过从感受野的角度来构造多尺度目标的特征图, 对于获取不同尺寸目标的特征过程中结构与权重相同, 促进不同尺寸的目标能够形成相似特征. 实验证明, 不同的感受野对不同尺度目标的检测有着不同的影响, 大的感受野对尺度较大的目标检测性能较好, 小的感受野对小目标的检测性能较好.

TridentNet 算法针对多尺度特征提取做了以下两点创新:

(1) 多分支结构, 在最后一层增加了多分支卷积层, 用来提取不同尺度目标的特征图;

(2) 不同分支之间结构相同, 权重共享, 唯一的不同在于不同分支所对应的感受野不一样 (使用空洞卷积来实现).

上述的设置既能够保证为不同尺寸目标提取出特征, 同时还能促进不同尺寸目标生成相似特征, 即结构相同, 权重共享. 文献 [20] 通过实验验证采用 3 个不同卷积空洞率时算法的性能最佳, 通过与现有算法 (算法都采用相同的骨干网络 ResNet-101) 的性能比较, 采用多分支不同空洞率和可行变卷积的 TridentNet 算法性能明显提高, 表明不同感受野的应用有助于算法对不

同目标的检测.

从 R-CNN、SPP Net、Fast R-CNN、Faster R-CNN、Mask R-CNN、Cascade R-CNN 和 TridentNet 算法网络逐步优化, 每个算法均解决一部分难题, 具体来说:

(1) RCNN 解决了使用 CNN 进行目标定位问题;

(2) Fast R-CNN 解决了目标定位和分类同步问题;

(3) Faster R-CNN 解决了选择性搜索目标问题;

(4) Mask R-CNN 解决了同时进行目标定位、分类和分割问题;

(5) Cascade R-CNN 解决了 IoU 阈值选取问题;

(6) TridentNet 解决了从感受野提取图像特征问题.

5 基于回归的目标检测算法

5.1 YOLO 系列

2015 年 Redmon 等提出了基于回归的目标检测算法 YOLO (You Only Look Once)[30], 其直接使用一个卷积神经网络来实现整个检测过程, 创造性的将候选区和对象识别两个阶段合二为一, 采用了预定义的候选区 (并不是 Faster R-CNN 所采用的 Anchor), 将图片划分为 $S\times S$ 个网格, 每个网格允许预测出 2 个边框. 对于每个网格, YOLO 都会预测出 B 个边界框, 而每个边界框 YOLO 都会预测出 5 个值, 其中 4 个代表边界框的位置, 还有一个代表框的置信值.

YOLO 的网络结构示意图如图 10 所示, 其中, 卷积层用来提取特征, 全连接层用来进行分类和预测. 网络结构是受 GoogLeNet 的启发, 把 GoogLeNet 的 inception 层替换成 1×1 和 3×3 的卷积. 最终, 整个网络包括 24 个卷积层和 2 个全连接层, 其中卷积层的前 20 层是修改后的 GoogLeNet. 网络经过最后一个 FC 层得到一个 1470×1 的输出, 7×7×30 的一个张量, 即最终每个网格都有一个 30 维的输出, 代表预测结果.

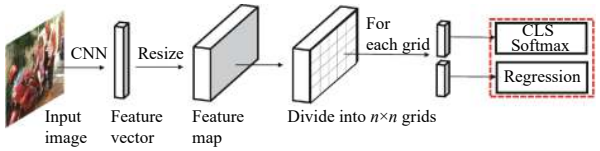

图 10 YOLO 网络结构图

YOLO 优点:

(1) 将目标检测问题转化为一个回归问题求解. 结构非常简单, 直接使用一个卷积神经网络实现同时预测边界框的位置和类别;

(2) 速度非常快, 可以实现视频的实时检测;

(3) 泛化能力强, 能学习到高度泛化的特征, 可以迁移到其他领域.

Redmon 等对 YOLO 网络结构做修改提出了 YOLOv2 方法[21], YOLOv2 用 DarkNet-19 用做基础网络, 包含 19 个卷积层、5 个最大值池化层. YOLOv2 网络通过在每一个卷积层后添加批量归一化层 (batch normalization), 同时不再使用 dropout. YOLOv2 引入了锚框 (anchor boxes) 概念, 提高了网络召回率, YOLOv1 只有 98 个边界框, YOLOv2 可以达到 1000 多个. 网络中去除了全连接层, 网络仅由卷积层和池化层构成, 保留一定空间结构信息. 结果 mAP 由 69.5% 下降到 69.2%, 下降了 0.3%, 召回率由 81% 提升到 88%, 提升 7%. 尽管 mAP 略微下降, 但召回率的上升意味着模型有更大的提升空间. 同时利用 K-means 聚类, 解决了 anchor boxes 的尺寸选择问题.

YOLOv3[23] 借鉴了 ResNet 的残差结构, 使主干网络变得更深 (从 v2 的 DarkNet-19 上升到 v3 的 DarkNet-53). 整个 YOLOv3 结构里面, 没有池化层和全连接层, 在前向传播过程中, 张量的尺寸变换是通过改变卷积核的步长来实现. 相应改进使 YOLOv3 与 SSD 相当的精确度下达到 50 fps 的检测速度, 并在 COCO 测试数据上 mAP 的最佳效果达到 33.0%, 与 RetinaNet 的结果相近, 速度快了 3 倍, 但整体模型变得更加复杂, 速度和精度相互制衡.

YOLOv3 改进之处:

(1) 多尺度预测, 借鉴 FPN, 采用多尺度来对不同大小的目标进行检测.

(2) 更好的分类网络, 从 DarkNet-19 到 DarkNet-53.

(3) 采用 Logistic 对目标进行分类, 替换之前用 Softmax 的分类方法, 且 Softmax 可被独立的多个 Logistic 分类器替代, 准确率不会下降.

5.2 SSD 系列

Liu 等提出的 SSD (Single Shot multibox Detector) 方法[22] 是对 YOLO 算法的改进, 其网络结构如图 11 所示. SSD 与 YOLO 主要不同在于以下几个方面:

(1) 采用多尺度特征图用于检测. SSD 使用 VGG16 作为主干网络, 并在 VGG16 的基础上添加了新的卷积层以获得不同大小的特征图, 较大的特征图用来检测小目标, 较小的特征图用来检测大目标.

(2) 采用卷积进行检测. YOLO 最后采用全连接, 而 SSD 直接采用卷积对不同的特征图进行提取特征. 对于形状为 $m\times n\times p$ 特征图, 只需要采用 $3\times3\times p$ 这样较小的卷积核得到检测值.

(3) 设置先验框. YOLO 中每个单元预测多个边界框, 但是都是相对于这个单元本身, YOLO 需要在训练过程中自适应目标的形状. SSD 借鉴了 Faster-RCNN 的 anchor 理念, 每个单元设置尺度或者长宽比不同的先验框, 减小了训练的难度, 对重叠或近邻的物体有更好的预测效果.

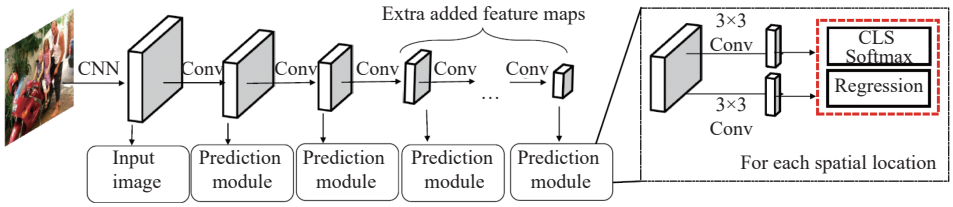

图 11 SSD 网络结构图

SSD 算法检测速度较快, 整个过程只需要一步. 首先在图片不同位置按照不同尺度和宽高比进行密集抽样, 然后利用 CNN 提取特征后直接进行分类与回归, 速度较快, 但均匀密集采样会造成正负样本不均衡使得训练比较困难, 导致模型准确度有所降低. SSD 对小目标的检测没有大目标好, 因为随着网络的加深, 在高层特征图中小目标的信息丢失掉, 适当增大输入图片的尺寸可以提升小目标的检测效果.

为了提高 SSD 对小目标的检测能力, Fu 等提出了 DSSD (De-convolutional Single Shot Detector) 方法[31]. DSSD 的核心思想: 提高浅层的表征能力. 首先将主干网络由 VGG 替换成更深的 ResNet-101, 增强了网络提取的能力, 其次修改了预测模块, 添加了类似于 ResNet 的 shortcuts 通道, 将各层次之间特征相结合. DSSD 的主要特点是增加了反卷积模块 DM (Deconvolution Module), DM 模块与 SSD 中的卷积层网络构成了不对称的“沙漏”结构. DM 模块与整个沙漏结构, 更充分利用了上下文信息和浅层的特征, 从而与 SSD 相比在小目标和密集目标的检测率上有很大的提高. 但是由于 ResNet-101 网络层数太深, 网络结构过于复杂, 导致检测速度大幅度降低, DSSD 检测 513×513 图像时速度只有 6 fps.

5.3 RetinaNet

Lin 等[24] 认为基于回归的目标检测方法精度不及基于候选区域的目标检测方法的根本原因在于“类别不平衡”, 基于候选区域的目标检测算法由于 RPN 网络的存在, 过滤掉了大部分背景框, 从而缓解了“类别不平衡”的问题. 而 one-stage 算法直接在生成的“类别极不平衡”的边框中进行难度极大的细分类, 意图直接输出边框和标签. 而原有交叉熵损失作为分类任务的损失函数, 无法抗衡“类别极不平衡”, 容易导致分类器训练失败. 因此, one-stage 目标检测算法虽然保住了检测速度, 却丧失了检测精度.

文献 [24] 提出的 RetinaNet 采用 Focal Loss 损失函数代替交叉熵误差, 来抑制量大的类别所贡献的损失值. 通过此种方法, 使得训练过程中量少的类别的权重更大, 从而缓解了“类别不平衡”的问题. RetinaNet 的网络结构如图 12 所示, 采用 ResNet+FPN 网络提取图像的多尺度信息, 再利用 one-stage 目标识别法+Focal Loss, 这个结构在 COCO 数据集上的 mAP 达到了 39.1%, 速度为 5 fps, 精度超过同期所有 two-stage 的检测器.

5.4 CornerNet

目前大部分常用目标检测算法 (比如 RCNN 系列, SSD, YOLO 系列等) 都是基于锚框 (即 anchor boxes) 进行目标检测, 引入锚框的缺点在于: (1) 正负样本不均衡. 大部分检测算法的锚框数量成千上万, 而目标数量相对少很多, 导致正样本数量会远远小于负样本. (2) 引入更多的超参数, 比如 anchor 的数量、大小和宽高比等. Law 等[25] 舍弃了传统的锚框思路, 提出了一种无锚框的目标检测新算法, 即 CornerNet 算法, 该算法使用单个卷积神经网络将目标边界框检测为一对关键点 (即边界框的左上角和右下角), 也就是使用单一卷积模型生成热点图和嵌入式向量.

CornerNet 算法架构包含 3 部分: 环面网络、右下角和左上角的热图、预测模块, 如图 13 所示.

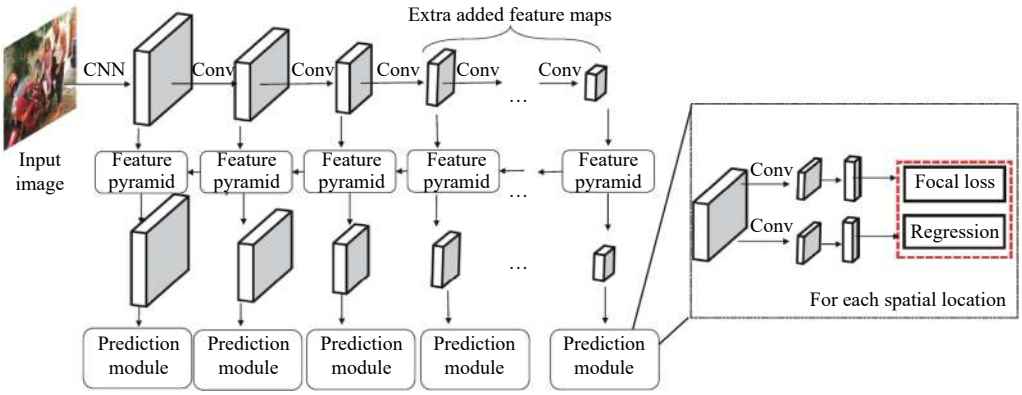

图 12 RetinaNet 网络结构图

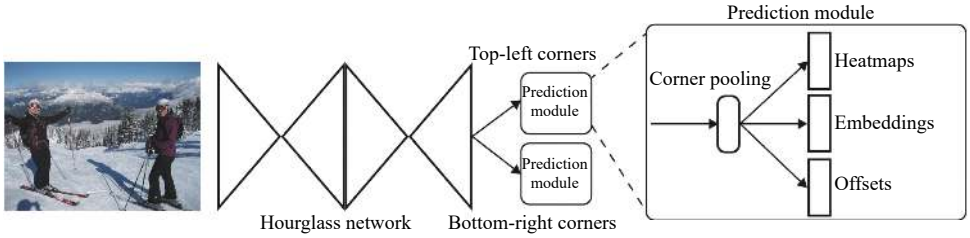

图 13 CornerNet 网络结构图

(1) 环面网络 (即 Hourglass 网络) 是人体姿态估计的典型架构, 堆叠两个环面网络生成两组热力特征图来预测不同分类下的角点, 其中一组负责预测左上角点, 另一组负责预测右下角点, 每一个角都包括角合并、对应的热图、嵌入式向量和偏移;

(2) 环面网络同时包含多个从下往上 (从高分辨率到低分辨率) 和从上往下 (从低分辨率到高分辨率) 过程, 目的是在各个尺度下抓取信息;

(3) 嵌入式向量使相同目标的两个顶点 (左上角和右下角) 距离最短, 偏移用于调整生成更加紧密的边界定位框.

CornerNet 算法消除了现有 one stage 检测算法中对锚框的需要, 整个检测网络的训练从头开始并不基于预训练的分类模型, 用户能够自由设计特征提取网络, 不用受预训练模型的限制. 同时该算法提出了一种新的池化方法 (即 corner pooling), 能够帮助网络更好地定位边界框的角点, 提高算法的检测精度.

5.5 CenterNet

CornerNet 算法在生成边界框的时候在生成边界框的时候受限于检测的角点对, 即如果生成角点对的质量够高, 则对应的边界框的质量也高; 反之, 则会产生错误的边界框.

在生成边界框是不能仅依靠角点对的信息, 还需要角点对生成的区域内部信息. Duan 等[26] 在 CornerNet 算法基础上改进并提出的 CenterNet 算法, 使模型能够自行判断生成边界框的正确性. CenterNet算法将左上角、右下角和中心点结合成为三元组进行物体框的判断, 不仅预测角点, 也预测中心点, 如果角点对所定义的预测框的中心区域包含中心点, 则保留此预测框, 否则弃掉. 同时如果预测的边界框与标注框有很大的交并比, 即该预测框的质量较高, 该预测框的中心区域应该包含中心关键点. 文献 [26] 为了提高检测角点对和中心点的质量, 提出具有创新性的级联角点池化 (cascade corner pooling) 和中心池化 (center pooling) 改善上述关键点的生成.

(1) 级联角点池化: 针对角点对的预测, 级联角点池化首先提取物体边界最大值 (corner pooling), 然后在边界最大值处继续向内部提取最大值, 并与边界最大.

(2) 值相加, 以结合更多内部信息, 使其具有感知内部信息的能力.

(3) 中心池化: 针对中心点的预测, 中心池化提取

中心点水平方向和垂直方向的最大值并相加, 以此给中心点提供所处位置以外的信息, 将有助于中心关键点获取目标的更多信息、感知边界框的中心区域.

CornerNet 算法引入了目标内的信息, 同时通过级联角点池化和中心池化两种策略来改善了各关键点的生成, 利用生成边界框内部的信息来筛选出高质量的边界框, 从而显著提升检测效果.

6 目标检测算法比较

目前基于深度学习的目标检测算法受到了学者的广泛关注和深入研究, 主要分为两大类算法: 基于候选区域的 two-stage 算法和基于回归的 one-stage 算法. 通过国内外学者的广泛研究两类算法衍生出了不同神经网络机制和特性, 不同算法之间的优缺点和适用场景均不同, 表 2 显示了几种典型算法优缺点.

表 2 不同目标检测算法优缺点总结

类型	算法	优点	缺点	应用场景
Two-stage	Faster R-CNN	(1) RPN提取候选区域 (2) 多任务损失函数边框回归 (3) 实现端到端的目标检测框架 (4) 检测精度较高	(1) 无法达到实时检测 (2) 获取候选框计算量较大	检测实时性要求不高的场景: 高空电力线路巡检; 农作物检测; 河流河道检测; 医学影像检测; ……
	Mask R-CNN	(1) 引入实例分割分支 (2) 掩模预测和分类预测拆解 (3) 像素级别目标检测 (4) 检测精度高	(1) 无法达到实时检测 (2) 实例分割分支增大计算量	
	Cascade R-CNN	(1) 使用级联检测器优选IoU阈值 (2) 解决高阈值引起的过拟合问题 (3) 检测精度高	无法达到实时检测	
	TridentNet	(1) 使用不同感受野多分支结构提取不同尺度目标特征 (2) 多分支结构权重共享 (3) 检测精度高	无法达到实时检测	
One-stage	YOLOv3	(1) 多尺度特征图提取 (2) 精度更高的分类网络(DarkNet-53) (3) 使用Logistic分类方法 (4) 检测速度快、精度较高	小目标/多目标检测精度差	检测实时性要求较高的应用场景: 表面缺陷在线检测; 火灾监控在线检测; 高空作业在线检测; 可疑人员排查在线检测; 自动驾驶目标检测; ……
	SSD	(1) 多尺度特征图提取 (2) 卷积特征检测 (3) 设置预选框 (4) 检测速度快、精度较高	(1) 小目标/多目标检测精度差 (2) 人工设置预选框参数, 经验依赖程度较高	
	RetinaNet	(1) 采用Focal loss损失函数 (2) 解决“类别不平衡”问题 (3) 检测速度快、精度较高	小目标/多目标检测精度差	
	CornerNet	(1) 关键点检测算法, 无锚框 (2) 解决锚框检测的样本不均衡和超参数问题 (3) 检测速度快、精度较高	(1) 小目标/多目标检测精度差 (2) 没有考虑边界框的内部信息	
	CenterNet	(1) 关键点检测算法, 无锚框 (2) 使用级联角点池化和中心池化 (3) 提高关键预测精度 (4) 检测速度快、精度高	小目标/多目标检测精度较差	

Two-stage 目标检测算法由于事先获取候选区域, 能够充分学习到目标的特征, 其检测精度和定位精度高, 但是网络结构复杂, 计算量大, 检测速度较慢, 不适用于实时性要求较高的应用场景. One-stage 目标检测算法结构简单, 可直接对输入图像进行处理, 检测精度较高并且检测速度快, 可以实现实时性检测, 能满足一些实时在线检测应用场景, 如表面缺陷实时检测, 火灾实时检测, 高空作业实时检测等, 但是 one-stage 算法对小目标、多目标物体检测精度较低, 特别是在复杂场景下, 检测精度并不能满足要求, 如自动驾驶领域的

目标检测. 目前 TridentNet 和 CenterNet 分别为 two-stage 和 one-stage 目标检测系列算法中检测精度相对较高的算法, 但它们对小目标检测的精度仍比较低, 对于目标尺度跨度非常大的应用场景仍不满足要求, 因此, 目前的绝大多数目标检测算法均只能应用于场景相对较简单却目标尺度跨度不大的应用场景.

7 总结和展望

本文对基于深度学习的主流目标检测算法做了简要综述, 主要包括典型算法的思路、创新策略、检测精度和应用场景等方面. 虽然在过去的 20 年中物体目标检测取得了显著的成就, 仍然存在许多难以解决的问题[32], 下面是对面临的难题以及未来发展方向的一些讨论:

(1) 小目标检测. 目前算法对于小目标的检测能力相对较弱, 但无人机航拍、卫星遥测、红外目标识别等领域对小目标检测有较强的需求. 小目标图像往往面临着分辨率低、像素少和训练数据难以标记的问题. 例如在无人机高清航拍过程中, 即使照片分辨率已经达到 4k 级别, 但由于小目标所占面积较小, 导致难以标定和训练.

(2) 弱监督目标检测方法. 目前绝大多数目标检测算法的精度均依赖大量标注完整的图像数据集, 在大型数据集中标注工作量大, 标注时间长, 对于算法模型训练时间也长, 因此目前目标检测算法的实施成本较高, 难度相对较大. 弱监督目标检测方法是利用少量完整标注的图像自动检测大量未完整标注的图像, 这将大大降低目标检测模型的开发难度和周期, 因此开发弱监督目标检测方法是一个值得进一步研究的重要问题.

(3) 多领域目标检测: 目前算法基本只针对特定场景特定目标物进行检测, 特定领域的检测器只能够在指定数据集上实现较高的检测性能, 特定领域的检测器应用场景单一, 不具备多领域多场景通用性. 为了得到一种适用于各种图像检测领域的通用检测器, 多领域目标检测器可以在不存在新领域先验知识的情况下解决这一问题, 但检测领域转移是一个具有挑战性的课题, 有待进一步研究.

(4) 多任务学习: 多层次特征的聚合骨干网络是提高检测性能的重要方法. 当同时进行多个计算机视觉任务, 如目标检测、语义分割、实例分割、边缘检测、突出点检测等, 可以获得更丰富的信息, 大大提高单独任务的性能. 多任务学习是一种将多个任务聚合到一个网络中的学习方法, 其在保持处理速度和提高准确率同时, 对研究者提出了很大的挑战.

参考文献

1 Zou ZX, Shi ZW, Guo YH, *et al*. Object detection in 20 years: A survey. arXiv: 1905.05055, 2019.

2 Oksuz K, Cam BC, Kalkan S, *et al*. Imbalance problems in object detection: A review. arXiv: 1909.00169, 2019.

3 Jiao LC, Zhang F, Liu F, *et al*. A survey of deep learning-based object detection. IEEE Access, 2019, 7: 128837–128868. [doi: 10.1109/ACCESS.2019.2939201]

4 Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. Nature, 1986, 323(6088): 533–536. [doi: 10.1038/323533a0]

5 Hinton G, Deng L, Yu D, *et al*. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine, 2012, 29(6): 82–97. [doi: 10.1109/MSP.2012.2205597]

6 Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. Science, 2006, 313(5786): 504–507. [doi: 10.1126/science.1127647]

7 Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Communications of the ACM, 2017, 60(6): 84–90. [doi: 10.1145/3065386]

8 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. Proceedings of the 3rd International Conference on Learning Representations. San Diego, CA, USA. 2015.

9 Szegedy C, Liu W, Jia YQ, *et al*. Going deeper with convolutions. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, MA, USA. 2015. 1–9.

10 He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA. 2016. 770–778.

11 LeCun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015, 521(7553): 436–444. [doi: 10.1038/nature14539]

12 Oquab M, Bottou L, Laptev I, *et al*. Learning and transferring mid-level image representations using convolutional neural networks. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA. 2014. 1717–1724.

13 Kavukcuoglu K, Ranzato MA, Fergus R, *et al*. Learning

invariant features through topographic filter maps. Proceedings of 2009 IEEE Conference on Computer Vision and Pattern Recognition. Miami, FL, USA. 2009. 1605–1612.

14 Girshick RB, Donahue J, Darrell T, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA. 2014. 580–587.

15 Kavukcuoglu K, Sermanet P, Boureau YL, *et al*. Learning convolutional feature hierarchies for visual recognition. Proceedings of the 23rd International Conference on Neural Information Processing Systems. Vancouver, Canada. 2010. 1090–1098.

16 Girshick R. Fast R-CNN. Proceedings of 2015 IEEE International Conference on Computer Vision. Santiago, Chile. 2015. 1440–1448.

17 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149. [doi: 10.1109/TPAMI.2016.2577031]

18 He KM, Gkioxari G, Dollár P, *et al*. Mask R-CNN. Proceedings of 2017 IEEE International Conference on Computer Vision. Venice, Italy. 2017. 2980–2988.

19 Cai ZW, Vasconcelos N. Cascade R-CNN: Delving into high quality object detection. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA. 2018. 6154–6162.

20 Li YH, Chen YT, Wang NY, *et al*. Scale-aware trident networks for object detection. Proceedings of 2019 IEEE/CVF International Conference on Computer Vision. Seoul, Republic of Korea. 2019. 6053–6062.

21 Redmon J, Farhadi A. YOLO9000: Better, faster, stronger. Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA. 2017. 6517–6525.

22 Liu W, Anguelov D, Erhan D, *et al*. SSD: Single shot multibox detector. Proceedings of the 14th European Conference European Conference on Computer Vision. Amsterdam, the Netherland. 2016. 21–37.

23 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv: 1804.02767, 2018.

24 Lin TY, Goyal P, Girshick R, *et al*. Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 318–327. [doi: 10.1109/TPAMI.2018.2858826]

25 Law H, Deng J. CornerNet: Detecting objects as paired keypoints. International Journal of Computer Vision, 2020, 128(3): 642–656. [doi: 10.1007/s11263-019-01204-1]

26 Duan KW, Bai S, Xie LX, *et al*. CenterNet: Keypoint triplets for object detection. Proceedings of 2019 IEEE/CVF International Conference on Computer Vision. Seoul, Republic of Korea. 2019. 6568–6577.

27 LeCun Y, Bottou L, Bengio Y, *et al*. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11): 2278–2324. [doi: 10.1109/5.726791]

28 Huang G, Liu Z, Van Der Maaten L, *et al*. Densely connected convolutional networks. Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA. 2017. 2261–2269.

29 He KM, Zhang XY, Ren SQ, *et al*. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904–1916. [doi: 10.1109/TPAMI.2015.2389824]

30 Redmon J, Divvala S, Girshick R, *et al*. You only look once: Unified, real-time object detection. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA. 2016. 779–788.

31 Fu CY, Liu W, Ranga A, *et al*. DSSD: Deconvolutional single shot detector. arXiv: 1701.06659, 2017.

32 Wu XW, Sahoo D, Hoi SCH. Recent advances in deep learning for object detection. Neurocomputing, 2020, 396: 39–64. [doi: 10.1016/j.neucom.2020.01.085]