深度学习目标检测方法及其主流框架综述

段仲静[1]，李少波[1,2*]，胡建军[2]，杨静[2]，王铮[2]

[1]贵州大学现代制造技术教育部重点实验室，贵州 贵阳 550025；

[2]贵州大学机械工程学院，贵州 贵阳 550025

摘要 目标检测作为机器视觉中重要任务之一，是人工智能体系中一个具有重要研究价值的技术分支。对于卷积神经网络框架、anchor-based 模型和 anchor-free 模型三个主流的目标检测模型进行梳理。首先，综述了主流卷积神经网络框架的网络结构、优缺点以及相关的改进方法；其次从 one-stage 和 two-stage 两个分支对 anchor-based 类模型进行深入分析，总结了不同目标检测方法的研究进展；从早期探索、关键点和密集预测三部分分析 anchor-free 类模型。最后对该领域的未来发展趋势进行了思考与展望。

关键词 图像处理；深度学习；目标检测；网络框架；anchor-based 模型；anchor-free 模型

中图分类号 TP391 **文献标志码** A **doi**：10.3788/LOP57.120005

Review of Deep Learning Based Object Detection Methods and Their Mainstream Frameworks

Duan Zhongjing[1], Li Shaobo[1,2*], Hu Jianjun[2], Yang Jing[2], Wang Zheng[2]

[1] *Key Laboratory of Advanced Manufacturing Technology of Ministry of Education, Guizhou University, Guiyang, Guizhou* 550025, *China*;

[2] *School of Mechanical Engineering, Guizhou University, Guiyang, Guizhou* 550025, *China*

Abstract As one of the important tasks in machine vision, object detection is a technology branch with important research value in artificial intelligence systems. The three mainstream object detection models of convolutional neural network framework, anchor-based model, and anchor-free model are analyzed. First, the network structure and the advantages and disadvantages of the mainstream convolutional neural network framework, and the related improvement methods are reviewed. Second, the anchor-based model is deeply analyzed from one-stage and two-stage branches, and the research progresses of different object detection methods are summarized. The anchor-free model is analyzed from three parts: early exploration, key points, and intensive prediction. Finally, the future development trend of the field is considered and prospected.

Key words image processing; deep learning; object detection; network framework; anchor-based model; anchor-free model

OCIS codes 100.4996; 150.1135; 150.4065

1 引 言

基于深度学习的目标检测作为机器视觉中重要的任务之一，在自动驾驶、图像分类、人脸检测、视觉搜索、目标跟踪与检测、医疗诊断等领域得到广泛的应用。文献[1]讨论了基于视觉的目标检测与跟踪的研究进展。文献[2]对基于深度学习的自动驾驶技术进行了归纳分析。针对道路车辆多目标检测和车型分类难等问题，文献[3-5]提出了基于深度学习的车辆检测算法。文献[6-7]针对行为理解和智能

收稿日期：2019-11-11；**修回日期**：2019-11-28；**录用日期**：2019-12-06

基金项目：国家自然科学基金(51475097,91746116)、工信部资助项目(工信部联装[2016]213 号)、贵州省科技计划(黔科合平台人才[2015]4011、黔科合平台人才[2016]5103)、黔教合协同创新字[2015]002、贵州省研究生创新基金(黔教合YJSCXJH[2018]052)

* **E-mail**：lishaobo@gzu.edu.cn

监控分别综述了基于深度学习的目标学习算法的研究进展。文献[8]对基于机器视觉的缺陷检测技术进行了归纳总结。传统的目标检测算法多是基于滑动窗口模型，对特征进行手工提取和匹配有着单一性、计算复杂、适用性不佳的缺点，检测效率和准确度较差。随着机器视觉的发展，基于深度学习的目标检测技术以网络结构简单高效的特点，超越了传统算法，准确度和效率大幅提升，逐渐成为当前的主流算法。

目标检测算法模型主要分为 anchor-based 类模型和 anchor-free 类模型，其中 anchor-based 类模型又分为基于回归的 one-stage 和基于候选框生成与分类的 two-stage 方法，one-stage 方法不需要预生成候选框，只需要完成特征抽取、分类和定位回归三个任务；two-stage 方法主要有 4 个任务：特征抽取、生成候选框、分类和定位回归。在检测性能上，one-stage 方法检测速度更快，适合在移动终端等平台上使用，而 two-stage 方法需要生成目标候选区域，可获得较丰富的特征和较高的准确率，但检测速度比较慢。

基于以上分析，对基于深度学习的目标检测算法及主流神经网络框架进行总结和比较。梳理了主流卷积神经网络框架，分析了网络结构和算法性能及优缺点；从 one-stage 和 two-stage 两个方面对目标检测算法的 anchor-based 类模型进行深入分析；从 anchor-free 类模型早期探索、基于关键点和密集预测进行深入剖析；针对基于深度学习的目标检测算法的未来研究趋势和应用方向进行了思考和展望。

2 主流的卷积神经网络框架

2.1 LeNet

1998 年，LeNet 由在深度学习领域有着三巨头之一美誉的 LeCun 等[9]提出，这是最早、最著名的神经网络结构之一。LeNet 主要用于识别手写字符，最高识别准确率为 98%。LeNet 奠定了现代卷积神经网络的基础。

图 1 是 LeNet 结构图，它是一个 6 层的网络结构，包括 3 个卷积层(C)、2 个下采样层和 1 个全连接层。其中 C5 层也可视为完全连接层，因为它的卷积核大小与输入图像的大小相同，即 5×5。每个卷积层包括卷积、池化以及 sigmoid 激活函数三部分，使用卷积提取空间特征，降采样层采用平均池化，最后采用 softmax 作为分类器。

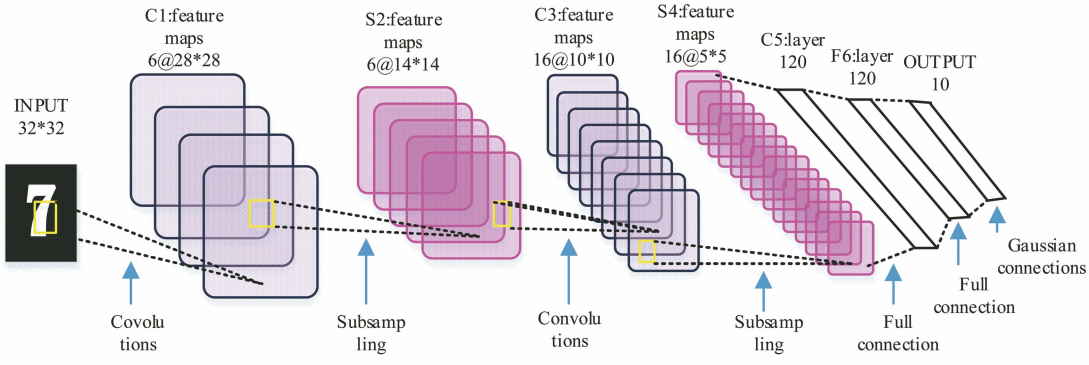

图 1 LeNet 网络结构图[9]

Fig. 1 LeNet network structure[9]

2.2 AlexNet

为了增加 LeNet 网络的深度和广度，2012 年，Krizhevsky 等[10]设计了 AlexNet 网络，以极大的优势赢得了当年的 ImageNet[11]大规模视觉识别挑战赛(ILSVRC)，证明了复杂模型下卷积神经网络的有效性，建立了神经网络在计算机视觉领域中的主导地位。

AlexNet 的结构和参数设置如图 2 所示，它是一个 8 层的网络结构[不包括激活、池化、LRN (local response normalization)和 dropout[12]层]，其中包括 5 个卷积层和 3 个全连接层。第一个卷积层为 11×11 的卷积核，步长设置为 4；第二个卷积层的卷积核为 5×5，步长为 1；其余卷积层的大小均为 3×3，步长为 1；激活函数使用 ReLU(rectified linear units)[13]，池化层使用大小为 3×3 的重叠池，步长为 2；将 dropout 层添加到完全连接层的好处是：1)对训练模型进行了并行化加速，极大缩短了训练周期；2)ReLU 作为激活函数对深度网络梯度分散问题具有较大的帮助；3)使用数据增强[14]、dropout 和 LRN 层来阻止网络过度拟合，提高模型的泛化能力。

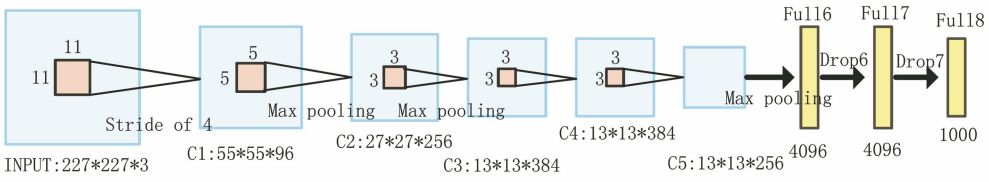

图 2 AlexNet 网络结构图[10]

Fig. 2 AlexNet network structure[10]

2.3 VGGNet

VGGNet[15]是由 Google DeepMind 公司和牛津大学计算机视觉组联合提出的深度卷积网络，它证明网络深度是影响性能的关键因素。该网络有着良好的泛化性能，易于移植到别的图像识别项目，并可下载 VGGNet 已经训练好的参数以实现良好的初始化权重操作。许多卷积神经网络都是基于 VGGNet，如 FCN[16]（fully convolutional networks）、UNet[17]、SegNet[18]等。VGGNet 版本很多，常用的是 VGG16、VGG19 网络。

VGG16 的网络结构如图 3 所示，共有 16 层（不计池化层和 softmax），池化层为 2×2、步长也是 2 的最大池化，卷积核都是 3×3，卷积层深度分别为 64，128，256，512，512。VGG16 网络结构与 AlexNet 相似，区别在于：1）VGGNet 拥有 16～19 层的网络层数，而 AlexNet 只有 8 层；2）VGG16 把卷积层上升到卷积块的概念，卷积块包括 2～3 个卷积层，增大了网络感受野，减少了网络参数，并且通过反复使用激活函数 ReLU，可得到更多的线性变换，进一步提高了学习能力。多尺度用于训练和预测期间的数据增强，将相同的图像缩放到不同的尺寸以进行预测，最后取平均值。

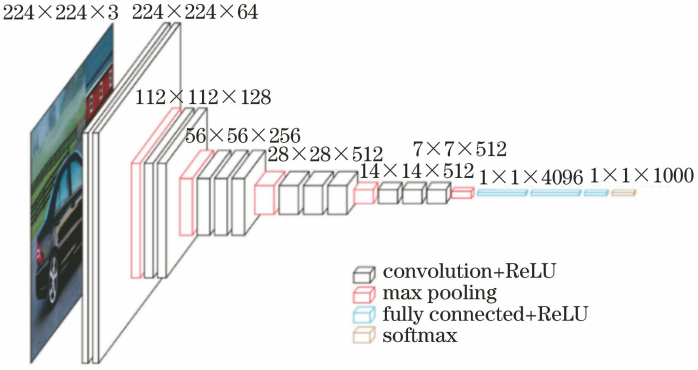

图 3 VGG16 网络结构图[15]

Fig. 3 VGG16 network structure[15]

2.4 GoogLeNet

GoogLeNet 是 Szegedy 等[19]提出的一种全新深度学习结构，网络结构和 VGGNet 类似，在主干卷积环节使用 5 个模块，模块之间使用步幅为 2 的 3×3 最大池化层以减小输出的高和宽。区别于 VGG 在不同层通过增加 loss 损失函数和 inception 结构两种方式，GoogLeNet 的好处是很大程度上加深和加宽了网络，减少了参数量，将错误率降至 6.656%，但该模型的计算复杂度高，修改通道数困难。

如图 4 所示，Inception 块有 4 条并行路径，无全连接层，可减少运算时间，在后 3 条线路中添加 1×1的卷积层以缩小输入通道数，降低模型复杂度，使 GoogleNet 的参数量是 AlexNet 的 1/12，远少于

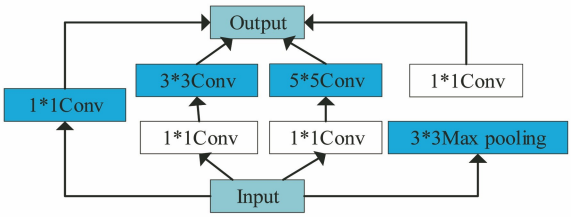

图 4 GoogLeNet 网络结构图[19]

Fig. 4 GoogLeNet network structure[19]

AlexNet 和 VGGNet。为了方便对齐，前 3 条路径使用窗口大小分别为 1×1、3×3 和 5×5 的卷积层来获取多尺度空间信息，4 条路径都使用相应的填充来使输入与输出的高和宽保持一致，最后将每条路径的输出在通道维上连接，并输入到接下来的层级中。

2.5 ResNet

2015 年 He 等[20]提出 ResNet 网络，使用 152 层超深卷积神经网络对输入数据进行训练，取得较好的效果。ResNet 网络在 5 个重要任务轨迹中都是最好的：对于 ImageNet 分类任务，错误率为3.57%；对于 ImageNet 检测任务，准确率超过第二名 16%；对于 ImageNet 定位任务，准确率超过第二名 27%；对于 COCO 检测任务，准确率超过第二名 11%；对于 COCO 分割任务，准确率超过第二名 12%。这让 ResNet 成为目前最好的卷积神经网络模型之一。

图 5 为 ResNet 的基本模块，x 为输入样本，$F(x)+x$是输出结果，$F(x)$表示网络中数据的运算方式。如果 $H(x)=F(x)+x$ 是神经网络的最优拟合结果，则最优的 $F(x)$就是 $H(x)$和 x 的残差，采用拟合残差的方法来改善网络的性能，在训练期间保证了残差为零，因此含残差学习单元的深度学习模型网络性能不会受到影响。ResNet 模型就是残差学习单元的连续叠加，理论上无限叠加也不会改变网络性能。

ResNet 的创新点是：1)实现了深层的神经网络结构，解决了因不断深化神经网络而使得准确率达到饱和的问题；2)输入和输出能直接相连，这样学习残差就是整个网络的工作，很好地简化了学习目标与难度；3)ResNet 是一种迁移性很好的网络结构，易于与其他网络集成。

2.6 DenseNet

Huang 等[21]提出了DenseNet网络，主要构建

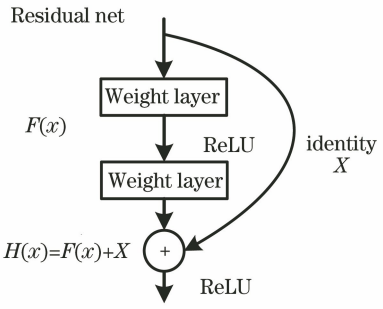

图 5 ResNet 网络的基本模块[20]

Fig. 5 Basic module of ResNet network[20]

模块是稠密块和过渡层。稠密块为稠密连接的 highway 的模块，过渡层为相邻 2 个稠密块的中间部分。稠密块定义输入和输出的连接方法，过渡层用于确定通道数。稠密块内部特征图大小必须一致，层级输入是多个字符串的连接，区别于 ResNet 的 element-wise 连接，内部每个节点代表 BN＋ReLU＋Conv。

传统卷积神经网络中，如果有 L 层，就有 L 个连接，但在 DenseNet 中，每个稠密块都利用该模块中前面所有层的信息，如图 6 所示，即每层都与前面层有 highway 的稠密连接，连接数目为 $L\times(L+1)/2$。highway 的稠密连接方式缓解了深层网络的梯度消失问题，特征得到了重用，大幅度减少了模型参数，甚至减少了在小样本数据上的过拟合。其缺点是：随着稠密块深度的加深，深层输入特征图谱的维度和最终输出的维度都非常大，针对该问题，采取在稠密块里添加Bottleneck单元和在过渡层里添加

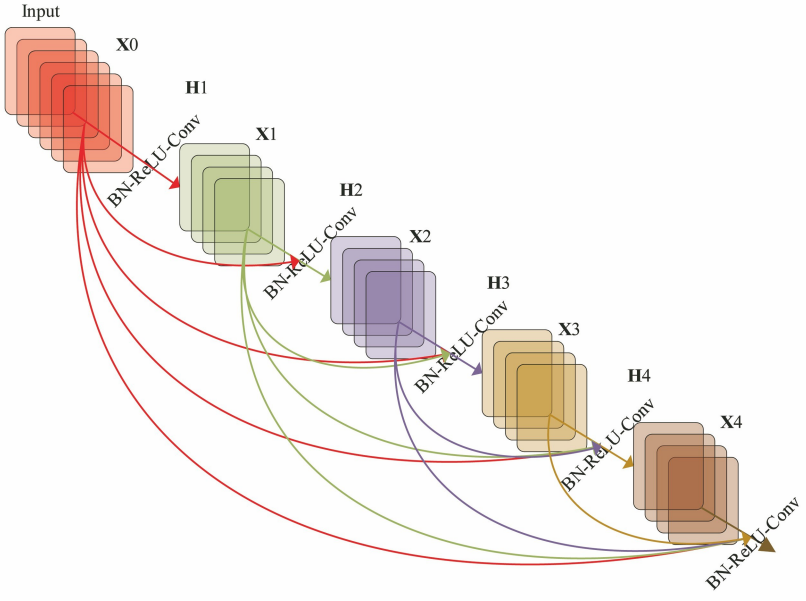

图 6 DenseNet 网络结构图[21]

Fig. 6 DenseNet network structure[21]

1×1 卷积的方式来降维。

3 Anchor-based 类模型

目前主流的 anchor-based 类目标检测方法分为 two-stage 的目标检测算法和 one-stage 的目标检测算法。Two-stage 方法将通过算法得到的一系列候选框作为样本，由卷积神经网络完成样本分类。One-stage 方法没有生成候选框环节，直接把目标边框的定位问题转换为回归问题。在检测准确率和定位精确度上 two-stage 方法较优，而 one-stage 方法的检测速度更快。图 7 给出了近几年涌现出来的优秀目标检测算法。

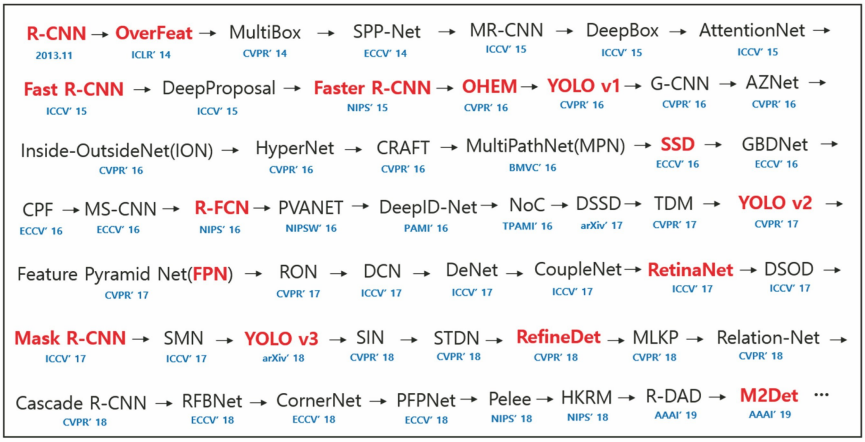

图 7 2013 年 11 月至 2019 年 10 月目标检测算法总览

Fig. 7 Overview of target detection algorithms proposed from November 2013 to October 2019

3.1 Two-stage 方法

3.1.1 R-CNN

2013 年 Girshick 等[22]提出 R-CNN(region-based convolutional neural networks)，将 PASCAL VOC2007 测试集的 mAP 提高到了 58.5%。经过改进的 R-CNN 将在该数据集上的 mAP 提高到 66%，在 ILSVRC 2013 数据集上的 mAP 提高至 31.4%。图 8 为 R-CNN 特征学习过程示例，R-CNN 算法使用选择性搜索[23]算法来评估相邻图像子块的特征相似性，给合并后的相似图像区域打分，选择感兴趣区域(ROI)的候选框作为卷积神经网络样本输入，由标定框与候选框组成的正负样本特征形成相应的特征向量，采用 SVM(support vector machine)对特征向量进行分类，最后返回标定框与候选框，以达到目标检测的目的。

R-CNN 主要缺点：1)重复计算量大，约有 2000 个候选框的方案中，每个候选框都需要经过 backbone 网络单独提取特征，候选框会重叠，产生大量重复计算；2)训练测试复杂，候选区域获取、特征获取、分类和回归都是单独运行的，中间数据也是单独保存的；3)速度缓慢，前两个缺点是R-CNN速度慢的主要原因，难以满足实时性需求；4)输入图像大小的限制，输入图像大小被强制缩小为 277 pixel×277 pixel，这将导致检测目标对象形变，使检测性能下降；5)需要进行 SVM 与特征回归的后期操作，并在 SVM 与特征回归期间不学习更新 CNN 特征。

R-CNN 创新点：1)将大规模的卷积神经网络(CNNs)应用于自下而上的候选区域以定位和分割对象；2)当标记的训练集不足时，对辅助任务执行监督训练，然后执行特定任务的优化，提高模型性能。

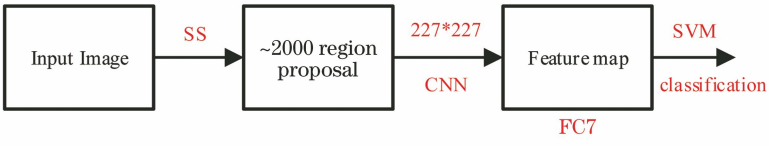

图 8 R-CNN 实现流程图[22]

Fig. 8 R-CNN implementation flow chart[22]

3.1.2 SPP-Net

针对卷积神经网络重复运算和形状扭曲变形等问题，He 等[24]基于 R-CNN 于 2015 年提出 SPP-Net (Spatial Pyramid Pooling Networks)算法。

SPP-Net 舍弃了 R-CNN 在输入神经网络之前剪裁候选框和图像子块缩放操作，在卷积层与全连接层中间添加 SPP（spatial pyramid pooling）结构，提升了候选框的生成速率，节省了计算开销。SPP-Net 实现示意图如图 9 所示，该方法需要生成候选框，与传统方法不同的是从特征图上获取候选框特征向量的过程被设置到卷积操作后，将 R-CNN 中的若干次卷积转换为一个卷积，减少了模型的计算量。

SPP-Net 缺点：1）与 R-CNN 设计相同，训练经历了多个阶段，中间特征数据也必须保存，增加了时间开销；2）分类网络的初始参数被承接到 backbone 网络中，并未针对检测问题进行优化；3）训练样本的大小不一致，这将增大候选框的 ROI 感受野，权重不能被神经网络快速更新；4）SPP中的微调只更新 SPP 层后面的全连接层，当网络很深时这样做难以奏效。

SPP-Net 创新点：1）利用空间金字塔池化结构；2）对整张图片只进行一次特征提取，运算速度较快。

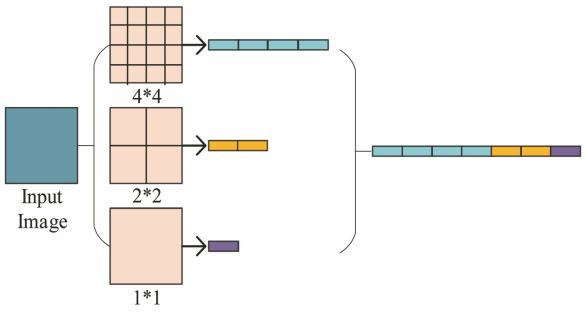

图 9 SPP-Net 实现示意图[24]

Fig. 9 Schematic diagram of SPP-Net implementation[24]

3.1.3 Fast R-CNN

针对 SPP-Net 存在的问题，Girshick 等[25] 于 2015 年提出 Fast R-CNN（fast region-based convolutional neural networks），改进了 ROI pooling 层，将不同大小候选框的特征图采样成大小固定的特征。ROI 池化层的功能和 SPP 层类似，但 ROI 更简单，仅采用单个尺度来划分网格和池化，该层可以直接执行求导操作，并直接将梯度传输到 backbone 网络。该算法针对在训练期间多阶段和特征重复计算造成的时间代价及中间特征数据需要存储造成的空间成本问题进行了研究，把深度网络与 SVM 分类相结合，构成 multi-task 模型，分类和回归由全连接层网络同时执行[26]。把 R-CNN 在 PASCAL VOC2007 数据集中训练的时间从 84 h 缩短到 9.5 h，检测时间从 45 s 缩短到 0.32 s。图 10 是 Fast R-CNN 实现流程图。

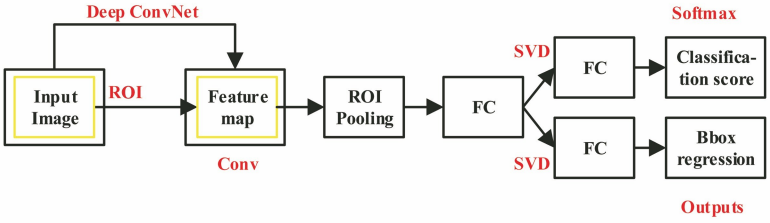

图 10 Fast R-CNN 实现流程图[25]

Fig. 10 Fast R-CNN implementation flow chart[25]

3.1.4 Faster R-CNN

SPP-Net 与 Fast R-CNN 都需要单独的候选区域模块，该模块运算量大。为了解决该问题，Ren 等[27] 在 Fast R-CNN 的基础上提出 Faster R-CNN（faster region-based convolutional neural networks）算法。如图 11 所示，在主干网络结构中添加 RPN[28] 是 Faster R-CNN 的主要创新，按照既定规则设置多尺度的锚点。用 RPN 卷积层中获取的候选框替换选择搜索传递的候选框，以及通过建议生成窗口的 CNN 与目标检测的 CNN 共享，实现网络端到端的训练。在训练期间，除通过模型各单元学习实现对应任务外，还配合自主学习。

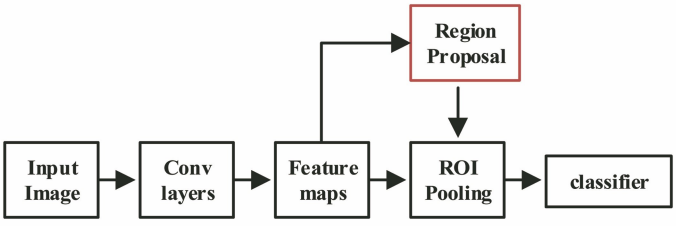

图 11 Faster R-CNN 实现流程图[27]

Fig. 11 Flow chart of Faster R-CNN implementation[27]

图 12 是对 R-CNN、Fast R-CNN、Faster R-CNN 三者实现过程的简单对比，它们的算法思想是一脉相承的，是迭代更新的关系，R-CNN 包括提取候选区域模块、提取特征向量网络、分类器 SVM 和边界框回归 4 部分。Fast R-CNN 利用 ROI pooling 层把 R-CNN 的 SVM 分类与边界框回归并入神经网络。Faster R-CNN 采用 RPN 层把 Fast R-CNN 的候选区域提取层融合到神经网络中，实现了端到端的训练。表 1 中给出三者在使用方法、缺点和改进方面的差异对比。

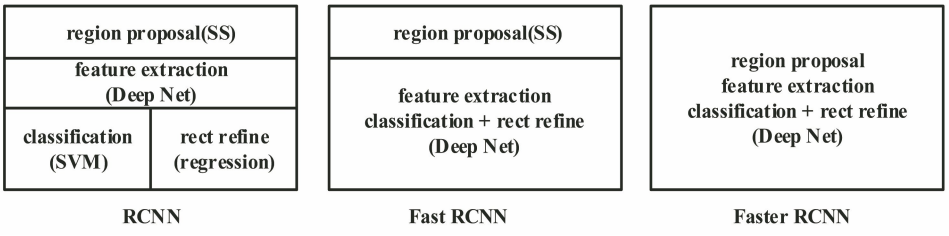

图 12 R-CNN，Fast R-CNN，Faster R-CNN 实现过程对比

Fig. 12 Implementation process comparison of R-CNN, Fast R-CNN, and Faster R-CNN

表 1 R-CNN，Fast R-CNN，Faster R-CNN 对比

Table 1 Comparison of R-CNN, Fast R-CNN, and Faster R-CNN

Model	Used method	Disadvantage	Improvement
R-CNN	1) Region proposal (SS); 2) extraction feature (ConvNet); 3) classification (SVM); 4) regression (Candidate Bbox)	1) Complex training steps; 2) training and testing are slow and take up a lot of disk space; 3) CNN features are not learned and updated during SVM and regression	1) Refresh mAP of DPM HSC from 34.3% to 66%; 2) region proposal and convolution network are used
Fast R-CNN	1) Region proposal(SS); 2) extraction feature (ConvNet); 3) classification(softmax); 4) Bbox regression (multi-task loss function)	1) RP is still extracted with SS (consuming time of 2–3 s); 2) difficult to meet real-time requirements; 3) GPU is utilized, but the region proposal method is implemented on CPU	1) mAP is increased by 4% from 66%; 2) speeds of training and testing are improved
Faster R-CNN	1) Region proposal network(RPN); 2) extraction feature (ConvNet); 3) classification(softmax); 4) Bbox regression (multi-task loss function)	1) Real-time object detection is not realized; 2) computation of obtaining region proposal and reclassification is very large	1) It only takes 10 ms to generate suggestion box by using convolution network; 2) accuracy and speed of detection are improved; 3) implement end-to-end target detection framework

3.1.5 R-FCN

针对 Faster R-CNN 仅学习 ROI 池化层以前的卷积网络特征参数，2016 年 Dai 等[29]基于 FCN 提出 R-FCN 方法，用基于位置敏感分布的卷积网络替换 ROI 池化层后的全连接网络[30-31]，降低了 ROI 池化层后网络对各个样本区域的计算时间成本。R-FCN 整个网络实现特征共享，缓解了目标分类对平移不变性的要求及目标检测对有平移变化要求之间的矛盾，主要不足是缺乏对候选区域全局信息与语义信息的利用。

R-FCN 沿用了 Faster R-CNN 的框架结构，区别在于引入位置敏感的分图取代 ROI-wise subnetwork，位置敏感的分图使用 ROI Pooling 来完成信息采样，融合分类与位置信息。R-FCN 在 PASCAL VOC2007 上的准确率为 79.5%，每个样本测试的平均时间为 170 ms，比 Faster R-CNN 快 2.5～20 倍。R-FCN 会在得到特征图谱时生成一个随着分类类别数线性增长的 channel 数，其好处是能提升目

标检测精度，但检测速度会减小，实时性变差。

3.1.6 FPN

FPN（feature pyramid network）算法由Lin等[32]在2016年提出。FPN改进了CNN网络对特征的提取方式，让特征能更好地表达出图片各个维度的信息。低层特征只有较少的语义信息，但目标位置准确；高层特征拥有丰富的语义信息，但目标相对粗糙。FPN很好地将低层特征的高分辨率和高层特征的语义信息相结合，同时使用不同层的特征来实现预测。

FPN的预测是在各个特征层上独立执行，让深层特征经过上采样与低层特征进行融合，再用3×3的卷积核卷积各个合并结果，以消除上采样的混叠效应。FPN主要分为三步：1）从下到上不同维度的特征生成；2）从上到下对特征进行补充增强；3）输出的不同维度特征和CNN网络提取的特征之间的关联表达。

3.1.7 Mask R-CNN

针对two-stage方法中以R-CNN为代表存在检测速度慢的问题，2020年He等[33]提出Mask R-CNN算法。Faster R-CNN在进行ROI池化和下采样时对特征图大小都进行取整运算，这使检测任务，特别是语义分割这种像素级任务的精度深受影响。如图13所示，Mask R-CNN舍弃对图片大小的取整操作，提出用ROI Align取代ROI Pooling层，用双线性插值法填充非整数位置的像素，使下游特征图谱向上游映射时不会产生位置误差。在边框识别基础上增加FCN层，用于语义Mask识别，添加的掩模预测结构较好地缓和了特征图谱和输入图像ROI的位置回归问题，避免了对ROI边界进行任何量化操作，使得掩码准确率提高10%～50%，此外Mask R-CNN基于基础网络ResNet[34]在COCO数据集上的检测准确率提高至39.8%。

Mask R-CNN在实例分割和检测精度方面都达到当时的最高水准，即使其后的一些算法对其性能有所提高，但基本都保持在同一层次。该算法的最大缺陷是检测速度难以满足实时需要，标注代价过于昂贵也是实例分割面临的一大问题。

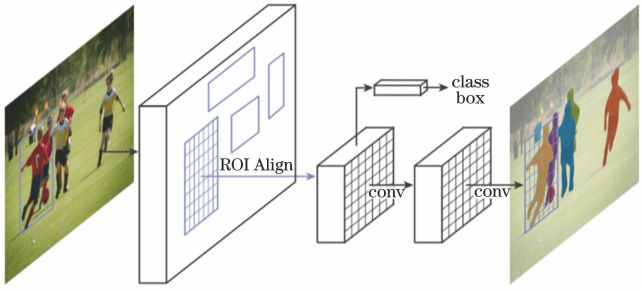

图13 Mask R-CNN实现流程图[33]

Fig. 13 Flow chart of Mask R-CNN implementation[33]

3.1.8 MegDet

基于CNN的物体检测研究一直在进步，从R-CNN到Fast/Faster R-CNN，再到Mask R-CNN，新的网络结构，新的范式或新的损失函数是主要的改进方式。然而目前的研究仍然缺乏对训练中的关键因素mini-batch的关注。

2017年Peng等[35]提出一种大mini-batch的目标检测算法MegDet，使用更大的mini-batch来训练网络，mini-batch大小从16增加到256，并且在训练时多块GPU能被高效地联合使用（在文献[35]中最多用了128块GPU），极大缩短了训练时间。同时，MegDet解决了Batch Normalization统计不准确的问题，建立了一种学习率的选择策略和跨GPU的批量标准化[36]方法，当二者被一起使用时，mini-batch物体检测器的训练时间减少，且可获得更高的检测准确度。

3.1.9 Two-stage目标检测相关改进工作

RCNN解决了为什么不用CNN进行分类的问题，但该方法要进行边界框回归操作和SVM分类。Fast-RCNN解决了bounding box和label不能共同输出的问题，但选择搜索候选区域的时间太长，Faster-RCNN解决了选择搜索的问题。Mask-RCNN是一个灵活的框架，可以完成目标分类和检测、语义和实例分割、人体姿势识别等多种任务。之后的two-stage目标检测算法基本都离不开RCNN。如MR-CNN[37]联合三种方法来精修样本边框，提高了定位效果。HyperNet[38]综合网络低中高多个层级的特征形成Hyper特征图。CRAFT[39]在生成候选框和候选框分类两个阶段增加不同的分类器，对候选区域进行筛选和更准确的

位置回归。A-Fast-RCNN[40]使用对抗学习机制，增强对遮挡与变形目标的检测效果。针对 two-stage 方法速度慢的问题，Light-Head R-CNN[41]设计一种新的两阶段检测结构。Cascade R-CNN[42]通过逐步提高判别正负样本的 IoU(Intersection over Union)阈值，以保证在样本数不减少的前提下训练出高质量的检测器。SNIP[43]利用图片金字塔在训练期间减少尺寸差异，解决了检测数据集上尺寸极端变化的问题。NAS-FPN[44]是对 RetinaNet 框架的改进，NAS 使用强化学习训练控制器，从给定的搜索空间里选择最优的模型架构。TridentNet[45]针对卷积核难以充分训练不同尺寸目标的问题，提出不同分支参数共享策略。

3.2 One-stage 方法

3.2.1 OverFeat

2013 年 Sermanet 等[46]提出 OverFeat 方法。OverFeat 的候选框是通过滑动窗口和规则块获得，然后通过多尺度滑动窗口来改善检测结果，缓解了图像目标形状复杂、尺寸不同造成的特征学习困难问题。最后，利用卷积神经网络与回归模型对目标进行分类和定位。OverFeat 是第一个同时处理分类、定位及目标检测多个计算机视觉任务的算法，mAP 为 24.3%的成绩夺得 ILSVRC 2013 任务 3(分类＋定位)的桂冠，但很快就被同期的 R-CNN 算法超越了。

3.2.2 YOLO

2015 年 Redmon 等[47]提出了继承 OverFeat 的 YOLO(you only look once)方法，其检测速度达到 45 张/s，速度优势让它成为了端到端的领跑者。YOLO 与 two-stage 类检测方法的主要区别在于使用图像的全局信息来预测目标，将输入图像大小调整为固定的 448 pixel×448 pixel，对不同位置的对象检测效果更好。输入样本会被分成 7 pixel×7 pixel的网格，每个网格单元负责预测对象中心落在该网格中的目标。另一方面，YOLO 只分析最后 7 pixel×7 pixel的特征图谱，导致对小目标的检测质量不佳，难以区分多个目标在同一个网格单元的情况。

为了简化网络结构，YOLO 去掉了提取候选框分支(proposal 阶段)，用一个无分支卷积网络来提取特征、回归候选框和分类，直接预测各网格内的边框坐标以及所有类别的置信度。训练时使用 P-ReLU 激活函数。YOLO 的检测速度较 Faster R-CNN 提高了 10 倍，这一革新使得基于深度学习的目标检测算法满足实时性检测的需求。

3.2.3 SSD

针对 YOLO 算法定位精度差的问题，Liu 等[48]提出了一种结合 YOLO 回归思想与 Faster R-CNN 的 anchor box 机制的 SSD(single shot multi box detector)方法，使用 VGG16 基础网络作为特征提取网络，为检测不同尺度的目标，在前面卷积层输出的特征图中检测小的目标，大的目标则在后面卷积层传递的特征图中被检测。多尺度区域的局部特征图被用于回归整个图像上的所有位置边框，兼顾了 Faster R-CNN 算法边框定位准确和 YOLO 算法快速的优点。用 PASCAL VOC2007 数据集测试 300 pixel×300 pixel 的输入图像，取得 59 frame/s 的运算速度(Titan X 的 GPU)、76.8%的 mAP 的好成绩。

该方法的主要创新点包括多尺度特征检测、匹配策略、修改 VGG16 结构、加入 atrous 算法[49]等；其主要缺点：1)采用多层级特征分类，使末尾卷积层的感受野很大，导致小目标的特征较模糊，不利于检测；2)当没有候选区域时，难以回归，易导致不收敛问题。

3.2.4 DSSD

针对 SSD 算法难以检测小目标的问题，Fu 等[50]提出了 DSSD(de-convolutional single shot detector)方法。用 ResNet101 网络取代 SSD 的 VGG16，增强了网络提取特征的能力。DSSD 借鉴 FPN 的思路，利用反卷积结构从高维空间传递出图像的深层特征，合并浅层信息，让各层次之间的图像语义关系相结合。预测模块结构将各层次之间特征相结合，输出目标预测类别信息。DSSD 有两个特殊结构：预测模块和反卷积模块，前者是为提高准确性和防止梯度直接流入 ResNet 主网络而采用增强每个子任务表现力的方法。

通过实验对比发现，DSSD 对于小目标鲁棒性较差，主要有两个贡献：1)DSSD 用 ResNet 网络取代 SSD 的 VGG16，提取网络特征的能力得以提高；2)用反卷积层增加了大量上下文信息。

3.2.5 YOLOv2/YOLO9000

Redmon 等[51]对 YOLO 网络结构进行了改进，提出了 YOLOv2 和 YOLO9000 方法。YOLOv2 用 Darknet19 作为特征提取网络，并添加 Batch Normalization 进行预处理。该方法训练了一个高分辨率(448 pixel×448 pixel)的分类网络，提高了输入图像的分辨率，mAP 提高 4%。YOLO 通过将单个网格单元合并成全连接层来预测边框，丢失较

多的空间信息，不利于目标定位。YOLOv2 对此进行了改进：1）增加候选框的预测并用强约束定位方法，使算法召回率有了很大提高；2）为更好地检测小目标，融合了图像细粒度特征，使浅层特征和深层特征相结合；3）YOLOv2 借鉴 Faster R-CNN 的 anchor 机制，为得到更好的 anchor 模板，采用 K-Means 聚类方法来聚类计算训练集；4）为使模型更稳定，针对网格单元采取预测其相对坐标位置的方法，再用逻辑函数对 ground truth 进行归一化处理。

3.2.6 RetinaNet

One-stage 方法的准确率不及 two-stage 方法是因为样本类别不均衡。由于 one-stage 方法中的交叉熵损失函数不能抵制类别极不平衡，所以 RetinaNet[52] 采用 focal loss 替换交叉熵损失函数，降低分类良好样本的分类损失，将训练重点放在一组稀疏的样本上，防止在训练期间大量易辨识的负例给检测器带来压制影响。

RetinaNet 是为了评估分类损失的有效性而设计的，主要由 ResNet、FPN 及两个 FCN 子网络组成，效果最好的 RetinaNet 结构是以 ResNet-101-FPN 为基本框架，其在 COCO 测试集上的 mAP 为 39.1%，速度为 5 frame/s，精度超过同期所有 two-stage 的检测器。FPN 针对 ResNet 中形成的多尺度特征进行强化，获得表达力更强、包含多尺度目标区域信息的特征图谱，最后在 FPN 的特征图谱集合上分别使用两个结构相同但不共享参数的 FCN 子网络，完成目标框类别分类和 Bbox 位置回归任务。

3.2.7 YOLOv3

YOLOv3[53] 使用 YOLOv2 的 Darknet53 网络，并与 FPN 网络结构相结合，再由卷积网络得出预测结果。相应改进使 YOLOv3 与 SSD 相当的精确度下达到 22.2 ms/张的速度，并在 COCO test-dev 上 IoU 的阈值为 0.5 时 mAP 值达到 33.0%，与 RetinaNet 的结果相近，速度快了 4 倍，但整体模型变得更加复杂，与速度和精度相互制衡。

3.2.8 One-stage 改进工作

YOLO 核心是用整张图作为网络输入，直接在输出层回归边界框的位置和类别。YOLOv2 舍弃了 YOLO 的全连接层，用 anchor 预测边界框。针对距离近的物体，YOLOv3 有很好的鲁棒性，超越了 SSD512。针对 Faster RCNN 只能在一层特征图上预测目标的问题，SSD 在不同特征图上进行预测，其缺点是不同 anchors 的设置较麻烦。DSSD 的核心思想是利用中间层的上下文信息，并提出了自己的预测模块。RetinaNet 针对类别不均衡问题提出一种新的损失函数，即 focal loss。关于 One-stage 的更多工作：G-CNN[54] 避免直接从大量的候选框中搜索目标，解决了检测效率受限于 proposal 体量的问题。R-SSD[55] 通过增加不同层特征之间的联系和特征金字塔中特征图的数量来缓解 SSD 中候选框对同一个对象进行重复检测和小目标检测效果不佳的问题。DSOD[56] 也是对 SSD 的改进，用 DenseNet 网络提取特征，优化梯度消失、模型参数量大及输入图像信息缺失的问题。RON[57] 综合两类方法的优点，对多尺度目标定位与负样本空间更加关注，能直接预测各个特征图谱所有位置的最后结果。STDN[58] 提出利用尺度转移模块来获取不同分辨率的特征图，既不增加参数和计算量又实现了网络正确率和速度的提升。PFPNet[59] 采用 SSD 的思想来保持运算速度，采用 SPP 的思想并用 MSCA 模块进行特征融合，提高了目标检测效果。M2Det[60] 基于多级特征金字塔网络 MLFPN，结合 SSD，在 COCO 数据集上达到同样的精度时比 YOLOv3 运算速度更快。

4 Anchor-Free 类模型

经典的 YOLO 方法是最早的 Anchor-Free 模型之一，最近较知名的 Anchor-Free 模型有 FASF[61]、FCOS[62]、FoveaBox[63] 等。下面将从早期探索、基于关键点、密集预测三个部分来介绍有代表性的 Anchor-Free 模型。

4.1 早期探索

DenseBox[64] 于 2015 年 9 月在 arxiv 上被发布，DenseBox 基于 VGG19，主要贡献包括三点：1）证明了单个 FCN 网络可以检测出遮挡严重、多尺度的目标；2）提出了新的 FCN 模型，不需区域提议，可被用于训练端到端网络；3）结合了多任务学习，使精度进一步提高。

DenseBox 与 YOLO 有以下区别：1）DenseBox 用于人脸检测，检测类别有人脸和背景两类，而 YOLO 是通用检测，检测类别数超过两类；2）DenseBox是密集逐像素预测，而 YOLO 是先将图片网格化，再预测每个单元格；3）DenseBox 的真值是由 Bbox 中心圆形区域来确定，而 YOLO 的真值是由 Bbox 中心点所在的单元格确定。

4.2 基于关键点

1）CornerNet 与 ExtremeNet

2018 年，Law 等[65] 提出了 CornerNet 算法，主

要贡献包括：1）通过检测 Bbox 的一对角点来检测目标；2）提出角合并（corner pooling），以更好地定位 Bbox 的角点。

ExtremeNet[66]提供了一种目标检测的新方向：先采用标准关键点估计网络来检测 4 个极值点和 1 个中心点，然后通过几何关系对提取到的关键点进行分组，一组极值点对应一个检测结果，使得目标检查问题转化成一个纯粹的基于外观信息的关键点估计问题，避开了区域分类和隐含特征学习。

CornerNet 与 ExtremeNet 的区别在于：1）CornerNet用预测角点来检测目标，而 ExtremeNet 是用预测极值点和中心点来检测目标；2）CornerNet 通过角点 embedding 之间的距离来判断是否为同一组关键点，而 ExtremeNet 通过枚举极值点、经过中心点判断 4 个极值点是否为一组。

2）CenterNet

CenterNet[67]模型的结构简单，用三种内部结构不同的网络进行目标检测，并用三个网络来输出预测值。在训练和检测中，如果两个物体在 ground truth 中的中心点重叠，会被当成一个物体。该模型的思想不仅可用于 2D 目标检测，而且只要略微扩展就可用于 3D 检测和人体姿态识别。这种轻量级的模型对于算力较小的嵌入式端平台有着一定的优势。

CenterNet 相比 CornerNet 有了改进，其检测速度和精度相比 one-stage 和 two-stage 方法都有提高。CenterNet 舍弃了 anchor，没有正负 anchor 的判断。因每个目标只对应一个中心点，该中心点通过 heatmap 预测得到，所以不需 NMS 进行筛选，直接检测目标的中心点和大小。CenterNet 没有 FPN 结构，所有中心点来自一个 feature map，因此分辨率不能太低。

4.3 密集预测

1）FSAF

FSAF 是一种目标探测器的特征选择 anchor-free 模块，它能添加具有 FPN 结构的 single-shot 探测器。FSAF 模块打破了 anchor-based 检测的两个局限性：1）启发式引导特征选择；2）基于覆盖锚取样。FSAF 在 COCO 数据集上的 mAP 达到 44.6%。

FSAF 是基于 FPN 的单阶段方法中的 RetinaNet 网络，如图 14 所示，在每层 feature map 上添加一个与 anchor-based 方法并行的 FSAF 模块，即 anchor-free 分支。FSAF 模块附加两个卷积层，分别执行 anchor-free 分支的分类与回归预测；在线特征选择模块负责检测每个真值的特征图谱。将实例输入到特征金字塔的所有层，然后计算所有 anchor-free 分支焦点损失和 IoU 损失，选择损失和最小的特征层学习实例。训练期间，基于实例信息选择的特征层检测给出的实例。推理阶段，FSAF 模块和 anchor-based 分支单独或者联合运行。

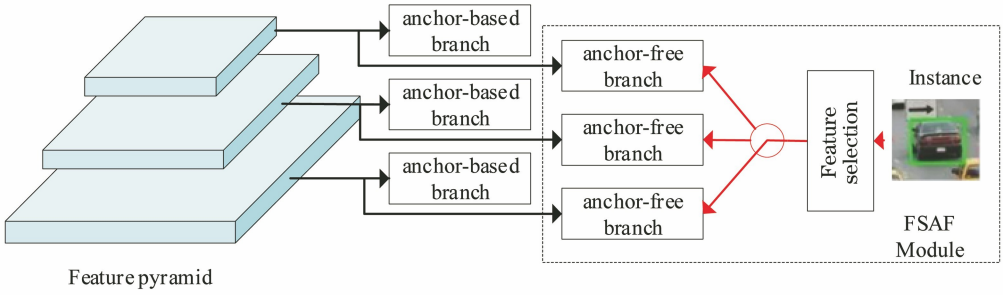

图 14 FSAF 实现流程图

Fig. 14 FSAF implementation flow chart

2）FCOS

FCOS 模型的结构主要包括 FPN 和 box regression、classification、center-ness 三个分支。其中 center-ness 分支可筛除低质量的预测，使性能得到巨大提升，但该分支的优化目标不稳定，训练时 loss 基本不降低，影响其他两个分支的预测。对于 box regression 分支的优化，由于 FCOS 完全不依赖 anchor box，没有 anchor 先验，损失会略高于 RetinaNet 的回归部分，但为了保证算法效果，RetinaNet 需要较多的 anchors，存在正负样本类别不均衡问题，其优点是少了很多超参数，如 anchor number、anchor size、anchor ratio 等；在训练时，box regression 分支省去了 IoU 计算，提速非常大，但它自身的计算时间较长。除此之外，center-ness targets 和 regression targets 的优化也是值得研究的方向。FCOS 的主要优点：1）将检测和其他使用 FCN 的任务统一，重用这些任务的思想；2）proposal free 和 anchor free 减少了超参数的设计；3）不使用

trick，达到了单阶段检测的最佳性能；4）经过小的修改，可将其立即拓展到其他视觉任务上。

3）FoveaBox

FoveaBox 的总体设计思想为 anchor-free，无需定义 anchor 的相关参数，但其他参数不能缺少，如每层区域范围、正样本区域的缩放因子等。但在预测坐标方面，与 DenseBox 和 UnitBox[68] 不同，FoveaBox 未直接学习目标中心到 4 个边的距离，而是学习一个预测坐标与真实坐标的映射关系。

FoveaBox 包括基础网络和两个子网络，子网络分别用于预测分类和预测边界框。主干网络负责计算输入样本上的卷积特征图谱。第一个子网络对基础网络的输出进行逐像素分类；第二个子网络在对应位置上进行 Bbox 预测。与 anchor-based 的方案相比，FoveaBox 有以下优点：1）在每个位置上只预测一个目标，因此输出空间为 anchor-based 方法的 $1/A$，其中 A 是每个位置的 anchor 数量；2）没有位置定义，优化的目标更为直接；3）没有 anchor，使用更灵活，检测网络更加简单直接，扩展性更好，FoveaBox 在 COCO 数据集上的 mAP 为 42.1%。

4.4 Anchor-Free 类模型区别与改进总结

CenterNet 精度高，核心思想是通过中心点抑制误检。用物体中心点对提取到的 bounding box 进行过滤，如果 box 的中间区域没有中心点，则认为此 box 不可靠。CenterNet 用两个角点和一个中心点来表示，Objects as Points[69] 用一个中心点和长、宽值来表示，FCOS 用点到框的 4 个距离来表示，Grid-RCNN[70] 用 $N \times N$ 个点来表示，ExtremeNet 用 4 个极值点和 1 个中心点来表示。Objects as Points 速度快，用中心点完成尽可能多的任务，通过检测中心点及预测各种长宽和 offsets，不仅能进行二维检测，还能进行三维检测。CornerNet-Lite[71] 是 CornerNet 的两种有效变体的组合：CornerNet-Saccade 追求高准确率的同时，尽可能提高速度；CornerNet-Squeeze 追求高实时性的同时，尽可能提高准确率。

FSAF、FCOS、FoveaBox 的相同点包括：1）利用 FPN 来进行多尺度目标检测；2）将分类和回归解耦成 2 个子网络来处理；3）通过密集预测进行分类和回归。FSAF、FCOS、FoveaBox 的区别在于：1）FSAF 和 FCOS 是对到 4 个边界距离进行回归预测，而 FoveaBox 回归预测的是一个坐标转换；2）FSAF 通过在线特征选择来提升性能，FCOS 通过 center-ness 分支剔除低质量 Bbox 以提升性能，FoveaBox 通过只预测目标中心区域来提升性能。

DenseBox、YOLO、FSAF、FCOS、FoveaBox 的相同点是通过密集预测进行分类和回归，区别在于：1）DenseBox、YOLO 只进行单尺度目标检测，而 FSAF、FCOS、FoveaBox 利用 FPN 进行多尺度目标检测；2）DenseBox、YOLO 将分类和定位进行统一，而 FSAF、FCOS、FoveaBox 将分类和回归解耦成 2 个子网络来得到。

5 思考与展望

近年来目标检测的研究主要集中在特征表达增强上，为此相继提出了混合特征图、特征金字塔以及特征注意力单元等，并取得了较好的研究成果。但在目标检测的研究道路上还有很多需要解决的难题，下面将面临的问题和未来发展方向归纳为三方面：

1）网络结构方面

基础网络结构的改进开发也是提高目标检测效果的重要途径之一。针对 Backbone 特征抽取器，可以考虑设计更有效的分类器。结合信息流和特征复用假设使网络层数不断加深，一直是 CNN 发展的主要方向之一。跨网络层连接特征图是卷积神经网络的核心范式，怎样使连接方式更有效也是研究的热点方向。

对于 anchor-based、anchor-free 类模型，可从以下方面进行改进：使 ROI 之间共享更多计算量，充分利用 CNN 提取的特征，从精简网络结构、模型压缩、优化 IoU 和损失函数、软硬件协同设计等方面减小复杂度，提升模型检测性能。对于 anchor-based，预选框参数要根据具体的任务而定，调参费时费力，一定程度上抑制了模型的迁移能力。此外，anchor 设定很重要，要最大可能地覆盖对象尺寸和宽高比，因 anchor 是提前设定的，训练期间不会自适应变化，对此让对象检测器自己学习 anchor，成为了一种新的研究方向。新的实例分割方法 anchor-free 网络结构简单，期待其灵活性带来新的方法和思路，另外针对关键点定位组合的问题和在不使用 ROI Pooling 的情况下解决 feature align 问题可以进行更多的研究。

2）信息融合方面

不同层级语义信息融合：神经网络的低层通常保留了比较多的细节特征；而高层通常有更好的语义特征，从低层特征图和高层语义能得到更多有助于目标检测的信息，分割与语义信息相结合，使之能

得到更准确的预测边界框。所以不同层之间的特征融合也是一个研究热点。

场景信息的融合：目前的目标检测都是把前景从背景中脱离出来，这样场景信息没有被利用，场景信息有着丰富的环境上下文信息，深入分析和充分利用场景信息，可以获取场景的先验知识，减少复杂的背景和与目标相似物体的干扰；此外，利用场景信息为目标检测提供更多的线索，能提升检测与跟踪算法的准确性与鲁棒性。因此，对前景目标信息与背景信息的融合、目标状态与场景信息的融合的研究，将有益于优化算法实用性能。

多层级的与多维度的信息相结合：目前的追踪和检测算法一般是将时域和频域等特征的信息相结合来对运动目标进行检测，但是这些算法就维度和层级而言都比较单一，如何将时间和推理等多维度与特征和决策等多层级的、多源的、互补的信息相结合，以提高检测和追踪的效果也是研究者可以思考的方向。

3）实际应用方面

小目标检测：实际应用中小目标作为一个重要的对象，其分辨率低，像素少，信息量少，训练数据难以标记，目前还没有出现对小目标检测效果比较好的算法。小目标检测一直是该领域具有挑战性的难题，当前小目标检测算法主要是对目标检测算法进行的优化和改进，这成为小目标检测研究的重点方向。

实际应用场景检测：目标检测应用到实际场景中时，怎样获得更多又全面的数据集是一大难题。对于运动目标的检测与跟踪，现有模型主要针对特定环境下的运动目标进行研究，难以应用到复杂的自然环境中去，未来可对该方向进行更多的研究；对于更多的实际应用，如复杂背景中的目标检测、多尺度目标检测、特定任务的目标检测等都将是该领域需要深入研究的课题。

6 结束语

在图像中搜索到感兴趣的对象并确定其大小和位置，即目标检测是机器视觉核心问题之一。各种对象呈现的方式千差万别（如不同的外观、姿态、形状）和采集数据时的环境不同（如光照、重叠等因素），使目标检测成为机器视觉领域中最富有挑战性的任务之一。回顾了机器视觉领域中基于深度学习的目标检测算法及主流框架。对该领域面临的难题和未来发展的方向进行了展望。随着研究人员对深度学习技术在各个图像领域应用的深入研究，新的理论和新的方法也在逐渐增多。基于回归思路的 one-stage 方法和基于候选框生成与分类的 two-stage 方法的结合，以及 anchor-based 类模型和 anchor-free 类模型的结合，获得了不错的进展，这为该领域的研究提供了较好的基础。

参 考 文 献

[1] Yin H P, Chen B, Chai Y, et al. Vision-based object detection and tracking: a review [J]. Acta Automatica Sinica, 2016, 42(10): 1466-1489.
尹宏鹏, 陈波, 柴毅, 等. 基于视觉的目标检测与跟踪综述[J]. 自动化学报, 2016, 42(10): 1466-1489.

[2] Zhang X Y, Gao H B, Zhao J H, et al. Overview of deep learning intelligent driving methods[J]. Journal of Tsinghua University (Science and Technology), 2018, 58(4): 438-444.
张新钰, 高洪波, 赵建辉, 等. 基于深度学习的自动驾驶技术综述[J]. 清华大学学报(自然科学版), 2018, 58(4): 438-444.

[3] Li H B, Xu C Y, Hu C C. Improved real-time vehicle detection method based on YoLOV3 [J]. Laser & Optoelectronics Progress, 2020, 57(10): 101507.
李汉冰, 徐春阳, 胡超超. 基于 YOLOV3 改进的实时车辆检测方法[J]. 激光与光电子学进展, 2020, 57(10): 101507.

[4] Li X, Shi B B, Liu Y, et al. Multi-target recognition method based on improved YOLOv2 model[J]. Laser & Optoelectronics Progress, 2020, 57(10): 101010.
李珣, 时斌斌, 刘洋, 等. 基于改进 YOLOv2 模型的多车辆目标识别方法[J]. 激光与光电子学进展, 2020, 57(10): 101010.

[5] Wang D C, Chen X N, Zhao F, et al. Vehicle detection algorithm based on convolutional neural network and RGB-D images [J]. Laser & Optoelectronics Progress, 2019, 56(18): 181003.
王得成, 陈向宁, 赵峰, 等. 基于卷积神经网络和 RGB-D 图像的车辆检测算法[J]. 激光与光电子学进展, 2019, 56(18): 181003.

[6] Gowsikhaa D, Abirami S, Baskaran R. Automated human behavior analysis from surveillance videos: a survey[J]. Artificial Intelligence Review, 2014, 42(4): 747-765.

[7] Huang K Q, Chen X T, Kang Y F, et al. Intelligent visual surveillance: a review[J]. Chinese Journal of Computers, 2015, 38(6): 1093-1118.
黄凯奇, 陈晓棠, 康运锋, 等. 智能视频监控技术综述[J]. 计算机学报, 2015, 38(6): 1093-1118.

[8] Li S B, Yang J, Wang Z, et al. Review of development and application of defect detection technology [J/OL]. (2019-04-04) [2019-11-26]. https://doi.org/10.16383/j.aas.c180538.
李少波, 杨静, 王铮, 等. 缺陷检测技术的发展与应用研究综述[J/OL]. (2019-04-04) [2019-11-26]. https://doi.org/10.16383/j.aas.c180538.

[9] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[10] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks [C] // Proceedings of the 25th International Conference on Neural Information Processing Systems, December 3-6, 2012, Lake Tahoe, Nevada. New York: ACM, 2012, 1:1097-1105.

[11] Image data set [Online]. [2019-11-11]. http://www.image-net.

[12] Yang G C, Yang J, Li S B, et al. Modified CNN algorithm based on Dropout and ADAM optimizer [J]. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2018, 46(7): 122-127.
杨观赐, 杨静, 李少波, 等. 基于 Dopout 与 ADAM 优化器的改进 CNN 算法[J]. 华中科技大学学报(自然科学版), 2018, 46(7): 122-127.

[13] Nair V, Hinton G E. Rectified linear units improve restricted Boltzmann machines [C] // Proceedings of the 27th International Conference on Machine Learning (ICML), Haifa, 2010: 807-814.

[14] Yang J, Li S B, Gao Z, et al. Real-time recognition method for 0.8 cm darning needle and KR22 bearing based on convolution neural network and data increase [J]. Applied Sciences, 2018, 8(10): 1857.

[15] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [EB/OL]. (2014-09-04) [2019-11-26]. https://arxiv.org/abs/1409.1556v1.

[16] Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(4): 640-651.

[17] Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation [EB/OL]. (2015-05-18) [2019-11-26]. https://arxiv.org/abs/1505.04597.

[18] Badrinarayanan V, Handa A, Cipolla R. SegNet: a deep convolutional encoder-decoder architecture for scene segmentation [EB/OL]. (2015-11-01) [2019-11-26]. http://de.arxiv.org/pdf/1511.00561.

[19] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions [EB/OL]. (2014-09-17) [2019-11-26]. https://arxiv.org/abs/1409.4842.

[20] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition [EB/OL]. (2015-12-10) [2019-11-26]. https://arxiv.org/abs/1512.03385.

[21] Huang G, Liu Z, Laurens V D M, et al. Densely connected convolutional networks [EB/OL]. (2016-08-25) [2019-11-26]. https://arxiv.org/abs/1608.06993.

[22] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation [EB/OL]. (2013-11-11) [2019-11-26]. https://arxiv.org/abs/1311.2524.

[23] van de Sande K E A, Uijlings J R R, Gevers T, et al. Segmentation as selective search for object recognition [C] // 2011 International Conference on Computer Vision, November 6-13, 2011, Barcelona, Spain. IEEE, 2011: 154-171.

[24] He K M, Zhang X Y, Ren S Q, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904-1916.

[25] Girshick R. Fast R-CNN [EB/OL]. (2015-04-30) [2019-11-26]. https://arxiv.org/abs/1504.08083.

[26] Scholkopf B, Platt J, Hofmann T. Multi-task feature learning [EB/OL]. (2006-12-04) [2019-11-26]. https://dl.acm.org/citation.cfm?id=2976462.

[27] Ren S Q, He K M, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[28] Hosang J, Benenson R, Dollar P, et al. What makes for effective detection proposals? [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(4): 814-830.

[29] Dai J, Li Y, He K, et al. R-FCN: object detection via region-based fully convolutional networks [EB/OL]. (2016-05-20) [2019-11-26]. https://arxiv.org/abs/1605.06409?context=cs.

[30] Agrawal P, Girshick R, Malik J. Analyzing the performance of multi-layer neural networks for object recognition [EB/OL]. (2014-07-07) [2019-11-26]. https://arxiv.org/abs/1407.1610.

[31] Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors [J]. Computer Science, 2012, 3(4): 212-223.

[32] Lin T Y, Dollar P, Girshick R, et al. Feature pyramid networks for object detection [EB/OL]. (2016-12-09)[2019-11-26]. https://arxiv.org/abs/1612.03144.

[33] He K M, Gkioxari G, Dollar P, et al. Mask R-CNN [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 386-397.

[34] Xie S, Girshick R, Dollar P, et al. Aggregated residual transformations for deep neural networks [EB/OL]. (2016-11-16) [2019-11-26]. https://arxiv.org/abs/1611.05431.

[35] Peng C, Xiao T, Li Z M, et al. MegDet: a large mini-batch object detector [EB/OL]. (2017-11-20) [2019-11-26]. https://arxiv.org/abs/1711.07240.

[36] Qin L K, Gong Y F, Tang T Q, et al. Training deep nets with progressive batch normalization on multi-GPUs [J]. International Journal of Parallel Programming, 2019, 47(3): 373-387.

[37] Gidaris S, Komodakis N. Object detection via a multi-region & semantic segmentation-aware CNN model[EB/OL]. (2015-05-07)[2019-11-26]. https://arxiv.org/abs/1505.01749.

[38] Kong T, Yao A B, Chen Y R, et al. HyperNet: towards accurate region proposal generation and joint object detection [EB/OL]. (2016-04-03) [2019-11-26]. https://arxiv.org/abs/1604.00600.

[39] Yang B, Yan J J, Zhen L, et al. CRAFT objects from images [EB/OL]. (2016-04-12) [2019-11-26]. https://arxiv.org/abs/ 1604.03239.

[40] Wang X, Shrivastava A, Gupta A. A-Fast-RCNN: hard positive generation via adversary for object detection [EB/OL]. (2017-04-11) [2019-11-26]. https://arxiv.org/abs/1704.03414.

[41] Li Z M, Peng C, Yu G, et al. Light-head R-CNN: in defense of two-stage object detector[EB/OL]. (2017-11-20) [2019-11-26]. https://arxiv.org/abs/1711.07264.

[42] Cai Z W, Vasconcelos N. Cascade R-CNN: delving into high quality object detection [EB/OL]. (2017-12-03) [2019-11-26]. https://arxiv.org/abs/1712.00726.

[43] Singh B, Davis L S. An analysis of scale invariance in object detection-SNIP[EB/OL]. (2017-11-22) [2019-11-26]. https://arxiv.org/abs/1711.08189.

[44] Ghiasi G, Lin T Y, Pang R, et al. NAS-FPN: learning scalable feature pyramid architecture for object detection [EB/OL]. (2019-04-16) [2019-11-26]. https://arxiv.org/abs/1904.07392.

[45] Li Y, Chen Y, Wang N, et al. Scale-aware trident networks for object detection [EB/OL]. (2019-01-07) [2019-11-26]. https://arxiv.org/abs/1901.01892?context=cs.CV.

[46] Sermanet P, Eigen D, Zhang X, et al. OverFeat: integrated recognition, localization and detection using convolutional networks [EB/OL]. (2013-12-21) [2019-11-26]. https://arxiv.org/abs/1312.6229.

[47] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection [EB/OL]. (2015-06-08) [2019-11-26]. https://arxiv.org/abs/1506.02640.

[48] Liu W, Anguelov D, Erhan D, et al. SSD: single shot MultiBox detector[M]// Computer Vision-ECCV 2016. Cham: Springer International Publishing, 2016: 21-37.

[49] Chen L C, Papandreou G, Kokkinos I, et al. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 834-848.

[50] Fu C Y, Lin W, Ranga A, et al. DSSD: deconvolutional single shot detector [EB/OL]. (2017-01-23) [2019-11-26]. https://arxiv.org/abs/1701.06659v1.

[51] Redmon J, Farhadi A. YOLO9000: better, faster, stronger [C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017. Honolulu, HI. IEEE, 2017: 6517-6525.

[52] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[EB/OL]. (2017-08-07)[2019-11-26]. https://arxiv.org/abs/1708.02002v2.

[53] Redmon J, Farhadi A. YOLOv3: an incremental improvement[EB/OL]. (2018-04-08)[2019-11-26]. https://arxiv.org/abs/1804.02767.

[54] Najibi M, Rastegari M, Davis L S. G-CNN: an iterative grid based object detector[EB/OL]. (2015-12-24)[2019-11-26]. https://arxiv.org/abs/1512.07729.

[55] Jeong J, Park H, Kwak N. Enhancement of SSD by concatenating feature maps for object detection [EB/OL]. (2017-05-26) [2019-11-26]. https://arxiv.

org/abs/1705.09587v1.

[56] Shen Z Q, Liu Z, Li G, et al. DSOD: learning deeply supervised object detec-tors from scratch[EB/OL]. (2017-08-03)[2019-11-26]. https://arxiv.org/abs/1708.01241?context=cs.LG.

[57] Kong T, Sun F C, Yao A B, et al. RON: reverse connection with objectness prior networks for object detection [EB/OL]. (2017-07-06)[2019-11-26]. https://arxiv.org/abs/1707.01691?context=cs.

[58] Zhou P, Ni B B, Geng C, et al. Scale-transferrable object detection[EB/OL]. (2019-04-05)[2019-11-26]. http:// openaccess.thecvf.com/content_cvpr_2018/papers/Zhou _ Scale-Transferrable _ Object _ Detection_CVPR_2018_p- aper.pdf.

[59] Lin T Y, Dollar P, Girshick R, et al. Feature pyramid networks for object detection[EB/OL]. (2016-12-09)[2019-11-26]. https://arxiv.org/abs/1612.03144?context=cs.

[60] Zhao Q J, Sheng T, Wang Y T, et al. A single-shot object detector based on multi-level feature pyramid network [EB/OL]. (2018-11-12)[2019-11-26]. https://arxiv.org/abs/1811.04533.

[61] Zhu C C, He Y H, Savvides M. Feature selective anchor-free module for single-shot object detection [EB/OL]. (2019-03-02)[2019-11-26]. https://arxiv.org/abs/1903.00621v1.

[62] Tian Z, Shen C H, Chen H, et al. FCOS: fully convolutional one-stage object detection[EB/OL]. (2019-04-02)[2019-11-26]. https://arxiv.org/abs/1904.01355.

[63] Kong T, Sun F C, Liu H P, et al. FoveaBox: beyond anchor-based object detector [EB/OL]. (2019-04-08)[2019-11-26]. https://arxiv.org/abs/1904.03797.

[64] Huang L, Yang Y, Deng Y, et al. DenseBox: unifying landmark localization with end to end object detection [EB/OL]. (2015-09-16)[2019-11-26]. https://arxiv.org/abs/1509.04874.

[65] Law H, Deng J. CornerNet: detecting objects as paired keypoints[EB/OL]. (2018-08-03)[2019-11-26]. https://arxiv. org/abs/1808.01244v1.

[66] Zhou X Y, Zhuo J C, Krähenbühl P. Bottom-up object detection by grouping extreme and center points[EB/OL]. (2019-01-23)[2019-11-26]. https://arxiv.org/abs/1901.08043v1.

[67] Duan K W, Bai S, Xie L X, et al. CenterNet: keypoint triplets for object detection [EB/OL]. (2019-04-17)[2019-11-26]. https://arxiv.org/abs/1904.08189?context=cs.

[68] Yu J H, Jiang Y, Wang Z Y, et al. UnitBox: an advanced object detection network[EB/OL]. (2016-08-04)[2019-11-26]. https://arxiv.org/abs/1608.01471.

[69] Zhou X Y, Wang D Q, Krähenbühl P, et al. Objects as points [EB/OL]. (2019-04-16)[2019-11-26]. https://arxiv.org/abs/ 1904.07850v1.

[70] Lu X, Li B Y, Yue Y X, et al. Grid R-CNN[EB/OL]. (2018-11-29)[2019-11-26]. https://arxiv.org/abs/1811.12030v1.

[71] Law H, Teng Y, Russakovsky O, et al. CornerNet-Lite: efficient keypoint based object detection[EB/OL]. (2019-04-180)[2019-11-26]. https://arxiv.org/abs/1904.08900.