文章编号：2095-2163(2020)09-0001-05 中图分类号：TP391 文献标志码：A

YoloV3 算法在安全帽检测中的应用

梁思成[1]，徐志明[1]，宋 毅[2]
（1 哈尔滨工业大学 计算机科学与技术学院，哈尔滨 150001；2 哈尔滨华德学院 电子与信息工程学院，哈尔滨 150025）

摘 要： 在施工场景中，正确有效的检测工人佩戴安全帽是工地安全重要的一环。基于人工检测安全帽佩戴情况，既耗时耗力，监管效率也不高。为了降低因为安全帽佩戴问题导致的施工问题，本文采用目标检测算法中的 YoloV3 算法进行安全帽检测，利用闸机处情境照片作为原始数据集，采用 LabelImg 对原始照片进行安全帽区域的人工标注，并确定边界框（bounding box）的个数以及位置，建立了人工标注的安全帽检测的训练数据集和测试集。利用上述的训练数据，训练安全帽检测算法，并进行检测。实验结果显示：本文的安全帽检测算法的 mAP 值达到 98%，检测速率为 20fps，该算法在取得了较高准确率的同时，也满足了实时性的要求。

关键词： 目标检测；安全帽检测；YoloV3；实时监测

YoloV3 application insafety helmet detection

LIANG Sicheng[1], XU Zhiming[1], SONG Yi[2]
（1 School Of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China;
2 School of Electronic Information Engineering, Harbin Huade University, Harbin 150025, China）

【Abstract】 In the construction scene, correct and effective detection of workers wearing safety helmets is an important part of site safety. It is time-consuming and labor-intensive to monitor the wearing of a hard hat based on manual detection, and the supervision efficiency is not high. In order to reduce the construction problems caused by the absence of helmets, this article uses YoloV3 in the target detection algorithm for helmet detection, which uses the scene photos at the gate as the data set and LabelImg to label the photos and determines the number of detection frames and location during model training. The experimental mAP value can reach 98% and the detection rate is 20fps. On the basis of meeting the accuracy rate, a certain real-time performance is guaranteed.

【Key words】 Target detection; safety helmet detection; YoloV3; real-time detection

0 引 言

中国作为基础设施建设大国，数以百万计的工人在工地工作，工地安全问题也存在是整个社会极为关注的一个问题，这关系着无数的家庭和生命。2018 年，全国共发生房屋市政工程事故 734 起，死亡 840 人。其中高处坠落事故 383 起，占据整个事故数量的一半以上，每一起安全事故的发生都是对社会的警醒。安全帽作为施工场地的安全保证，准确高效的检测是帮助施工单位降低事故率的保障。

对于传统的检测方法，依靠监督管理人员的主观判断，浪费人力资源、效率低下，且时效性差，无法实时监控。随着机器学习的热潮不断上涨，目标检测技术日新月异，在行人检测和车辆识别方面得到广泛的应用[1,2]，另外，研究人员也开始了对安全帽佩戴检测的研究。一些研究人员利用传统的机器学习算法，进行了安全帽检测的研究工作。冯国臣等利用机器视觉的相关方法，先判断目标图像是否属于人体[3]，再定位到人体头部进行安全帽检测；胡恬利用小波变换和 BP 神经网络进行人脸检测，有效地提高了安全帽检测算法的稳定性和正确率[4]；刘晓慧等利用肤色检测的方法定位人脸区域[5]，再使用神经网络和支持向量机（SVM）的算法进行安全帽检测。但是，传统的机器学习方法普遍存在着准确率偏低、对环境需求偏高的问题，难以保证检测的速度。

近年来，关于目标检测方向的深度学习得到更多注视，更多学者在此方面投入大量的研究时间，目标检测算法的准确率也在不断提高。基于深度学习的目标检测算法有一步走（one-stage）和两步走（two-stage）的两种策略。在两步走算法方面，2014 年 Ross Girshick 提出了 R-CNN（Region with CNN）网络[6]，该算法抛弃了人工选取特征与滑动窗口，利用选择性搜索算法，首先生成候选区域，然后进行目标预测；2015 年何凯明提出了 SPP-Net 算法[7]，

基金项目： 国家自然科学基金（61672185）；黑龙江省教育科学“十四五”规划 2021 年度重点课题（GJB1421618）。

作者简介： 梁思成（1995-），男，硕士研究生，主要研究方向：图像生成、深度学习；徐志明（1967-），男，博士，教授，主要研究方向：社会计算、自然语言处理；宋 毅（1981-），女，硕士，副教授，主要研究方向：自然语言处理、机器学习。

收稿日期： 2020-05-18

利用空间金字塔池化结构,对整张图片一次提取,运算速度更快;Ross Girshick 于 2015 年提出 R-CNN 的改进版 Fast R-CNN[8],将算法的串行结构改为并行结构,显著地提高检测速度;Shaoqin Ren 等人基于 Fast R-CNN 算法做出改进,提出 Faster R-CNN 算法[9],该算法采用 RPN 网络自行学习生成候选区域,再次减少了参数量和检测所需时间;2017 年何凯明等人提出了 Mask R-CNN 算法[10],加入了图像的语义信息,同时进行目标检测和语义分割的任务。

在一步走算法方面,2013 年 Yann Lecun 等人提出 OverFeat 算法[11],利用多尺度滑动窗口来改善检测效果;2015 年 Joseph Redmon 提出了 Yolo 算法[12],以回归的方式输出目标边框与类别,相比于 one-stage 方法,显著提升了检测速度;2016 年 W Liu 等人提出了 SSD (Single Shot MultiBox Detector) 检测算法[13],改善 Yolo 小目标检测效果的问题,添加了 Anchor 的概念,并将不同分辨率的卷积层进行融合,使小物体的信息加入到高分辨率的特征图,提升同一图片中的小物体识别准确率;Joseph Redmon 在 2016 年和 2018 年分别提出了改进的 Yolo 算法:YoloV2 和 YoloV3[14-15]。这两次的算法改进,使得检测算法的 mAP 值达到了 RetinaNet 的水平,同时获得了更好的检测速度。

本文采用目标检测算法中的 YoloV3 算法,进行安全帽检测的研究工作。首先,利用闸机处情境照片作为原始数据集,采用 LabelImg 对原始照片进行安全帽区域的人工标注,确定边界框(bounding box)的个数以及位置,建立了人工标注的安全帽检测的训练数据集和测试集;然后,利用上述的训练数据集,训练安全帽检测算法,并开展了安全帽检测的实验。实验结果显示:本文的安全帽检测算法的 mAP 值达到 98%,检测速率为 20fps,该算法在取得了较高准确率的同时,也满足了实时性的要求。

1 安全帽检测方法

1.1 整体流程

本文的 YoloV3 算法采用端到端的训练方式,将特征提取、候选框预测、非极大抑制和目标识别等步骤连接在一起,目的是提升该算法的性能。首先,Darknet 网络结构对目标进行特征提取,得到 3 个不同维度的特征图;其次,将输入图片划分成 $S \times S$ 个网格,当一个网格中出现该物体的中心点,那么该网格就负责对该物体进行检测,每个网格都会预测 B 个边界框,每个边界框会输出 5 个对应的参数,即边界框的中心坐标(x,y),宽高(w,h) 以及置信度评分。置信度评分综合反映了当前边界框内存在目标的可能性和边界框预测目标位置的准确性,即:交并比(IOU)。最后的特征图包含两个维度:(1)26 × 26 的网格数。(2)$B \times (5 + Y)$ 的维度。其中 B 是每个网格边界框的数量,Y 是预测物体的类别数量,5 则是边界框的中心坐标(x,y),宽和高(w,h) 以及该框的置信度评分。根据置信度评分,确定该框分类,并由中心坐标,宽和高确定边界框绘制时所在图片位置。

1.2 主干网络与多尺度融和

YoloV3 算法改进了 YoloV2 算法的 Darknet 网络,去掉了 YoloV2 算法所使用的池化层和全连接层,仅保留了一些卷积层(convolution layers)、激活层(leaky relu)、批标准化层(Batch Normalization)。卷积层通过卷积核对目标进行局部感知,提取标志性特征;批标准化层对卷积层输出批量归一化;激活层将批归一化层的输出进行非线性映射。通过改变卷积核的张量不断改变张量尺寸,最终得到三层维度不同特征图。

整个主干网络中包含两个组件:

(1)由全连接层、批标准化层、激活层构成的 Darknet 网络的最小组件 DBL;

(2)借鉴 ResNet 的残差结构,以 DBL 为基础组件得到的残差结构。网络中前 74 层中存在 53 个卷积层,其余为 Res 层。整体构成 Darknet-53 结构,网络中使用一系列 1×1 和 3×3 大小卷积核的卷积层。从 75 层到 105 层是 YoloV3 的特征融合层,输出得到 3 个尺度分别为 13×13,26×26,52×52 大小,每个尺度先得到各自尺度下的特征,在通过卷积核上采样实现不同尺度特征图的融合。yolo 结构如图 1 所示。

1.3 边界框

对于边界框的选取,YoloV3 算法采用 K-means 聚类方法,特征图中的每一个网格都会预测 3 个边界框。每个边界框都存在 3 类预测:

(1)预测框的位置即坐标中心,以及预测框的高度和宽度;

(2)置信度;

(3)预先设定好的类别。

本文的目标检测算法的类别设置为两类。在实验中,最后得到的三层输出的输出维度分别为 13×13×21、26×26×21 和 52×52×21。每个网格单元都会预测 3 个边界框,每个边界框预测 $(x, y, w, h,$ confidence$)$。这 5 个参数分别代表边界框坐标 (x, y)、边界框高度 h、宽度 w 和置信度。三层输出分

别经由 32 倍下采样、16 倍下采样和 8 倍下采样时检测。针对输出下采样倍数的不同，检测到的感受野也不一样，32 倍下采样检测的感受野最大，适合检测大的目标，16 倍下采样输出次之，8 倍下采样最小。16 倍下采样得到的特征图可以看成是浅层特征，在 16 倍下采样所得到的数据基础上，进行一次下采样再上采样得到深层特征。通过这种方式将 16 倍下采样和 8 倍下采样的特征相拼接，16 下采样和 32 倍下采样的特征拼接之后，再进行卷积操作，得到 3 个不同维度的边界框信息，可以同时学习浅层特征和深层特征，使得模型的表达效果更好。

YoloV3 算法对每个边界框逻辑回归分析后，得到分类得分，依据分类得分决定所属类别。在每一类得分中，边界框与真实框越为吻合，得分为 1，表示保留该边界框，倘若与真实边框相差太远，低于所设定阈值，则得分置为 0，表示忽略该边界框。

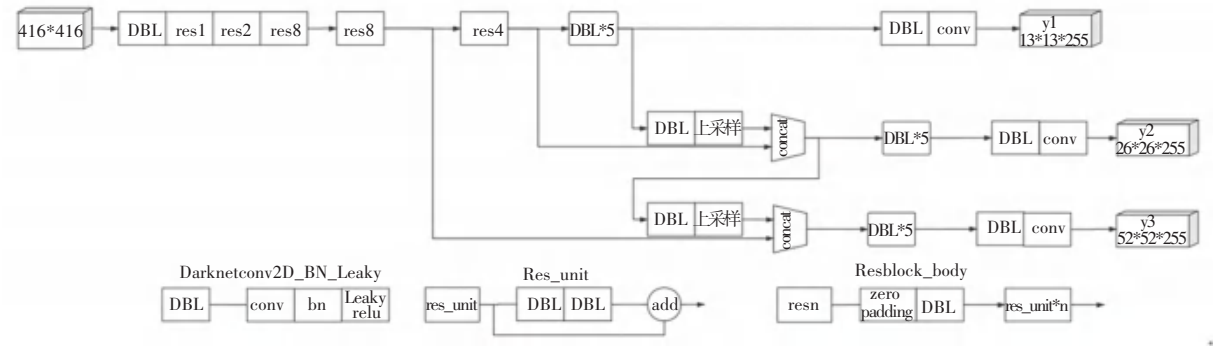

图 1　yolo 结构图

Fig. 1　yolo structure map

1.4　损失函数

YoloV3 算法的损失函数为公式(1)，由 4 部分组成：中心坐标误差、宽高坐标误差以及置信度误差。

$$
\begin{aligned}
Loss = {} & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} [(x_i^j - \hat{x}_i^j)^2 + (y_i^j - \hat{y}_i^j)^2] + \\
& \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left[\left(\sqrt{w_i^j} - \sqrt{\hat{w}_i^j} \right)^2 + \left(\sqrt{h_i^j} - \sqrt{\hat{h}_i^j} \right)^2 \right] - \\
& \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} [\hat{C}_i^j \log(C_i^j) + (1 - \hat{C}_i^j) \log(1 - C_i^j)] - \\
& \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{noobj} [\hat{C}_i^j \log(C_i^j) + (1 - \hat{C}_i^j) \log(1 - C_i^j)] - \\
& \sum_{i=0}^{S^2} I_{ij}^{obj} \sum_{c \in classes} ([\hat{P}_i^j \log(P_i^j) + (1 - \hat{P}_i^j) \log(1 - P_i^j)]) .
\end{aligned}
\tag{1}
$$

其中，λ_{coord} 表示预测数据和标注数据之间的坐标误差，设置为 5；λ_{obj} 表示 IOU 误差权重，取值为 0.5；S^2 表示划分的网格总数；B 表示每个单元格所得边界框的个数；I_{ij}^{obj} 表示该单元格是否负责目标，取值为 0 或 1。

中心坐标误差如公式(2)所示：

$$
\sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} [(x_i^j - \hat{x}_i^j)^2 + (y_i^j - \hat{y}_i^j)^2] . \tag{2}
$$

因为整个网络输出的一部分是中心点坐标 x，y，使用该部分输出通过 sigmoid 激活函数并乘以步长，就可以映射到原始大小 416×416 的图片目标，因此中心坐标误差的实际含义就是当第 i 个网络的第 j 个锚框(anchor box) 负责当前目标，这个锚框产生的边界框就和真实目标进行比较，计算得到中心坐标误差。

宽高坐标误差如公式(3)所示：

$$
\sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left[\left(\sqrt{w_i^j} - \sqrt{\hat{w}_i^j} \right)^2 + \left(\sqrt{h_i^j} - \sqrt{\hat{h}_i^j} \right)^2 \right] . \tag{3}
$$

因为网络输出的一部分是边界框的宽和高 (w，h)。利用(w,h)，计算宽高的误差。同中心坐标误差一样，经过 sigmoid 激活函数再乘以步长，映射到 416×416 大小目标的图片上来计算误差。中心坐标误差的本质就是第 i 个网格的第 j 个锚框负责的一个真实目标，锚框产生的边界框和真实目标去比较，计算得到的误差。

置信度误差如公式(4)所示：

$$
\begin{aligned}
& - \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} [\hat{C}_i^j \log(C_i^j) + (1 - \hat{C}_i^j) \log(1 - C_i^j)] \\
& - \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{noobj} [\hat{C}_i^j \log(C_i^j) + (1 - \hat{C}_i^j) \log(1 - C_i^j)] .
\end{aligned}
\tag{4}
$$

其中，参数置信度 $\hat{C}_i^j$ 表示真实值，其取值为每一个网格的锚框有没有负责这个对象，如果锚框负责该对象，那么 $\hat{C}_i^j = 1$，否则 $\hat{C}_i^j = 0$。因为置信度误差采用交叉熵来表示，无论锚框是否负责这个目标，都会计

算交叉熵。这部分损失函数分为两个部分,有物体和没有物体的部分。为了保证图像中绝大部分不包含待检测物体部分计算的贡献程度偏大,这里引入一个权重系数来减少没有物体计算部分的贡献权重。

第一部分是存在待检测物体的边界框的置信度误差,只有负责待检测对象的边界框,才会计算误差;第二部分是不存在待检测物体的边界框的置信度误差。因为不存在对象,则尽量减低这部分的置信度。如果产生较高置信度,会与真正预测的那个边界框产生混淆。因此,正确对象概率设置为 1,而其他对象概率设置为 0。

分类误差如公式(5)所示:

$$-\sum_{i=0}^{S^2} I_{ij}^{obj} \sum_{c\in classes} ([\hat{P}_i^j log(P_i^j) + (1-\hat{P}_i^j)\, log(1-P_i^j)\ ]). \quad (5)$$

其中,分类误差也选择了交叉熵作为损失函数。当第 i 个网格的第 j 个锚框负责某一个目标待检测对象时,这个锚框产生的边界框才会计算分类损失函数。

在损失函数中,通过训练网络,最终可以获得一张图片目标边界框的中心坐标(x,y)和边界框的宽和高(w,h)、置信度(一般取 1 或 0)和分类概率。当第 i 个网格负责一个真实目标,那么对这个锚框产生的边界框,则求取中心坐标误差、宽高误差、置信度误差、分类误差。如果不对这个目标负责,只需要求取一个置信度误差即可。

2 实验结果

2.1 实验数据集

本文的实验数据集来自于建筑工地闸机处摄像头所拍摄的真实图片,包含正脸、侧脸和背影等不同的角度,拍摄时间包含不同的时间段,以及不同的光照条件。图片以是否佩戴安全帽作为类别区分。其中,包含大量的单样本图片和少量的多样本图片。

利用 LabelImg 对这些原始图片进行安全帽区域的人工标注。在标注过程中,对于多样本图片,以佩戴安全帽的人物的正脸和侧脸为准,决定是否标注,对仅能看见后脑部分、没有明显人脸特征的不予标注;对于佩戴安全帽的人物,以是否有明显人脸特征为准,决定是否标注,当安全帽完全覆盖人脸时,不予标注。因拍摄问题出现人脸残缺时,以露出的人脸部分为准,决定是否标注,露出左半边人脸或者右半边人脸都予以标注,仅露出眼部以下,而没有明确的安全帽特征或者人物额头部分的图片不予标注,露出人脸且明显包含安全帽特征时,给予标注。根据上述的标注标准,得到标注的 xml 文件。每一张图片对应一份 xml 文件,xml 文件中记录了对应图片的标注框的个数、边界值以及类别。最终标记了 17 309 张图片,每张图片根据具体场景不同,具体包含人脸个数不同。对这些标注的数据集随机抽取,训练集和测试集的划分比例为 9:1,即训练集样本 15 353 张,测试集样本为 1 956 张。

2.2 实验设置

通过修改 darknet 在训练 coco 训练集的预训练权重,输出的训练类别数目,以及对应的类别名称,将安全帽的类别个数设定为 2。将原 weights 模型转换成 h5 模型以供训练,训练迭代次数为 100 次。训练过程中保存好权重文件,根据 loss 值来优化调参并确定最优权重,最终在测试时使用 loss 值最小的迭代次数产生的文件作为最终测试权重文件。

YoloV3 算法的评价方法,以 mAP 值作为精确率,表示测试集中识别正确的样本在所有测试样本中所占的比例,计算方式如公式(6);以召回率表示每一类识别正确的样本在所在类别中的占比,计算方式如公式(7),TP(true positive),FP(false positive),FN(false negative)由 IOU 阈值来确定。IOU 是边界框与检测框的交并比。

$$Precision = TP / (TP + FP), \quad (6)$$

$$Recall = TP / (TP + FN). \quad (7)$$

实验中,对测试集中每一张图片的每一个人物头像进行精确率和召回率的计算,所有值按照置信度降序排列,其线下面积就是一类的 AP 值,如公式(8),实际计算中采用平滑处理,对 PR 曲线上每个点的精确值取右侧最大值,如公式(9)。YoloV3 的 AP 计算示意图如图 2 所示。当计算出所有类别的 AP 值,加和求平均,最后得到的就是 mAP。

$$AP = \int_0^1 p(r)\, dr, \quad (8)$$

$$P_{smooth}(r) = \max_{r'=r} P(r'). \quad (9)$$

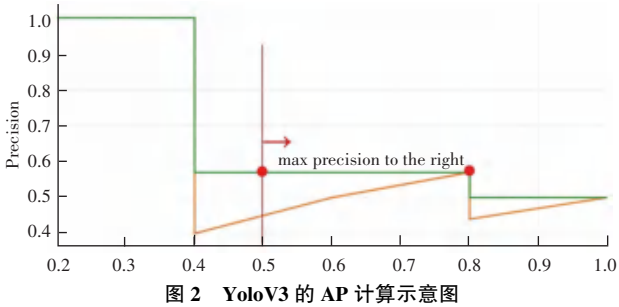

图 2 YoloV3 的 AP 计算示意图

Fig. 2 YoloV3 AP calculation diagram map

2.3 实验结果分析

实验结果如图 3 所示,每一张图片都将所有的

识别出的边界框绘制出来，并以不同颜色加以区分，表示为不同分类。边界框上写出分类类别和所属类别得分，方便使用者进行观察。测试视频来自于施工场地入口一个正面摄像头的记录数据，由于视频角度和光照的缘故，入口处的人脸因为有遮挡导致测试效果不佳，其他位置安全帽佩戴测试效果可以满足整个任务需求。测试视频效果如图 4 所示，输出成视频格式展现效果同图片测试效果。

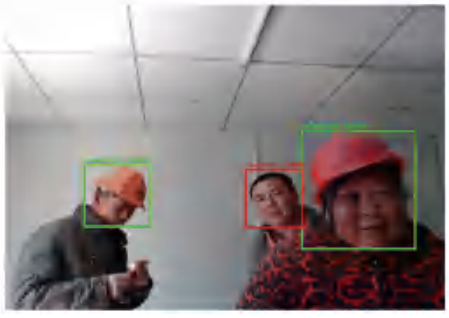

图 3　YoloV3 图片检测效果图

Fig. 3　YoloV3 picture detection effect map

图 4　YoloV3 视频检测效果图

Fig. 4　YoloV3 video detection effect map

测试集的样本数量见表 1，共二类图片 1 956 个文件。其中，戴安全帽的样本为 1 101 个；不戴安全帽的样本为 1 352 个。在检测结果中，戴安全帽识别正确个数为 1 078；不戴安全帽识别正确数量为 1 338 个，识别错误个数分别为 62 个和 72 个。在识别佩戴安全帽与不佩戴安全帽二类任务的准确率均达到 98%以上，测试的平均均值精度（mAP）达到了 98.22%。在测试速度方面，单 GPU 为 GTX1080Ti 的情况下，测试单张图片约为 50 ms，满足了实时性要求。

表 1　安全帽检测结果

Tab. 1　Safety helmet detection result

	戴安全帽	不戴安全帽
样本总数	1 101	1 352
识别正确	1 078	1 338
识别错误	62	72
精确率	98%	99%
AP	98%	99%

3　结束语

本文采用 YoloV3 算法进行施工场地的安全帽检测。首先对施工现场闸机处拍摄的 17 300 余张图片数据进行了人工的数据标注，按着 9∶1 的划分比例，分别建立了训练集和测试集，开展了安全帽检测的实验。实验结果表明，该算法在安全帽检测的准确率和实时性方面均达到了实际应用的需求。

参考文献

[1] 谢林江，季桂树，彭清，等. 改进的卷积神经网络在行人检测中的应用[J]. 计算机科学与探索，2018，12(116)：32-42.

[2] 彭清，季桂树，谢林江，等. 卷积神经网络在车辆识别中的应用[J]. 计算机科学与探索.

[3] 冯国臣，陈艳艳，陈宁，等. 基于机器视觉的安全帽自动识别技术研究[J]. 机械设计与制造工程，2015，44(383)：42-45.

[4] 胡恬. 利用几何分析法和 BP 神经网络进行人脸识别的研究[J]. 计算机工程与设计，2002，23(9)：18-21.

[5] 刘晓慧，叶西宁. 肤色检测和 Hu 矩在安全帽识别中的应用[J]. 华东理工大学学报(自然科学版)(3)：99-104.

[6] GIRSHICK R，DONAHUE J，DARRELL T，et al. Rich feature hierarchies for accurate object detection and semantic segmentation [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014：580-587.

[7] HE K，ZHANG X，REN S，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.

[8] GIRSHICK R. Fast r-cnn [C]//Proceedings of the IEEE international conference on computer vision. 2015：1440-1448.

[9] REN S，HE K，GIRSHICK R，et al. Faster r-cnn：Towards real-time object detection with region proposal networks [C]// Advances in neural information processing systems. 2015：91-99.

[10] HE K，GKIOXARI G，DOLLÁR P，et al. Mask r-cnn[C]// Proceedings of the IEEE international conference on computer vision. 2017：2961-2969.

[11] SERMANET P，EIGEN D，ZHANG X，et al. Overfeat：Integrated recognition，localization and detection using convolutional networks [J]. arXiv preprint arXiv：1312.6229，2013.

[12] REDMON J，DIVVALA S，GIRSHICK R，et al. You only look once：Unified，real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016：779-788.

[13] LIU W，ANGUELOV D，ERHAN D，et al. Ssd：Single shot multibox detector[C]//European conference on computer vision. Springer，Cham，2016：21-37.

[14] REDMON J，FARHADI A. YOLO9000：better，faster，stronger [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017：7263-7271.

[15] REDMON J，FARHADI A. Yolov3：An incremental improvement [J]. arXiv preprint arXiv：1804.02767，2018.