

文章编号:1007-5321(2019)06-0049-09

DOI:10.13190/j.jbupt.2019-180

基于深度学习的 OBD 端口占用状态自动识别算法

苏 东, 余宁梅

(西安理工大学 自动化与信息工程学院, 西安 710048)

摘要: 针对光分路器(OBD)端口占用状态不能自动采集的问题,提出了一种改进型 YOLOv3 算法. 增加第 4 个上采样特征图,提升分辨率下密集小物体检测敏感度;针对端口固定高宽比特征,利用 k -means 聚类算法重新确定目标候选框个数和高宽比;提出软非极大值抑制算法,缓解端口靠近且被遮挡情况下引起的漏检、误检;针对 4 种疑难生产场景下的端口占用状态完成检测. 实验结果表明,改进后的 YOLOv3 准确率达 90.12%,相比原 YOLOv3 提升了 5.17%. 改进后的算法对于端口类物体具有更高的检测准确率.

关 键 词: 光分路器; YOLOv3; 聚类算法; 软非极大值抑制; 特征图

中图分类号: TN913.7

文献标志码: A

Research on Automatic Recognition Algorithm of OBD Port Occupancy State Based on Deep Learning

SU Dong, YU Ning-mei

(School of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China)

Abstract: Because that optical branching device (OBD) port occupancy state can not be automatically acquired, an improved YOLOv3 algorithm was proposed. Firstly, by adding the fourth upsampling feature map, the detection sensitivity of dense small objects at high resolution is increased. Secondly, for characteristics of the fixed height-width ratio of port, the k -means clustering algorithm is used to re-determine the number of target candidate boxes and the height-width ratio. Thirdly, the soft non-maximum suppression algorithm is proposed to alleviate the missed detection and false detection caused by the proximity and occlusion of the port. Finally, four difficult production scenario in the port occupancy state is detected to verify the performance of the improved YOLOv3 algorithm. Experiments show that the accuracy of the improved YOLOv3 is 90.12%, 5.17% higher than the original YOLOv3. In conclusion, the improved algorithm has higher detection accuracy for port-like objects.

Key words: optical branching device; YOLOv3; clustering algorithm; soft non-maximum suppression; feature map

光分路器(OBD, optical branching device)是固定宽带接入网中用于对光进行分波与合波的器件,通常作为末端接入设备使用,每个 OBD 端口可接入一个宽带客户,所以端口数量通常作为承载宽带客户规模的关键网络指标. 如图 1 所示,OBD 是一种

基于石英基板的集成波导无源光功率分配器件,基于目前的技术能力无法像有源器件一样通过系统对端口占用状态进行远程自动采集,所以运营商只能依靠维护人员现场核查、系统记录的方式进行 OBD 端口资源管理. 通过对某省主流宽带运营商的调研

收稿日期: 2019-09-03

作者简介: 苏 东(1979—), 男, 硕士生.

通信作者: 余宁梅(1963—), 女, 教授, 博士生导师, E-mail: yunm@xaut.edu.cn.

发现,随着宽带客户的不断增长和竞争的不断加剧,宽带客户由于更换运营商、乔迁等原因导致 OBD 端口变更比例约 30%,端口资源的人工同步更新周期长,且成本高,造成端口资源准确率长期在 70% 左右,不准确的端口无法重新投入使用而成为巨额的沉默成本。2018 年 10 月 23 日市场研究公司 Point Topic 在德国柏林举办的宽带世界论坛 (BBWF, broadband world forum) 上公布了最新宽带运营数据。全球固定宽带在同年 3 季度达 10.8 亿,使用 OBD 接入的光宽带客户数达 6.8 亿。依据运营商 50% 的平均端口配置比例,全球已建成 13.6 亿 OBD 端口,结合上层网络建设成本单端口造价约 80 美金。目前人工资源管理准确率仅有 70%,将有约 4 亿个不准确端口无法投入使用,沉默成本约 320 亿美金。因此,利用目标检测来实现 OBD 占用状态自动识别,替代人工现场采集,进而实现端口资源管理自动化,对于固定宽带运营降本增效有着十分重要的意义。



图1 OBD 图像

1 研究背景及相关工作

目标检测技术在计算机视觉领域有着广泛的应用,比如在交通检测、医疗等行业都有着举足轻重的价值。目标检测技术主要分为传统图像识别方法和基于深度学习的方法两大类。传统图像识别的方法依赖人为设计目标特征,如方向梯度直方图 (HOG, histogram of oriented gradient)^[1]、尺度不变特征变换 (SIFT, scale-invariant feature transform)^[2] 等。然后利用设计的特征来训练分类器进行目标检测,如支持向量机 (SVM, support vector machine)、朴素贝利

叶、决策树、极限学习机 (ELM, extreme learning machine) 等^[3]。但人为设计的特征泛化性差,运算复杂度高,限制了实际应用。而基于深度学习的方法需要预先建立一个图像训练库作为学习模型,并利用卷积神经网络从图像像素中学习目标特征,通过特征训练获得检测模型,实现目标检测的目的。基于深度学习的方法不需要人为设计特征,只需针对不同检测场景来训练相应模型,其在检测精度以及泛用性都比传统图像识别方法更优。

基于深度学习的目标检测方法中可分为 one-stage^[4] 算法和 two-stage^[5] 算法。one-stage 算法也称作端到端算法,算法直接产生物体的类别概率和位置坐标值,代表算法如 YOLO^[6] 系列和 SSD^[7] 等。two-stage 算法将检测划分为 2 个阶段,首先产生候选区域,然后对候选区域分类,代表算法是 R-CNN 系列,如 R-CNN^[8]、Fast R-CNN^[9]、Faster R-CNN^[10] 等。目标检测模型的主要性能指标是检测准确度和速度,对于准确度,目标检测要考虑物体的定位准确性,而不单单是分类准确度。一般情况下,two-stage 算法在准确度上有优势,而 one-stage 算法在速度上有优势。由于 one-stage 算法在速度上更有优势,所以更具备工程化的优势。YOLO 系列中的 YOLO 对相互靠近甚至重叠的物体以及小物体检测效果并不好,且泛化能力偏弱。而针对上述问题,YOLO 系列的改进算法 YOLOv2^[11] 和 YOLOv3^[12] 先后被提出。在速度和准确率上综合表现最好的 YOLOv3 目前在密集鱼群检测、莲蓬密集种植等密集小物体领域的应用效果提升明显。

将 OBD 端口定位、状态识别转化为目标检测任务,基于 YOLOv3 算法针对 OBD 特点进行了相应改进。主要是增加第 4 个上采样特征图优化了跨尺度预测结构,通过聚类分析得到初始目标候选框个数与高宽比维度,提出了软非极大值抑制算法过滤候选框。在实验阶段采用迁移学习思想在 COCO (common objects in context) 数据集完成网络粗调,建立 OBD 专用数据集并采用数据增强完成数据集扩充,进一步完成网络精调,期间采用动态调整策略提升学习速率。最后基于测试库开展对比实验,证实了改进后算法的有效性与可靠性。

2 基于深度学习的 OBD 端口检测算法

OBD 是固定宽带网络中应用最多的设备。OBD 端口检测是指对 OBD 图片的端口进行定位并给出

端口的占用状态. 通过引入图像识别技术对 OBD 的端口进行检测,可以解决运营商对 OBD 端口资源管理,实现自动化的技术瓶颈. OBD 端口检测是一个典型的目标检测问题.

2.1 算法选择

OBD 端口检测的难点主要包括:1)检测环境的复杂性,OBD 一般存放在机房以及机箱中,杂乱的尾纤与标签导致的遮挡、灰尘导致的异色老化、机房光线不足导致的拍摄光照不足等问题使 OBD 端口变得难以检测;2)OBD 端口自身较小,且排列密集,目标框之间容易重叠,造成漏检、误检概率较大;3)OBD 端口与其连接的尾纤插头种类繁多,造成结构、颜色不统一、不规则等,间接增加了检测的难度;4)在实际生产场景下,当维护人员进行 OBD 端口占用状态的图像采集时,需要实时查看检测结果来确定是否调整拍摄角度或者调整拍摄参数,所以对算法检测速度要求较高.

针对以上特性,YOLOv3 与 RCNN、SSD 等算法相比,在检测准确率与检测速度的均衡性上优势明显,所以 OBD 的目标检测问题选用 YOLOv3 更合适.

2.2 算法优化

YOLOv3 的基础网络 Darknet-53 结构如图 2 所示,其主要是由一系列的 1×1 、 3×3 卷积层组成.其中每个卷积层后都会连接一个 Batch Normalization 层和一个 LeakyReLU 层.常用的输入图像尺寸

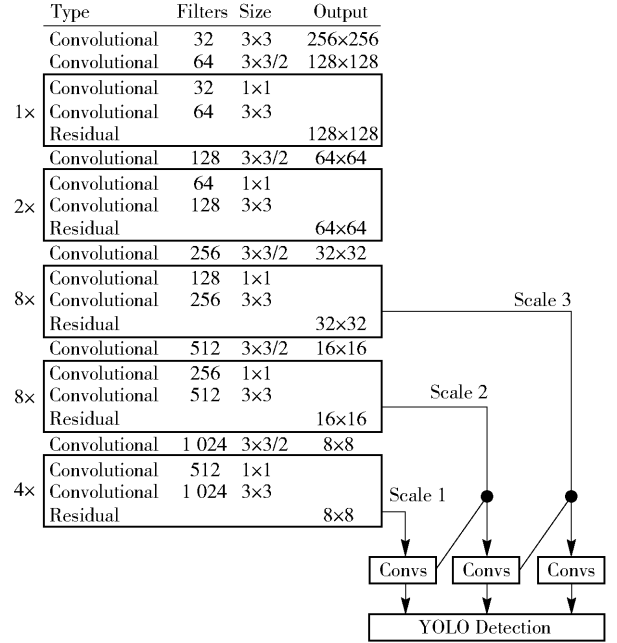


图 2 YOLOv3 的网络结构

是 312×312 、 416×416 及 608×608 均为 32 的倍数.

实际生产环境中普遍存在标签、尾纤等物体遮挡,连续多端口占用或连续多端口空闲的疑难场景,经常会造成漏检、误检.同时端口图像具有的高分辨率下小物体密集排列的特点,进一步增大了漏检、误检概率.因此,对 YOLOv3 算法进行了四尺度特征融合、目标候选框维度聚类、软非极大值抑制 (SoftNMS, soft non-maximum suppression)^[13] 过滤边界框三方面改进,具体算法流程如图 3 所示.

2.3 跨尺度预测优化

YOLOv3 通过结合 3 个不同尺度下的特征图进行检测,如图 4 所示.实际生产场景中输入图像多为 $1 088 \times 1 280$,甚至更大,在检测阶段会结合 32×32 、 16×16 以及 8×8 的特征图来对目标进行检测.尽管已经采用 3 个尺度的特征图来进行目标检测,但特征图的尺寸 (32、16、8) 相比于原图尺寸 $1 088 \times 1 280$ 已大大减少,高分辨率的特征在这 3 个尺度下已经得不到较好地体现.针对 OBD 图像高分辨率下小物体密集排列导致的检测准确率降低的问题,提出增加第 4 个尺度 64×64 的上采样特征图来进行检测.相对于原来 3 个尺度的特征图,4 个尺度的特征图能更好地保留高分辨率下的特征,尤其对于小目标的敏感度提升比较明显.

2.4 目标候选框维度聚类

YOLOv3 默认的目标候选框数量、高宽比维度是通过 VOC 20 类与 COCO 80 类数据集聚类获得,对通用领域的物体比较适用,但对 OBD 端口这类小目标密集排列的专用领域的物体并不适用.通过 OBD 图像分析发现,端口自身的高宽比为固定值且不会受到拍摄角度、多样化场景的影响,针对该特征可利用 k 均值聚类算法 (k -means) 基于训练数据集进行聚类,重新获得适应于 OBD 端口预测的最优目标候选框个数、高宽.聚类分析使用交并比 (IOU, intersection over union),即候选框与真实框交集与并集的比值,从而减小候选框产生的误差,距离函数如:

$$d(\text{box}, \text{centroid}) = 1 - \text{IOU}(\text{box}, \text{centroid}) \quad (1)$$

聚类的目标函数如:

$$S = \min \sum_{i=0}^k [1 - \text{IOU}(\text{box} - \text{truth})] \quad (2)$$

其中: k 为 anchor 个数,box 为候选框,truth 为目标真实框.如图 5 所示,当 k 大于 9 时,目标函数变化

Darknet53框架

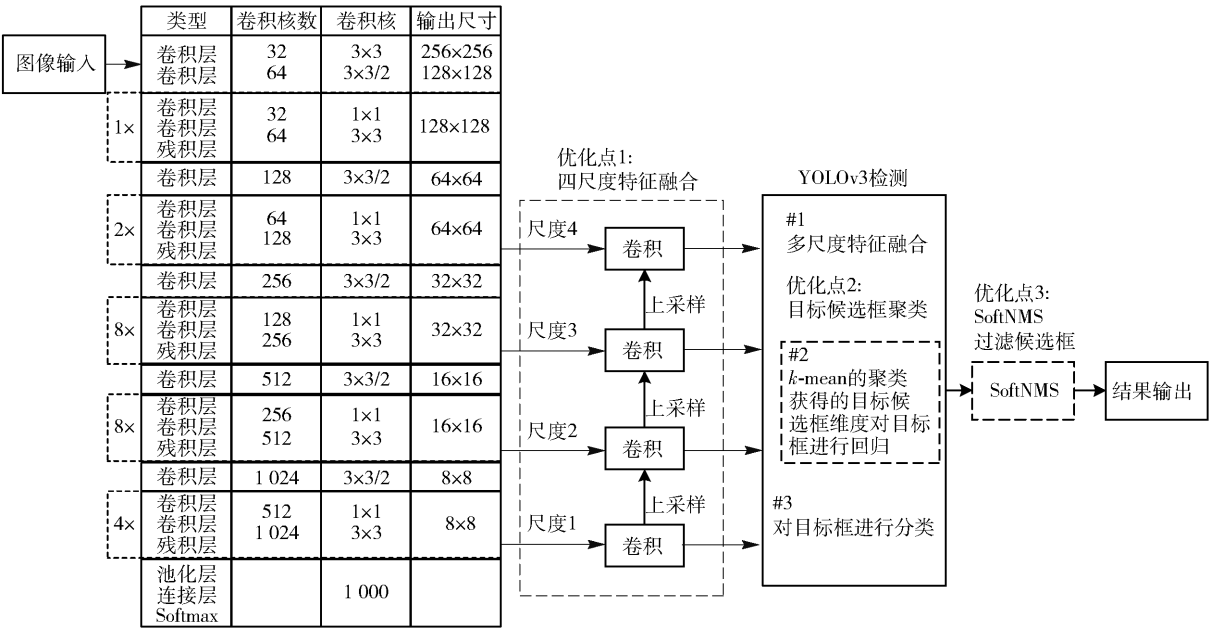


图 3 改进型 YOLOv3 算法流程

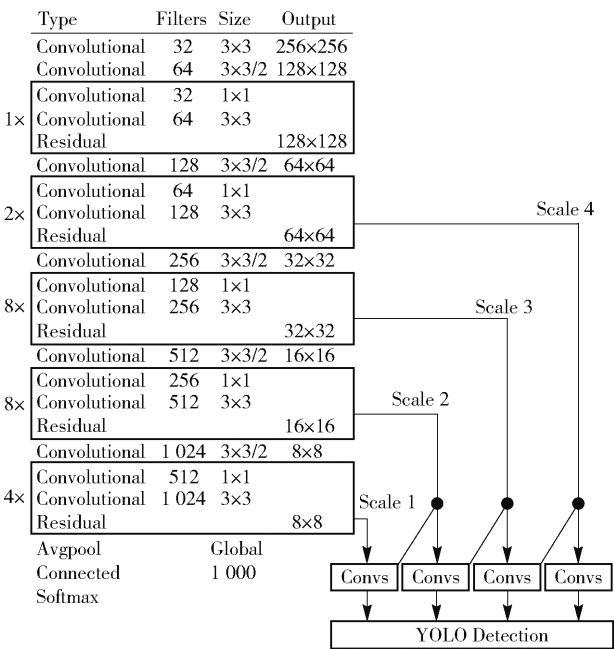


图 4 优化后的 YOLOv3 的网络结构

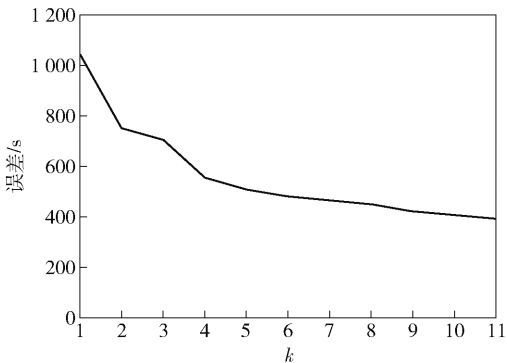


图 5 聚类误差变化曲线

表 1 OBD 候选框参数

序号	1	2	3	4	5	6	7	8	9
新候选框高	31	32	42	29	25	37	44	63	165
新候选框宽	30	40	37	90	115	103	136	174	40
原候选框高	10	16	33	30	62	59	116	156	373
原候选框宽	13	30	23	61	45	119	90	198	326

曲线逐步趋于平缓,所以选择 k 值为 9,即 anchor 个数为 9.

同时可以获得候选框的高、宽聚类结果,当 k 为 9 时如表 1 所示.

2.5 基于软非极大值抑制算法的边界框过滤

当出现 2 个被检测的物体较为靠近、被遮挡的情况下,使用非极大值抑制算法容易过滤掉分数较低的边界框,容易导致漏检情况.然而,非极大值抑

制算法中的阈值若设置过小会导致漏检,设置过大会发生误检.因此,提出了软非极大值抑制算法来解决上述问题.

软非极大值抑制算法如图 6 所示. 3 个输入 B 、 S 、 N_i 分别表示初始的边界框集合、边界框对应的分数集合以及非极大值抑制的阈值. 集合 D 用于存放最终的检测框,在集合 B 非空的前提下,搜索分数集合 S 中数值最大的数,假设其下标为 m ,那么 b_m

就是对应的边界框存入 M 集合,然后将 M 和 D 集合合并,并从 B 集合中去除 M 集合,再循环集合 B 中的每个边界框. 若是非极大值抑制算法,当集合 B 中的边界框 b_i 和集合 M 的 IOU 值大于阈值 N_i 时,就从集合 B 和集合 S 中去除该边界框. 可以看出,非极大值抑制算法采用了较为简单的方式对边界框的评分进行 s 评分,重评分函数如式(3)所示:

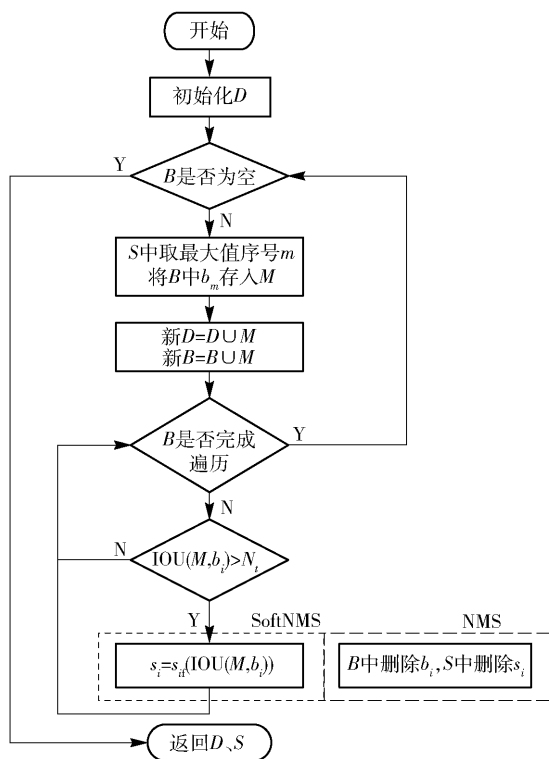


图6 软非极大值抑制算法流程

$$s_i = \begin{cases} s_i, & \text{IOU}(M, b_i) < N_i \\ 0, & \text{IOU}(M, b_i) \geq N_i \end{cases} \quad (3)$$

非极大值抑制算法通过目标边界框列表及其对应的置信度得分列表设定对应阈值,用阈值来删除重叠较大的边界框。

若是软非极大值抑制算法,对于 B 中的边界框 b_i 同样先计算其和集合 M 的 IOU,然后将该 IOU 值作为阈值函数的输入,最后和边界框 b_i 的分数 s_i 相乘作为该边界框 b_i 的最后分数. 显然相对于非极大值抑制算法,软非极大值抑制算法对边框的重评分函数更为平滑。

软非极大值抑制算法中的平滑函数可以分为2种:一种是线性加权函数;另一种是高斯加权函数. 这2种方法均是为了将非极大值抑制算法中的重评分函数进行平滑化. 引入高斯加权函数,得到新的重评分函数如

$$s_i = s_i e^{-\frac{\text{IOU}(M, b_i)^2}{\sigma}}, \forall b_i \notin D \quad (4)$$

软非极大值算法引入高斯加权函数后,通过 IOU 值计算一个因子,该因子的计算函数区别于非极大值抑制算法中的不连续函数,该函数具有连续性. 函数变化曲线如图7所示,当采用图7(a)中线性加权函数时,IOU 未达阈值,该目标边界框分数为正常结果;当 IOU 达到阈值,则对该目标边界框分数置0. 当采用图7(b)中高斯加权函数时,目标边界框遵循 IOU 越大得分越低的原则. 该因子与对应边界框分数乘积得到该边界框的最终分数,从而起到抑制多余边界框的作用。

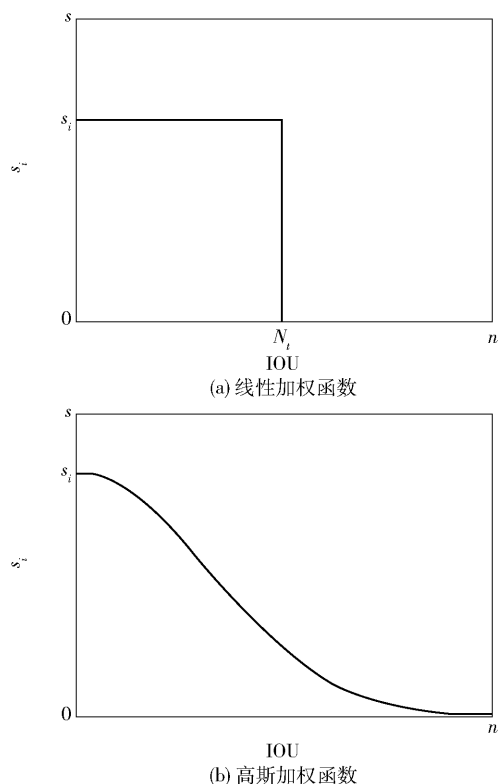


图7 2种加权函数的变化曲线

3 网络训练与实验

3.1 改进型 YOLOv3 训练策略

借鉴迁移学习^[14-15]思想,第1步,基于 COCO 数据集^[16]对改进型 YOLOv3 完成网络粗调. 避免数据集过小导致过拟合现象,减小网络训练中陷入局部最优解的机会,检测准确率可提高3%左右. 第2步,基于 OBD 数据集完成网络精调. 为确保网络训练过程中损失函数曲线平稳收敛,采用动态调整学习速率的方法进行网络训练,学习速率调整为

$$\alpha = d \times \alpha \quad (5)$$

其中: α 表示学习速率, d 表示衰减系数, $0 \leq d \leq 1$. 具体做法为通过设置衰减系数,在一定次数的迭代之后动态更新学习速率. 通过 10 万步完成训练,损失函数阈值最终为 0.000 01,函数曲线稳定且收敛,效果较好.

3.2 超参数设置

实验使用的数据集是在施工现场采集到的图像数据,该批图像经过筛选,能覆盖 90% 以上的 OBD 施工场景,具有一定的代表性,数据集一共 421 张样本. 神经网络需要大量数据训练才能保证结构的可靠性,为了提高网络的检测准确率,防止出现过拟合的现象,这里使用了数据增强来扩充训练集的规模. 如表 2 所示,用到的数据增强方法包括随机旋转、随机调整曝光度、饱和度. 通过数据增强后的数据样本量达 5 052 张.

表 2 数据增强的超参数设置

批次大小	学习速率	衰减系数	旋转角度	饱和度	曝光
32	0.001	0.1	0.2	1.5	1.5

为了保证训练过程能够平稳收敛,在训练时对学习速率进行动态调整,通过设置衰减系数,每经过一定的迭代次数后,动态更新学习速率,训练中在 3.5 万、4 万步时分别进行了一次衰减.

改进型 YOLOv3 在训练过程中支持最小的图像尺寸 320×320 到最大的图像尺寸 608×608 为训练输入. 网络使用中为了降低网络输出的差异,在预测阶段,均将图像调整为 608×608 作为输入. batch size 是参考本实验平台硬件参数设定了每一步的输入为 32 个样本. 在优化器的选择上,采用了 Adam 优化器^[17].

3.3 实验配置与训练结果

实验软硬件配置如表 3 所示.

表 3 软件硬件配置表

名称	相关配置
CPU	Intel(R) Core(TM) i7-8700K 3.70 GHz
内存	DDR4 2 400 MHz EEC Reg 128 GB
GPU	NVIDIA GeForce GTX 1080Ti 11G GDDR5 x2
GPU 加速器	CUDA9.0, CUDNN7.1.3
操作系统	Ubuntu16.04
数据处理	Python, OpenCV
学习框架	DARKNET

图 8 为网络训练 4.5 万步的损失值收敛曲线. 前期损失值曲线波动较大,从 4 万步开始收敛稳定,所以只截取了 4.5 万步,从整个损失函数变化曲线的收敛情况来看,网络训练效果还是比较理想.

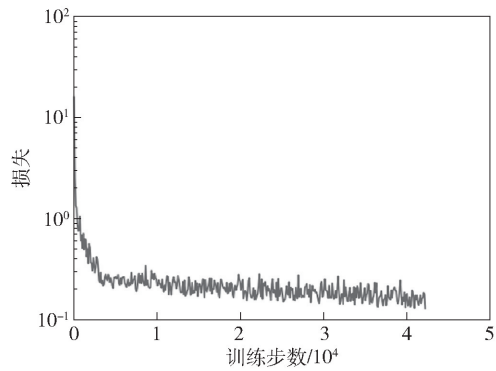


图 8 改进型 YOLOv3 训练的损失变化曲线

4 实验结果

4.1 跨尺度预测优化前后性能对比

增加第 4 个尺度为 64×64 的特征图能更好地保留高分辨率下的特征,便于对高分辨率的特征进行捕捉,尤其是端口类的密集排列小物体检测效果提升明显. 由表 4 可知,mAP 可提升 2.98%.

表 4 跨尺度预测优化前后 mAP 比较

实验方案	样本集	占用端口 AP	未占有端口 AP	mAP
3scales + NMS	训练	0.982 3	0.991 2	0.986 7
3scales + NMS	测试	0.844 3	0.854 8	0.849 5
4scales + NMS	训练	0.993 4	0.991 6	0.992 5
4scales + NMS	测试	0.872 3	0.886 4	0.879 3

4.2 目标候选框维度聚类前后性能对比

输入新候选框以及原候选框的高、宽进行平均 IOU 计算,由表 5 可知,新候选框的平均 IOU 相比原候选框的平均 IOU 提升了 18.7%,可见新候选框对 OBD 端口适应性更好.

表 5 候选框方案对比

候选框方案	anchor 个数	IOU
改进前	9	0.614
改进后	9	0.801

4.3 边界框过滤算法优化前后的性能对比

由表 6 可知,在不增加第 4 个上采样特征图的情况下使用软非极大值抑制算法过滤边界框,mAP 提升 0.58%. 主要体现在对重叠目标检测性能的提

升上,在样本中 OBD 出现大部分重叠端口属于低频情况,更多是部分重叠,因此对性能参数上的提升不是特别明显. 在增加第 4 个上采样特征图的情况下使用软非极大值抑制算法过滤边界框,mAP 相比原 YOLOv3 可提升 5.17%. 可以看出,四尺度融合特征图与软非极大值抑制同时使用的提升幅度相比 2 个优化点单独使用的性能提升累加值要高 1.61%. 说明算法优化点的联合改进对高分辨率下端口类小目标密集排列的情况效果更明显.

表 6 边界框预测算法优化前后 mAP 对比

实验方案	样本集	占用端口 AP	未占用端口 AP	mAP
3scales + NMS	训练	0.982 3	0.991 2	0.986 7
3scales + NMS	测试	0.844 3	0.854 8	0.849 5
4scales + NMS	训练	0.993 4	0.991 6	0.992 5
4scales + NMS	测试	0.872 3	0.886 4	0.879 3
3scales + SoftNMS	训练	0.990 1	0.993 2	0.991 6
3scales + SoftNMS	测试	0.849 4	0.861 2	0.855 3
4scales + SoftNMS	训练	0.995 2	0.993 9	0.994 5
4scales + SoftNMS	测试	0.889 3	0.908 8	0.901 2

对各个模型的验证集测试结果绘制 ROC (receiver operating characteristic) 曲线,以未占用端子作为负样本,占用端子作为正样本. ROC 曲线给出的是当阈值变化时假阳率和真阳率的变化情况. 左下角的点所对应的是将所有样例判为反例的情况,而右上角的点对应的则是将所有样例判为正例的情况. 理想情况下,最佳分类器应该尽可能地处于左上角,这就意味着分类器在假阳率很低的同时获得了很高的真阳率. 由图 9 可见,4 个模型分类器都能认定为较好的分类器,其曲线都处于图中左上角. 改进的模型对 OBD 样本有更好的 AUC (area under curve) 值,即 ROC 曲线下的面积更大. 是 4 个模型分类器中最好的分类器.

4.4 改进型 YOLOv3 的 5 折交叉验证分析

为了验证提出模型的可靠性和稳定性,对改进型 YOLOv3 进行了 5 折交叉验证分析. 将全量样本平均分为 5 个子集,每个子集均做一次测试集,其余的作为训练集. 交叉验证重复 5 次,实验结果如表 7 所示. 由验证数据可以看出,新模型具有较好的稳定性,在每个测试集获得的 mAP 数据相差不大. 另外,通过交叉验证获得的平均 mAP 与对比实验中的 mAP 相差相吻合.

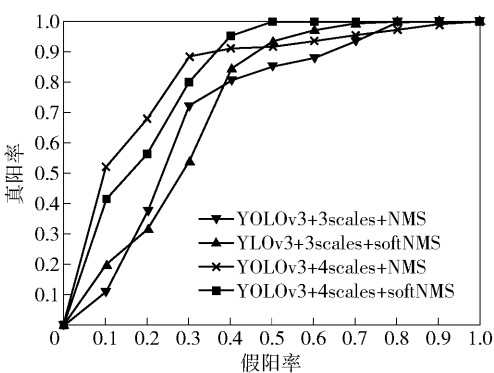


图 9 ROC 曲线对比图

表 7 改进型 YOLOv3 的 5 折交叉验证数据对比

测试集	占用端口 AP	未占用端口 AP	mAP
Fold1	0.890 1	0.910 5	0.900 3
Fold2	0.881 1	0.912 4	0.896 8
Fold3	0.903 7	0.911 3	0.907 5
Fold4	0.926 0	0.914 0	0.920 0
Fold5	0.921 7	0.908 3	0.915 0
平均	0.904 5	0.911 3	0.907 9

4.5 算法改进前后检测效果对比

改进后的算法有效地缓解了遮挡、连续多端口占用、连续多端口空闲等疑难场景下产生的误检、漏检问题. 检测输出图像中蓝色方框表示空闲,红色方框表示占用. 如图 10 所示,图(a)中当左起第 3 端口到第 7 端口连续出现空闲端口密集排列时,左起第 6 端口被漏检,即无检测方框出现. 图 10(b)为改进后算法可正确检测的效果,即红色方框标注. 可以看出,在未占用端口密集排列场景下,原 YOLOv3 方法容易导致漏检,而改进型 YOLOv3 能更好地检测到全部未占用端口. 另外,得益于软非极大值抑制算法的优点,即使出现目标框之间部分重叠的情况,目标框也能更好地框住目标.

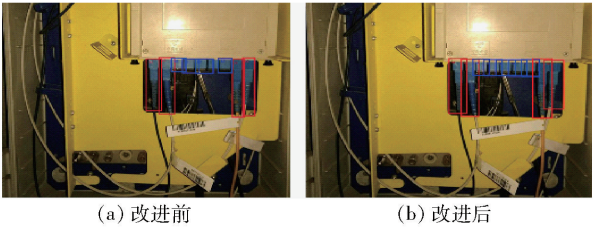


图 10 端口连续空闲下算法改进前后漏检纠正的检测效果

如图 11 所示,图(a)中当左起第 2 端口到第 9 端口连续出现占用端口密集排列时,左起第 7 端口

被漏检,即无检测方框出现。图(b)为改进后算法可正确检测的效果,即红色方框标注。可以看出,图中检测的主要目标为占用端口,由于占用端口具有一定长度,若 OBD 安装带有小角度的倾斜,会导致目标框之间部分重叠的情况,改进后的 YOLOv3 能更好地处理目标框部分重叠的情况。

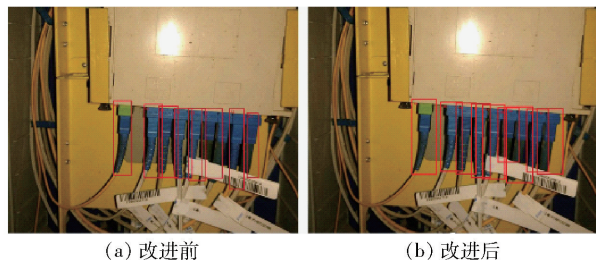


图 11 端口连续占用下算法改进前后漏检纠正的检测效果

如图 12 所示,图(a)为改进前检测效果,右起第 2 端口在白色标签遮挡情况误检为空闲状态,即蓝色方框标注。图(b)为改进后算法可正确检测出占用状态,即红色方框标注。如图 13 所示,图(a)中左起第 1 端口在黄色尾纤遮挡情况下被漏检,即无检测方框出现。图(b)为改进后算法可正确检测左起 1 端口占用,即红色方框标注。可以看出,在遮挡场景下,改进后的 YOLOv3 对误检、漏检问题改善明显。

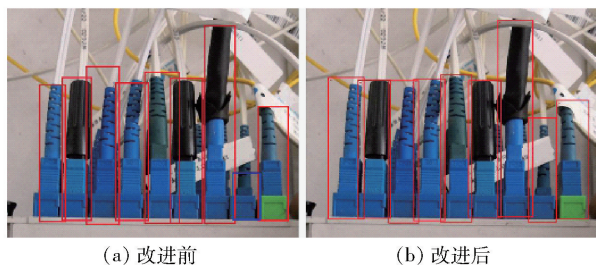


图 12 遮挡场景下算法改进前后误检纠正的检测效果

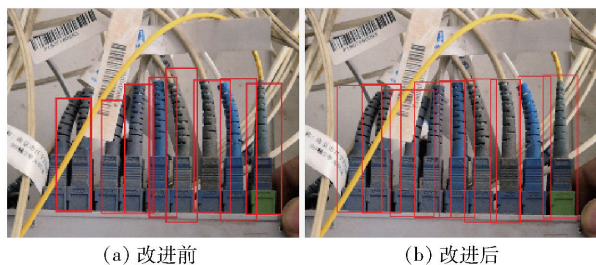


图 13 遮挡场景下算法改进前后漏检纠正的检测效果

5 结束语

针对 OBD 端口占用状态自动识别开展研究。

通过对深度学习算法 YOLOv3 进行改进完成检测,主要改进点为增加第 4 个上采样特征图、目标候选框维度聚类、软非极大值抑制。改进后的 YOLOv3 算法在 OBD 端口占用状态自动识别测试中准确率高 90.12%,相比 YOLOv3 提升了 5.17%。替代目前人工采集方式后固定宽带网络资源准确率可提升 20.12%,若全球范围内推广应用可大幅节省成本,具有广泛的适用性。

参考文献:

- [1] Llorca D F, Arroyo R, Sotelo M A. Vehicle logo recognition in traffic images using HOG features and SVM [C] // 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013). Hague: IEEE Press, 2013: 2229-2234.
- [2] Lowe D G. Distinctive image features from scale-invariant keypoints [J]. International Journal of Computer Vision, 2004, 60(2): 91-110.
- [3] 闫敬文, 陈宏达, 刘蕾. 高光谱图像分类的研究进展 [J]. 光学精密工程, 2019, 27(3): 680-693.
Yan Jingwen, Chen Hongda, Liu Lei. Overview of hyperspectral image classification [J]. Optics and Precision Engineering, 2019, 27(3): 680-693.
- [4] Zhang X, Yang W, Tang X, et al. A fast learning method for accurate and robust lane detection using two-stage feature extraction with YOLO v3 [J]. Sensors, 2018, 18(12), 4308-4320.
- [5] Zhang J P, Zhang J M, Yu S. Hot anchors: a heuristic anchors sampling method in rcnn-based object detection [J]. Sensors, 2018, 18(10): 3415-3431.
- [6] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection [C] // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE Press, 2016: 779-788.
- [7] Liu W, Anguelov D, Erhan D, et al. Ssd: single shot multibox detector [C] // European Conference on Computer Vision. Amsterdam: Springer press, 2016: 21-37.
- [8] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation [C] // 2014 IEEE Conference on Computer Vision and Pattern Recognition. Ohio: IEEE Press, 2014: 580-587.
- [9] Girshick R. Fast R-CNN [C] // 2015 IEEE International Conference on Computer Vision (ICCV). Chile: IEEE Press, 2015: 1440-1448.
- [10] 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. Montreal; MIT press, 2015: 91-99.
- [11] Joseph Redmon, Ali Farhadi. YOLO9000: better, faster, stronger [C] // 2017 IEEE Computer Vision and Pattern Recognition (CVPR). Hawaii; IEEE Press, 2017: 6517-6525.
- [12] Joseph Redmon, Ali Farhadi. YOLOv3: an incremental improvement[J]. arXiv: Computer Vision and Pattern Recognition, 2018;1804,02767.
- [13] Bodla N, Singh B, Chellappa R, et al. Soft-NMS: improving object detection with one line of code[C] // 2017 IEEE International Conference on Computer Vision (ICCV). Venice; IEEE Press, 2017: 5561-5569.
- [14] Ding J, Chen B, Liu H, et al. Convolutional neural network with data augmentation for SAR target recognition[J]. IEEE Geoscience and Remote Sensing Letters, 2016, 13(3): 364-368.
- [15] Pan S J, Yang Q. A survey on transfer learning[J]. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(10): 1345-1359.
- [16] Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: common objects in context[M] // Computer Vision - ECVC 2014. Zurich; Springer International Publishing, 2014: 740-755.
- [17] Kingma D P, Ba J. Adam: a method for stochastic optimization [C] // International Conference for Learning Representations. California; arXiv preprint, 2014: 1412, 6980.

~~~~~

(上接第48页)

- [5] Xu Zhiyuan, Tang Jian, Meng Jingsong, et al. Experience-driven networking: a deep reinforcement learning based approach[C] // IEEE INFOCOM 2018 - IEEE Conference on Computer Communications. Honolulu, USA; IEEE Press, 2018: 1871-1879.
- [6] Sutton R S, Barto A G. Reinforcement learning: an introduction[M]. Cambridge; MIT Press, 1998.
- [7] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[EB/OL]. 2013(2013-12-19) [2019-07-05]. <https://arxiv.org/pdf/1312.5602.pdf>.
- [8] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[C] // Advances in Neural Information Processing Systems. Long Beach, USA; Curran Associates, 2017: 486-496.
- [9] Lowe R, Wu Y, Tamar A, et al. Multi-agent-actor-critic for mixed cooperative-competitive environments[C] // Advances in Neural Information Processing Systems. Long Beach, USA; Curran Associates, 2017: 6379-6390.
- [10] Winstein K, Balakrishnan H. Tcp ex machina: Computer-generated congestion control[C] // ACM SIGCOMM 2013. Hong Kong, China; ACM Press, 2013: 123-134.
- [11] 张峰, 李刚, 宋丽. 一种适应网络拥塞的网络端到端时延估算模型[J]. 空军雷达学院学报, 2009, 23(3): 190-193.
- Zhang Feng, Li Gang, Song Li. An estimation model of end-to-end delay of network congestion[J]. Journal of Air Force Radar Academy, 2009, 23(3): 190-193.