

# 基于改进级联卷积神经网络的交通标志识别\*

王海<sup>1</sup>, 王宽<sup>1</sup>, 蔡英凤<sup>2</sup>, 刘泽<sup>1</sup>, 陈龙<sup>2</sup>

(1. 江苏大学汽车与交通工程学院, 镇江 212000; 2. 江苏大学汽车工程研究院, 镇江 212000)

**[摘要]** 自动驾驶场景中交通标志的检测和识别十分重要, 为提高自然场景下交通标志检测精度, 本文中提出了一种基于 Cascade-RCNN 改进的交通标志识别算法。首先, 针对交通标志这类小目标特殊任务, 将 FPN 模块的深层特征信息融合进浅层特征层。其次, 改进了目标检测任务中的评价指标 IoU, 引入目标检测任务的直接评价指标 GIoU 指导定位任务, 提高了检测精度。最后, 算法在德国交通标志数据集 GTSDb 下进行了实验验证, 以 Res-Net101 为基础特征提取网络, mAP 可达 98.8%, 实验结果表明了所提算法的有效性, 具有优越的工程实用价值。

**关键词:** 交通标志检测; 深度学习; 卷积神经网络; 级联 RCNN

## Traffic Sign Recognition Based on Improved Cascade Convolution Neural Network

Wang Hai<sup>1</sup>, Wang Kuan<sup>1</sup>, Cai Yingfeng<sup>2</sup>, Liu Ze<sup>1</sup> & Chen Long<sup>2</sup>

1. School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212000;

2. Automotive Engineering Research Institute, Jiangsu University, Zhenjiang 212000

**[Abstract]** The detection and recognition of traffic signs in automatic driving scene is very important. In order to improve the accuracy of traffic sign detection in the natural scene, this paper proposes an improved traffic sign recognition algorithm based on Cascade-RCNN. Firstly, the deep feature information of FPN module is fused into the shallow feature layer for the special task of small targets such as traffic signs. Secondly, the evaluation index IoU of the target detection task is improved by introducing in the direct evaluation index GIoU of the target detection task to guide the positioning task, which improves the detection accuracy. Finally, the algorithm is verified by experiments in GTSDb, a German traffic sign data set. When the network extraction is based on ResNet101 features, the mAP can reach 98.8%. The experimental results show that the proposed algorithm is effective and has superior engineering practical value.

**Keywords:** traffic sign detection; deep learning; convolutional neural network; Cascade-RCNN

## 前言

交通标志的检测和识别在自动驾驶系统中十分重要, 在自动驾驶过程中, 迅速并正确识别出驾驶场景的交通标志对于交通安全意义重大。常用的交通标志检测主要有以下方法: 基于颜色<sup>[1]</sup>、形状<sup>[2]</sup>的

检测或者基于二者结合进行检测, 再用模板进行匹配的方法<sup>[3-4]</sup>; 传统基于机器学习的检测方法<sup>[5]</sup>。其中基于标志颜色、形状的检测通常采用对图像进行颜色分割的方法, 运算量小, 实时性好, 然而颜色特征受光照、天气的影响较大。基于标志形状的检测算法可以将交通标志从复杂背景中有效提取出来, 通常采用霍夫变换进行直线和圆的检测, 检测效

\* 国家重点研发计划(2018YFB0105000)、国家自然科学基金(51875255)、江苏省自然科学基金(BK20180100)、江苏省六大人才高峰项目(2018-TD-GDZB-022)和镇江市重点研发计划(GY2017006)资助。

原稿收到日期为 2019 年 11 月 16 日, 修改稿收到日期为 2020 年 2 月 3 日。

通信作者: 蔡英凤, 教授, 工学博士, E-mail: caicaixiao0304@126.com。

果良好,然而受环境影响也较大。传统的基于机器学习方法通常在分类中应用了如支持向量机<sup>[6]</sup>、贝叶斯分类器<sup>[7]</sup>等,然而该方法需要针对不同检测任务设计不同特征,识别效果一般且不稳定。

基于深度学习的计算机视觉算法作为计算机视觉领域的后起之秀,随着近几年数据量的不断提升和硬件水平的突飞猛进,在各类计算机视觉任务中都取得了巨大的成功,例如目标分类、目标检测和语义分割等。其中,针对目标检测,目前已有大量效果出众且实时性良好的算法。这些算法根据是否使用了区域候选网络(RPN)进行一次正样本的检测框回归,分为单阶段和二阶段检测算法。单阶段目标检测算法有YOLOv3<sup>[8]</sup>、SSD<sup>[9]</sup>、Retinanet<sup>[10]</sup>等,双阶段检测算法有RCNN<sup>[11]</sup>、RFCN<sup>[12]</sup>、Faster-RCNN<sup>[13]</sup>和Cascade-RCNN<sup>[14]</sup>等。单阶段目标检测算法相比二阶段检测算法,减少了一次RPN的回归操作,有着更快的检测速度,但同时因为少了一次目标框的回归,有大量的简单负样本信息干扰单阶段检测算法最后的检测框回归操作,导致检测结果有较大的定位误差。而以Faster-RCNN为代表的一系列二阶段检测算法相比单阶段算法一方面存在着检测速度的劣势,另一方面二阶段检测算法的高查全率也会导致检测结果中有较多的假阳性目标。针对检测速度这一问题,考虑到二阶段算法中RoI pooling的插入,打破了原卷积网络的平移不变性,牺牲了训练和测试效率,RFCN检测算法提出了位置敏感积分图(position sensitive score map)这一概念,积分图包含位置信息,因此能够把RoI pooling层的输入直接连接全连接层用于分类和回归,大大提高了训练和检测的速度。针对假阳性问题,研究人员提出的Cascade-RCNN算法通过级联几个检测网络达到不断优化预测结果的目的,在进一步提高位置回归精度的同时也使得检测结果假阳性得到了一定的抑制。在目标检测领域中,交通标志检测是一个十分重要的方向,研究交通标志检测算法对于交通安全十分重要。

基于以上的研究基础,本文中提出了一种改进的卷积神经网络,基于Cascade-RCNN深度学习框架对交通标志进行检测和识别,算法在德国交通标志数据集(GTSDB)<sup>[15]</sup>上进行了验证,获得了较好的检测效果。

本文中的贡献包括以下内容:(1)对一些目前主流的深度学习目标检测算法在GTSDB数据集上进行横向比较研究,包括检测算法的精确度和实时

性等,并对测试结果进行了分析和总结;(2)引入了特征图像金字塔FPN<sup>[16]</sup>,针对交通标志这类小目标特殊任务,为更好地识别大小不同的物体,通过多尺度融合,将低层图像的位置信息与高层图像的特征语义信息融合,更好地表达图像信息;(3)通过引入文献[17]中提出的GIoU(generalized intersection over union)这一优化目标加快模型拟合预测框与真实框的位置关系,并提高交通标志识别回归框的准确率;(4)通过相关实验结果分析,本文所设计的目标检测模型在检测性能上相比其他目标检测模型有明显的提升。

## 1 基于 Cascade-RCNN 改进的交通标志检测算法框架

### 1.1 Cascade-RCNN 算法框架

Cascade-RCNN 是 Faster-RCNN 的改进版,Faster-RCNN 算法包含两个卷积神经网络,其结构如图1所示。Faster-RCNN 由区域生成网络(RPN)和 Fast-RCNN 两部分构成,RPN 用于生成目标可能存在的候选区域;Fast-RCNN 实现对候选区域的分类,并进行边界回归。这两部分共享一个提取特征的卷积神经网络,使得目标候选区域的检测时间几乎为0,大幅提高了目标检测的速度。

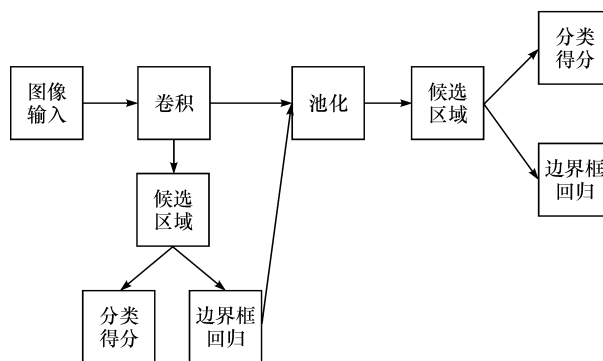


图1 Faster-RCNN 结构示意图

在目标检测算法中,IoU 阈值常用于定义并生成正负样本,当使用低 IoU 阈值训练检测器时,会带来检测噪声(检测精度不高);如果仅仅提高 IoU 阈值,检测性能也同样下降,原因如下:(1)IoU 阈值的提高,用于训练的正样本数量呈指数倍减少,会导致训练的过拟合;(2)预测阶段,输入与检测分支训练时采用的 IoU 阈值不一致,会导致性能下降。

双阶段检测网络会对提取的候选区域进行分类和回归。在分类过程中,每个候选区域中的正样本

和负样本会根据初始设定的 IoU 阈值进行分类。在训练回归任务中,网络不断地向正样本的边界框回归。因此正确的 IoU 设定对于网络的训练和验证具有重大影响。

于是 Cai 等<sup>[14]</sup>在 Fast-RCNN 和 Faster-RCNN 的基础上提出改进的目标检测算法 Cascade-RCNN。当设置一个阈值训练时,训练出的检测器效果有限,在级联 RCNN 中就采用了多阶段结构,每个阶段分别设置不同的 IoU 阈值。算法的主要思想是设置不同阈值,阈值越高,其网络对准确度较高的候选框的作用效果越好。通过级联的 RCNN 网络,每个级联的 RCNN 设置不同的 IoU 阈值,这样每个网络输出的准确度提升一点,用作下一个更高精度的网络的输入,逐步将网络输出的准确度进一步提高。对于交通标志检测识别,使用级联 RCNN 可使检测更加准确。

在交通标志识别中,为解决交通标志检测精度不高的问题,本文中采用级联 RCNN 网络结构(图 2)。与图 1 所示的 Faster-RCNN 相比,将多个不同的检测器根据 IoU 的值串联起来。本文中设定 IoU 分别是 0.3、0.4 和 0.5 的 3 个级联检测器,每个检测器的输入源自上一层检测器边界框回归后的结果。这种设定的网络可使正样本的图像信息得到充分利用。首先初始化搭建的卷积神经网络模型参数,其中卷积层参数由利用 ImageNet 图像数据集预先训练的 ResNet101<sup>[18]</sup>得到,其他层的参数都是采用随机方式初始化;为获得图像的多尺度特征表达,利用卷积神经网络提取层次特征,进而建立特征金字塔;然后根据 RPN 网络生成目标候选区域,在特征金字塔各个层级中找到对应的目标区域,从而提取出目标的多尺度特征;最后将池化后的特征进行特征融合。

本文中采用的级联网络结构前传步骤详细如

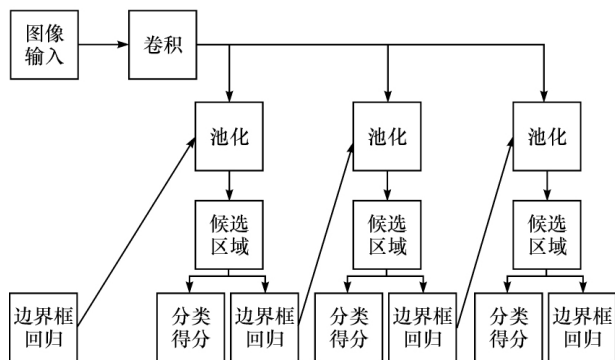


图 2 级联 RCNN 网络结构

下: 图片送入模型中,经过 RPN 网络,RPN 网络从全部 anchor 中分配正样本和负样本,大于阈值 0.7 的为正样本,小于阈值 0.3 的为负样本,从全部的正样本和负样本中随机抽取 200 个样本进行 loss 训练,200 个样本中正样本比例占 0.5,接着 RPN 后处理。按照预测的分类概率进行排序,利用非极大值抑制筛选后得到 2 000 个候选区域,然后将 2 000 个候选区域选出 512 个送入第 1 个 RCNN 网络中,正样本比例为 0.25,IoU 大于 0.3 的为正样本,IoU 小于 0.3 为负样本,接着再将上一步中的输出送入第 2 个 RCNN 网络中,正样本比例为 0.25,IoU 大于 0.4 的为正样本,小于 0.4 的为负样本,最后再将上一步中的输出送入第 3 个 RCNN 网络中,正样本比例为 0.25,IoU 大于 0.5 的为正样本,小于 0.5 的为负样本。这种设定在一定程度上减少了正样本的浪费和流失,使得检测网络适应各种目标的检测情况,提高网络的整体检测精度。

## 1.2 特征图像金字塔

通过本文中采用的级联 RCNN 网络结构,可提高网络整体的检测精度,但是针对交通标志这类小目标特殊任务,检测效果提升不明显,通常为了更好地识别大小不同的物体,特征图像金字塔<sup>[19]</sup>是一个常见方法。常见的特征图像金字塔将图像化成不同大小尺寸,对每一个尺寸的图像生成各自的特征并进行预测。虽然可保证特征提取的完备性,但是时间成本太大。通过该方式,网络速度快,内存占用小。但是由于忽略了低层网络的细节信息,网络精度有待提高。

基于上述原因,本文中创新性地引用 FPN (feature pyramid network) 的思想,在图像融合前,分别对两者生成特征图,由于图像底层的特征语义信息比较少,不能直接分类,但目标信息准确;高层特征语义信息比较丰富,但目标位置不准确,本文中采用如图 3 所示的特征金字塔网络 FPN,其网络结构为自底向上和自顶向下的特征映射的构建模块。将自底向上和自顶向下的路径进行横向信息融合,通过对高层特征上采样和底层信息进行 3×3 卷积操作融合,可充分还原最低层特征图的语义信息,并得到有丰富的特征信息和高分辨率的特征图,对小目标有很好的检测率,且生成的特征图满足后续的处理。

首先,通过设定空间分辨率采样为 2 像素来完成较为粗糙的分辨率特征映射。其次,通过逐个元素叠加的方法将上采样的映射与对应的自下而上的

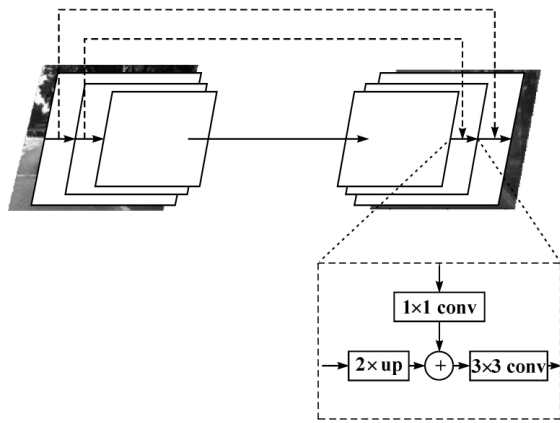


图3 FPN网络结构示意图

映射(其经历  $1 \times 1$  卷积层以减少信道维度)合并。迭代此过程,直到生成最佳分辨率的映射,然后添加一个  $1 \times 1$  卷积层来生成最粗糙分辨率映射以开始迭代。最后,由于要减少上采样的混叠效应,在每个合并的映射上添加一个  $3 \times 3$  卷积来生成最终的特征映射,即特征图。

整个算法大致分为3个部分:自底向上、横向连接和自顶向下。自底向上就是普通的卷积神经网络前馈过程,如果同一阶段的不同层产生相同大小的输出映射,则选择每个阶段最后一层的输出作为待处理特征图;自顶向下过程就是将自底向上过程得到的特征图进行上采样,该目的是使上采样后的特征图和相邻低层的特征图大小相同,可进行特征图相加;横向连接就是将上采样的结果和自下而上生成的特征图进行融合。这种做法只是在基础网络上增加了跨层连接,优点是会增加网络额外的时间和计算量。

### 1.3 针对目标定位损失函数的优化

本文将 GIoU 作为一种新的评价指标损失函数替代原有评价指标 IoU 引入到了级联 RCNN 中,由 Loss-GIoU 计算公式可知, Loss-GIoU 能保证损失函数目标检测框回归具有尺度不变性,同时使检测框的优化目标和损失函数保持一致。在交通标志识别中,由于正样本和负样本之间存在显著的不平衡,这使得训练边界框得分更加困难,使用 GIoU 可有进一步改进的机会。在很多目标检测算法中,定位损失函数都使用预测框与真实框角点坐标距离的 1-范数 Loss 或者 2-范数 Loss。目标检测领域 IoU 是对象检测基准中最常使用的评估指标,在最终评价模型性能时使用预测框与真实框的交并比来判断是否进行了正确的检测。IoU 作为定位回归的损失函

数主要有以下问题:(1)对于不重叠的两个对象, IoU 值将为 0,并不能反映两个形状之间的距离,在非重叠对象的情况下,如果使用 IoU 直接作为损失,则梯度为 0 无法进行优化;(2) IoU 对于不同的对齐方式也无法进行区分。

如图4所示,图中右上角在圆心的矩形框为真实框,右上角在圆上的矩形框为预测框。可以看出预测框与真实框的 2-范数 Loss 都是相等的,但是第3个的 IoU 明显高于前两个。在 1-范数与 2-范数取到相同值时,实际上检测效果却差异巨大,直接表现就是预测和真实检测框的 IoU 值变化较大,这说明 1-范数和 2-范数不能很好地反映检测效果,模型对这两者的优化不是等价的,所以检测结果也优于前两者。更精确地说,对于两个具有相同交点水平的不同方向重叠对象, IoU 将完全相等。因此, IoU 函数值并不能反映两个对象是如何重叠的。所以基于此问题,本文中提出了一种新的度量预测框与真实框拟合程度的方法,将 GIoU 作为一种损失函数引入到 2D 目标检测中。

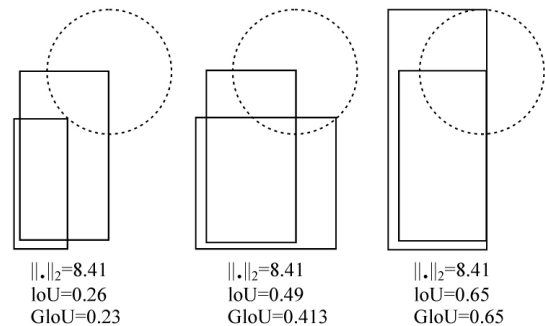


图4 相同 2-范数 Loss 不同 IoU

IoU 和 GIoU 的计算公式为

$$IoU = \frac{A \cap B}{A \cup B} \quad (1)$$

$$GIoU = IoU - \frac{|C - (A \cup B)|}{|C|} \quad (2)$$

式中:  $A$  为预测框;  $B$  为真实框;  $C$  为两者的最小闭包(包含  $A$ 、 $B$  的最小包围框)

GIoU 的详细计算方法如下所示。

输入:

预测框坐标信息  $Box^p = (x_1^p, y_1^p, x_2^p, y_2^p)$

真实框坐标信息  $Box^g = (x_1^g, y_1^g, x_2^g, y_2^g)$

输出:  $Loss_{IoU}$  和  $Loss_{GIoU}$

(1) 计算  $x_1^p = \min(x_1^p, x_2^p)$ ,  $x_2^p = \max(x_1^p, x_2^p)$ ,  $y_1^p = \min(y_1^p, y_2^p)$ ,  $y_2^p = \max(y_1^p, y_2^p)$

(2) 计算真实框的面积:  $Area^g = (x_2^g - x_1^g) \times (y_2^g - y_1^g)$ ; 预测框的面积:  $Area^p = (x_2^p - x_1^p) \times (y_2^p - y_1^p)$

(3) 计算预测框和标签框的交集:  $x_1^I = \max(x_1^p, x_1^g)$ ,  $x_2^I = \min(x_2^p, x_2^g)$ ,  $y_1^I = \max(y_1^p, y_1^g)$ ,  $y_2^I = \min(y_2^p, y_2^g)$

(4) 当  $x_2^I > x_1^I$   $y_2^I > y_1^I$  时 交集  $I = (x_2^I - x_1^I) \times (y_2^I - y_1^I)$

(5) 最小封闭图形 C 的坐标记做  $Box^C$ :  $x_1^C = \min(x_1^p, x_1^g)$ ,  $x_2^C = \max(x_2^p, x_2^g)$ ,  $y_1^C = \min(y_1^p, y_1^g)$ ,  $y_2^C = \max(y_2^p, y_2^g)$

(6)  $IoU = I/U$ , 其中  $U = Area^p + Area^g - I$ ,  $GIoU = IoU - (Area^C - U) / Area^C$

综上损失函数为

$$Loss_{IoU} = 1 - IoU, Loss_{GIoU} = 1 - GIoU$$

## 2 实验与结果

### 2.1 数据准备

本文中使用了世界知名的公开交通标志数据集—德国交通标志检测数据集 GTSDB, 目标图像大小为  $1\,360 \times 800$ , 本文中所采用的数据集包括不同的交通标志图像共 900 张, 其中训练数据为 600 张, 测试数据为 300 张。其中, 图像中交通标志的数量在 0 到 6 个不等。每张图片均为 RGB 三通道的彩色图像。图像分为 4 类: 禁止 (prohibitory)、警告 (danger)、指示 (mandatory) 与其它 (other)。交通标志示例如图 5 所示。实验平台为 Intel 酷睿 i7 8700K 处理器, 64GB 内存, 操作系统为 Ubuntu 18.04, 编程软件为 Pycharm, 使用的运算显卡为 GTX1080Ti。

### 2.2 实验步骤和实验细节

本文中选择了 4 种目前主流的深度学习目标检测算法以及本文中改进的级联 RCNN 算法共 5 种算法进行横向比较研究, 其中包含了两类单阶段检测模型 YOLOv3 和 RetinaNet, 两类二阶段检测模型 Faster-RCNN 和原版 Cascade-RCNN, 以及本文中改进的算法。训练和测试平台是一台搭载了 GTX1080-Ti GPU 的工作站, 在 GTSDB 数据集上评估这 4 种目标检测模型。对比实验统一将输入检测模型的图片尺寸定为  $1\,360 \times 800$ , 考虑到显卡内存的限制 batchsize 选择为 4, 训练时选择的优化器都是 SGD 算法<sup>[20]</sup>, 初始学习率设置为 0.001。除了



图 5 交通标志示例

YOLOv3 使用 Darknet-53 作为基础特征提取之外, 其余 4 类目标检测模型, 本文中均使用在 Imagenet 上进行预训练的 ResNet101 网络作为基础特征提取网络。

### 2.3 实验结果

本文中算法采用的评价指标为平均精度均值 (mean average precision), 实际上就是 4 类交通标志平均精度 (AP) 的平均值。每一个类别检测结果指标为精确率 (Precision), 即被正确检测出来的交通标志占全部检出区域的百分比, 其计算方式如式 (3) 所示; 召回率 (Recall), 即被正确检测出来的交通标志占全部交通标志的百分比, 其计算方式如式 (4) 所示。根据每一个类别检测结果的精确率和召回率绘制一条曲线, 曲线下的面积就是平均精度, 其计算如式 (5) 所示。5 种算法的实验结果如表 1 所示。

$$PR = \frac{TP}{TP + FP} \quad (3)$$

$$RE = \frac{TP}{TP + FN} \quad (4)$$

$$AP = \int_0^1 p(r) dr \quad (5)$$

式中:  $PR$  为精确率;  $RE$  为召回率;  $AP$  为平均精度;  $TP$  为正确检测出来的交通标志数量;  $FP$  为背景被检测为交通标志的数量;  $FN$  为交通标志被检测为背景的数量;  $p$  为纵轴精确率;  $r$  为横轴召回率。

表 1 不同算法的检测精度对比

方法	基础特征提取网络	AP / %				平均准确率 ( mAP) / %	每秒帧数
		AP( 禁止)	AP( 警告)	AP( 指示)	AP( 其他)		
YOLOv3	DarkNet53	92. 2	94. 6	93. 2	94. 0	93. 5	10. 17
RetinaNet	ResNet101	95. 4	95. 2	93. 8	94. 4	94. 7	3. 69
Faster-RCNN	ResNet101	96. 2	95. 4	94. 7	95. 6	95. 5	1. 84
原版 Cascade-RCNN	ResNet101	97. 4	96. 8	95. 7	97. 2	96. 8	2. 58
本文中算法	ResNet101	98. 4	99. 0	99. 2	98. 8	98. 8	2. 74

## 2.4 结果分析

表 1 清晰地显示了不同目标检测算法对于不同交通标志类别目标的检测精度,同时也反映了各算法对检测样本的查全和查准的能力。

两类单阶段模型 YOLOv3、Retinanet 在检测 GTSDb 的检测精度较低,相比于双阶段检测算法 Faster-RCNN、Cascade-RCNN 以及本文中的算法 mAP 较低,检测较不精确。

相比于改进前的 Cascade-RCNN,当使用本文中改进的方法后,其中禁止类标志牌 AP 提高了 1 个百分点,警告类标志牌 AP 提高了 2.2 个百分点,指示类标志牌 AP 提高了 3.5 个百分点,其他类标志牌 AP 提高了 1.6 个百分点,平均准确率 mAP 提高了 2 个百分点。

实验中,本文中算法得到的部分检测结果如图 6 所示,所提出的方法可以很好地检测不同尺寸、角



图 6 本文算法检测效果

度和位置的交通标志。在图 6 中,包围框区域代表交通标志牌的位置,数字表示置信度,由图可以看出,对于较小的交通标志,本文中使用的模型能够很好地将交通标志检测出来,较好地解决了对交通标志这样的小目标漏检的问题。

改进后算法与 YOLOv3 算法检测效果对比如图 7 所示,可知改进后算法的车牌检测效果相较于 YOLOv3 的算法定位更加准确。由图 8 可知,原版 Cascade-RCNN 算法中较小的交通标志被漏检,改进后的检测效果较好,改进后的检测效果准确率和召回率都更高。同时本文中改进后的 Cascade-RCNN



(a) YOLOv3检测效果



(b) 本文算法检测效果

图 7 本文算法与 YOLOv3 算法检测效果对比



(a) 原版Cascade-RCNN检测效果



(b) 本文算法检测效果

图8 本文算法与原版 Cascade-RCNN 检测效果对比

算法相比于原版算法在检测速度上几乎一样,改进后不产生明显的时间代价。

通过上述实验结果表明,测试的总体识别率达到 98.8%,对于不同光照下和不同场景以及不同距离都有着良好的检测效果,本文中提出的算法准确率高于其他算法,对于现实场景下的交通标志有着良好的检测效果,有着很高的准确性。

### 3 结论

本文中针对交通标志的识别采用了基于级联 Cascade-RCNN 改进优化方法,在原有网络基础上引入 FPN 模块,将深层特征信息融合进浅层特征层,同时引入了更加接近检测任务的评价标准 GIoU-Loss 作为代价函数以提高模型的定位能力。该方法能识别出交通标志,既可提高网络在各个候选区域的检测精度,又可重点解决小目标检测问题,使网络的整体检测精度有了较大的提升。经过多次合理的模型参数优化后,检测准确率达 98.8%。但由于基础特征网络 ResNet101 的层数较深,单帧图像的检测时间较长,实时性还有待提高,下一步可考虑减少模型体积压缩模型来完成交通标志的实时检测与识别。

### 参考文献

- [1] LIM K H, SENG K P, ANG L, et al. Intra color-shape classification for traffic sign recognition[C]. International Computer Symposium, 2010: 642-647.
- [2] BARNES N, ZELINSKY A, FLETCHER L, et al. Real-time speed sign detection using the radial symmetry detector[J]. IEEE Transactions on Intelligent Transportation Systems, 2008, 9(2): 322-332.
- [3] LA ESCALERA A D, MORENO L, SALICHS M A, et al. Road traffic sign detection and classification[J]. IEEE Transactions on Industrial Electronics, 1997, 44(6): 848-859.
- [4] FANG C Y, CHEN S W, FUH C S. Road-sign detection and stochastic gradient descent tracking[J]. IEEE Transactions on Vehicular Technology, 2003, 52(5): 1329-1341.
- [5] QIAN R, ZHANG B, YUE Y, et al. Robust chinese traffic sign detection and recognition with deep convolutional neural network[C]. International Conference on Natural Computation, 2015: 791-796.
- [6] GREENHALGH J, MIRMEHDI M. Real-time detection and recognition of road traffic signs[J]. IEEE Transactions on Intelligent Transportation Systems, 2012, 13(4): 1498-1506.
- [7] MEUTER M, NUNN C, GORMER S M, et al. A decision fusion and reasoning module for a traffic sign recognition system[J]. IEEE Transactions on Intelligent Transportation Systems, 2011, 12(4): 1126-1134.
- [8] REDMON J, FARHADI A. YOLOv3: an incremental improvement[J]. arXiv: Computer Vision and Pattern Recognition, 2018.
- [9] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot multibox detector[C]. European Conference on Computer Vision. Springer International Publishing, 2016: 21-37.
- [10] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 99: 2999-3007.
- [11] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]. Computer Vision and Pattern Recognition, 2014: 580-587.
- [12] DAI J, LI Y, HE K, et al. R-FCN: object detection via region-based fully convolutional networks[C]. Neural Information Processing Systems, 2016: 379-387.
- [13] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]. Neural Information Processing Systems, 2015: 91-99.
- [14] CAI Z, VASCONCELOS N. Cascade R-CNN: delving into high quality object detection[C]. Computer Vision and Pattern Recognition, 2018: 6154-6162.
- [15] HOUBEN S, STALLKAMP J, SALMEN J, et al. Detection of traffic signs in real-world images: the German traffic sign detection benchmark[C]. International Joint Conference on Neural Network, 2013: 1-8.

(下转第 1269 页)

度,实现无人驾驶车辆从高速公路收费窗口到匝道之间的安全行驶,该方法对于实现高速全程无人驾驶具有重要意义。利用6阶多项式优化从收费窗口到匝道之间参考路径,对于更复杂的道路结构,可以通过采集路点使用连接低阶贝塞尔曲线的方法获得参考路径。

针对决策树模型,设置驾驶速度范围在0到60 km/h之间,无人驾驶车辆与周围车辆的驾驶行为完全随机,执行超过100次仿真。汽车行驶仿真结合了汽车运动学模型,并采用纯追踪法调整转向角。仿真结果表明,本文中所提出的方法在周围车辆位置、速度任意时刻都已知的条件下,能够实现无人驾驶车辆从高速公路收费窗口到匝道之间的安全驾驶。

### 参考文献

- [1] MA L, XUE J, KAWABATA K, et al. Efficient sampling-based motion planning for on-road autonomous driving[J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16(4): 1961–1976.
- [2] HUANG X, CAO Q, ZHU X. Mixed path planning for multi-robots in structured hospital environment[J]. The Journal of Engineering, 2019(14): 512–516.
- [3] YANG Q, YOO S. Optimal UAV path planning: sensing data acquisition over IoT sensor networks using multi-objective bio-inspired algorithms[J]. IEEE Access, 2018, 6: 13671–13684.
- [4] LIU Y, WANG X, LI L, et al. A novel lane change decision-making model of autonomous vehicle based on support vector machine[J]. IEEE Access, 2019, 7: 26543–26550.
- [5] BEVLY D, CAO X, GORDON M, et al. Lane change and merge maneuvers for connected and automated vehicles: a survey[J]. IEEE Transactions on Intelligent Vehicles, 2016, 1(1): 105–120.
- [6] DO Q, TEHRANI H, MITA S, et al. Human drivers based active-passive model for automated lane change[J]. IEEE Intelligent Transportation Systems Magazine, 2017, 9(1): 42–56.
- [7] GURUJI A, AGARWAL H, PARSEDIYA D. Time-efficient A\* algorithm for robot path planning[J]. Procedia Technology, 2016, 23: 144–149.
- [8] DAKULOVIC M, PETROVIC I. Two-way D\* algorithm for path planning and replanning[J]. Robotics and Autonomous Systems, 2011, 59(5): 329–342.
- [9] LIKHACHEV M, FERGUSON D, GORDON G, et al. Anytime dynamic A\*: an anytime, replanning algorithm[C]. 15th International Conference on Automated Planning and Scheduling, 2005: 262–271.
- [10] OLIVEIRA R, CIRILLO M, MARTENSSON J, et al. Combining lattice-based planning and path optimization in autonomous heavy duty vehicle applications[C]. IEEE Intelligent Vehicles Symposium, 2018: 2090–2097.
- [11] KHAILI M E. Visibility graph for path planning in the presence of moving obstacles[J]. IRACST-Engineering Science and Technology: An International Journal (ESTIJ), 2014, 4(4): 118–123.
- [12] WANG J, CHEN Z. A novel hybrid map based global path planning method[C]. 3rd Asia-Pacific Conference on Intelligent Robot Systems, 2018: 66–70.
- [13] MIURA J. Support vector path planning[C]. International Conference on Intelligent Robots and Systems, 2006: 3435–3440.
- [14] RASEKHIPOUR Y, KHAJEPOUR A, CHEN S K, et al. A potential field-based model predictive path-planning controller for autonomous road vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 18(5): 1255–1267.
- [15] MORALES N, TOLEDO J, ACOSTA L. Path planning using a multiclass support vector machine[J]. Applied Soft Computing, 2016, 43: 498–509.
- [16] 3-second rule for safe following distance[EB/OL]. <https://www.travelers.com/resources/auto/travel/3-second-rule-for-safe-following-distance.aspx>.
- [16] LIN T, DOLLAR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C]. Computer Vision and Pattern Recognition, 2017: 936–944.
- [17] REZATOFIGHI H, TSOI N, GWAK J, et al. Generalized intersection over union: a metric and a loss for bounding box regression[C]. Computer Vision and Pattern Recognition, 2019: 658–666.
- [18] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]. Computer Vision and Pattern Recognition, 2016: 770–778.
- [19] ADELSON E H. Pyramid methods in image processing[J]. Rca Engineer, 1984: 29.
- [20] PARAS. Stochastic gradient descent[J]. Optimization, 2014, doi: 10.1007/978-1-4842-2766-4\_8.

(上接第1262页)