

基于 Dense-YOLOv3 的车型检测模型^①



陈立潮¹, 王彦苏¹, 曹建芳²

¹(太原科技大学 计算机科学与技术学院, 太原 030024)

²(忻州师范学院 计算机系, 忻州 034000)

通讯作者: 王彦苏, E-mail: 948084595@qq.com

摘 要: 针对传统 YOLOv3 的网络结构存在曝光过度或光线较暗等异常图片在提取特征时鲁棒性较差, 导致车型识别率低下的问题, 提出了一种用于交通车辆检测的 Dense-YOLOv3 模型. 该模型集成了密集卷积神经网络 DenseNet 和 YOLOv3 网络的特点, 加强了卷积层之间的车型特征传播和重复利用, 提高了网络的抗过拟合性能; 同时, 对目标车辆进行了不同尺度的检测, 构建了交叉损失函数, 实现了车型的多目标检测. 经过在 BIT-Vehicle 标准数据集上对模型进行训练和测试, 实验结果表明, 基于 Dense-YOLOv3 车型检测模型平均精度达到了 96.57%, 召回率为 93.30%, 表明了该模型对车辆检测的有效性和实用性.

关键词: 车型检测; YOLOv3; DenseNet; 鲁棒性; 平均精度; 召回率

引用格式: 陈立潮, 王彦苏, 曹建芳. 基于 Dense-YOLOv3 的车型检测模型. 计算机系统应用, 2020, 29(10): 158-166. <http://www.c-s-a.org.cn/1003-3254/7634.html>

Vehicle Recognition Algorithm Based on Dense-YOLOv3

CHEN Li-Chao¹, WANG Yan-Su¹, CAO Jian-Fang²

¹(School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030024, China)

²(Department of Computer, Xinzhou Teachers University, Xinzhou 034000, China)

Abstract: The traditional YOLOv3 network structure has poor robustness in extracting features such as over exposure or dark light, which leads to low recognition rate. A Dense-YOLOv3 model for traffic vehicle classification is proposed. The model integrates the characteristics of dense convolutional neural network DenseNet and YOLOv3 network, which strengthen the vehicle model feature propagation and reuse between convolution layers, and improve the anti-overfitting performance of the network. At the same time, the target vehicle is detected at different scales, and the cross-loss function is constructed to realize the multi-objective detection of the vehicle model. The model is trained and tested on BIT-Vehicle standard data sets. The experimental results show that the average accuracy of the model based on Dense-YOLOv3 vehicle detection reaches 96.57% and the recall rate is 93.30%, which indicates the effectiveness and practicability of the model for vehicle detection.

Key words: vehicle detection; YOLOv3; DenseNet; robustness; mean average precision; recall

目前, 交通强国作为国家的发展战略, 快速发展信息化、智能化的交通对一个国家至关重要, 因此, 智慧交通^[1] 市场存在非常大的潜力. 现阶段, 智慧交通利用

高新 IT 技术^[2] 如: 大数据、人工智能、云计算、物联网等, 将其融入到智能交通中, 向交通有关部门实时提供当地交通服务信息, 达到人、车、路和谐统一, 对改

① 基金项目: 山西省自然科学基金 (201801D221179, 201701D121059)

Foundation item: Applied Basic Research Project of Shanxi Province (201801D221179, 201701D121059)

收稿时间: 2020-03-03; 修改时间: 2020-03-27; 采用时间: 2020-04-17; csa 在线出版时间: 2020-09-30

善运输环境、保障交通安全和提高交通运输效率具有重要意义。其中,车辆检测作为智慧交通关键技术,具有重要的理论研究价值。

迄今为止,卷积神经网络^[3]掀起了两大检测技术潮流:一是以 R-CNN 为首的两刀流算法,先生成可能区域^[4]即采用 CNN 提取特征,再将提取到的特征放入分类器进行分类;二是以 YOLO 为首的一刀流算法,直接对图片或视频中的目标物体进行回归。Girshick 等^[5]提出的启发式方法 R-CNN,即通过生成候选区域来检测目标物体,并且降低了信息冗余度;He 等^[6]提出的 SPP-NET 利用先卷积再生成候选区域,减少 R-CNN 的存储量,一定程度上加快训练速度;Girshick^[7]提出的将 R-CNN 的串行结构改成并行结构的 Fast R-CNN、Ren 等^[8]又提出 Faster R-CNN 使得网络学习到深层、抽象、语义特征,提高了生成的候选区域可靠程度、He 团队^[9]又对 Faster R-CNN 做出改进即加入了一个图像的 Mask 信息输入的 Mask R-CNN 算法。以上方法在大量生成无效区域的情况下会产生无关的算力、少量的生成区域则会发生漏检,并且在跨平台的交互一定会损失算法效率。而 YOLO 系列的算法核心思想是利用网络结构中优秀分类效果的分类器,将图片中的目标物体先给出一个大致的范围进行检测,然后再不断迭代该范围直到一个精准的位置。其中典型的算法有 Redmon 等^[10]提出的 YOLO;Liu 等^[11]使用的 SSD;再由 Redmon 团队^[12,13]提出的 YOLO9000、YOLOv3 算法等。

综上所述,YOLO 系列算法和 R-CNN 系列算法分别各个不同领域方向具有自己独特的优势。为了提高车辆检测效果,本文采用 R-CNN 系列中的 DenseNet 网络与 YOLO 系列的 YOLOv3 集成的方式,综合利用 DenseNet 网络中稠密连接的优点,有效解决了传统 YOLOv3 的网络结构无法适应特殊车型数据集导致准确率低的现象,而且还满足实时性的要求。

1 Dense-YOLOv3 车辆检测模型

1.1 YOLOv3 模型

YOLOv3 模型是以框架 Darknet-53 作为车型特征提取容器与多尺度特征融合检测支路两部分共同构成网络结构,Darknet-53 结构如表 1 所示。YOLOv3 的核心思想:首先通过网络层的特征提取器对输入图片进行特征的提取,提取到不同大小的特征图;然后在网络

中将图像划分为 13×13 的网格;其次通过 Ground Truth 确定目标物体的中心坐标位于网格中的哪个单元格,利用该单元格对目标物体进行预测,网格中的每个单元格预测 3 个边界框;最后将这 3 个边界框与 Ground Truth 通过交并比即 IOU 技术^[14]选出最大边界框来检测出目标物体,输出目标物体的所在图片中的坐标信息以及置信度^[15]。

表 1 DarkNet-53 网络结构

	Type	Filters	Size	Output
	Convolution	32	3×3	256×256
	Convolution	64	3×3/2	128×128
1×	Convolution	32	1×1	
	Convolution	64	3×3	
	Residual			128×128
	Convolution	128	3×3/2	64×64
2×	Convolution	64	1×1	
	Convolution	128	3×3	
	Residual			64×64
	Convolution	256	3×3/2	32×32
4×	Convolution	128	1×1	
	Convolution	256	3×3	
	Residual			32×32
	Convolution	512	3×3/2	16×16
8×	Convolution	256	1×1	
	Convolution	512	3×3	
	Residual			16×16
	Convolution	1024	3×3/2	8×8
16×	Convolution	512	1×1	
	Convolution	1024	3×3	
	Residual			8×8

1.2 Dense-YOLOv3 网络

从理论上讲,加深网络结构模型会取得很好的识别结果。然而,大量实验结果表明,深层的网络结构存在退化问题,He 等^[16]提出的残差网络 ResNet 来解决了深层网络退化问题。YOLOv3 模型中加入了残差模块,使得车型特征在 YOLOv3 深层网络中得到很好的训练,网络的性能也提升了。但是,Darknet-53 中的 5 个具有不同尺度的、深度的残差层,各个残差层之间只进行求残差操作,即只是将该层与前边某两层进行相加连接,从信息流通角度来看,当前层并未彻底收到前层提取到的车型特征信息,很容易漏掉一些关键的车型信息。因此,为了使提取到的车型信息更加完善,本文算法将特征提取结构 DenseNet 网络与传统的 YOLOv3 模型相融合,综合利用了密集网络中每一层都会接收到前边所有层的特征信息,将其作为该层的额外输入,此连接机制更加激进。文中密集卷积神经网络将提取

到的车型特征信息进行重复利用以及其浅层网络复杂度低的优点,对存在异常情况下的车辆有很好的检测效果。

1.2.1 DenseNet 模块

Huang 等团队^[17]通过引用残差网络 (ResNet) 思想,但不同于 ResNet 网络的跳跃连接,而是将在网络

层一端输出的特征图和另一端得到的特征图拼接在一起进行学习,从而形成更多的特征图,相当于将前面层提取到的特征又在后边的网络进行重复利用。在网络中,通过利用这种拼接方式连在一块的多个层被称为密集卷积块 (Dense Block),并且在每个密集卷积块之间的层叫做转移层,如图 1 所示。

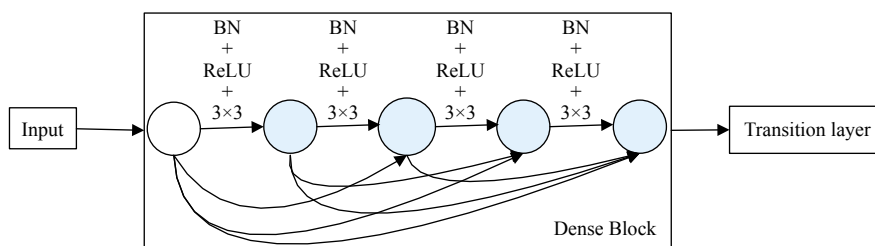


图 1 Dense Block 图

DenseNet 网络中,为了减少特征图的数量,在 Dense Block 中的 3×3 卷积核前加入了 1×1 的卷积核,不仅可以融合各个不同通道的特征,而且其降维的方式减少了计算量。此外,密集卷积块之间又通过 1×1 的卷积核进行连接,又压缩了参数量。在密集网络中,第 L 层的输出为:

$$X_l = H_l([X_0, X_1, \dots, X_{l-1}]) \quad (1)$$

其中, H_l 表示非线性变换,随着层数增加,就会将大量线性函数进行复合,相比于其他神经网络,此处的非线性变换使得 DenseNet 网络有很好的泛化性能, $[X_0, X_1, \dots, X_{l-1}]$ 表示第 0 到 $l-1$ 层输出的特征做通道合并。通过采用密集卷积块,对于网络层数增加不再是问题,有效解决了梯度消失,而且网络训练效果非常好。

1.2.2 Dense-YOLOv3 网络

Dense-YOLOv3 网络结构模型如图 2 所示,该模型利用 DenseNet 网络与 YOLOv3 多尺度预测两者相融合,密集连接网络 DenseNet 比其他网络效率更高,其关键在于网络中将后续层与其前边任何层进行直接连接,改善了层与层之间的信息交流,使得浅层网络提取到的特征进行了重复利用。

将标注好的车型数据集按照具有 RGB 三通道图像输入到 Dense-YOLOv3 模型的密集网络框架中进行特征的提取。首先,进入设定步长为 2 的 7×7 卷积核,紧接着采用 3×3 进行最大池化;接着,依次进入数量分别为 6、12、24、16 个图 1 中的 Dense Block 集,其中,将 3 个不同层次的特征图送到 y_1 、 y_2 和 y_3 通道,

如图 2 所示提取到的车型特征图,另外,密集卷积块之间通过转移层连接,该层使用卷积核大小为 1×1 、步长为 2 的 2×2 进行均值池化,到此阶段, Dense-YOLOv3 模型的车型特征的提取阶段已经完成;然后,利用神经网络每一层可以提取到高、中、低各个层次的车型特征信息,并且将不同层次特征信息进行组合的原理,在 Dense-YOLOv3 模型中分别采用 13×13 、 26×26 、 52×52 三个不同的尺度在不同网络深度通过卷积核操作再进行局部特征信息的交互:在最大尺度为 52×52 层中,先将通道为 1024 个经过一系列卷积操作减小为 $3 \times (4+1+\text{类别数})$,即 33 个,在这个基础上进行回归,同理,再对中尺度 26×26 、最小尺度 13×13 分别也进行通道缩减,都减小为 33 个,最后,将最小尺度层与中尺度层一起进行 2 倍的上采样、特征合并,实现最后目标物体车的检测。

在 Dense-YOLOv3 模型训练中,考虑到数据集中车型 Bus 与 Truck 在轮廓、颜色和纹理等基础结构上具有一定的相似性,并且网络开始阶段往往是对这些基础结构进行信息的提取。因此,在训练时利用高层独特的泛化性能及易训练优点,可以采用冻结低层网络方式提高测试阶段准确率。

1.2.3 算法流程与步骤

本文 Dense-YOLOv3 模型的算法总体流程图如图 3 所示,分为 3 个阶段:数据的预处理(清洗、分类和标注)、对标注好的车型数据集进行训练和未标注的车型图片进行测试。

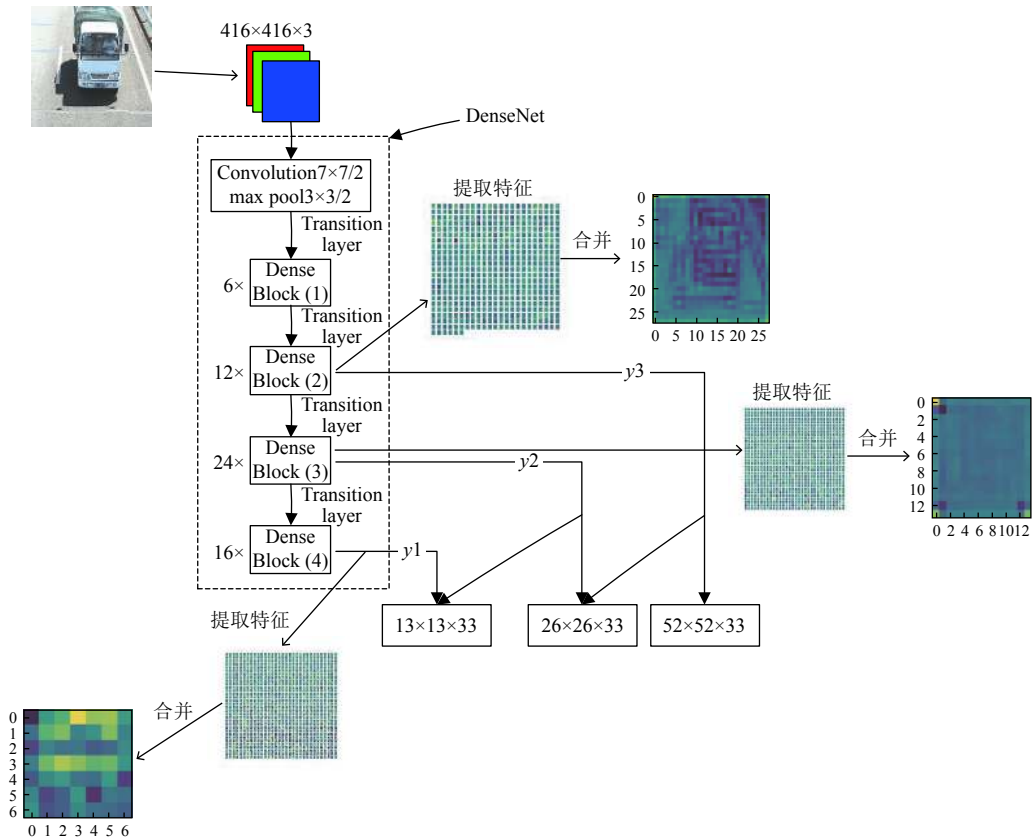


图 2 Dense-YOLOv3 模型

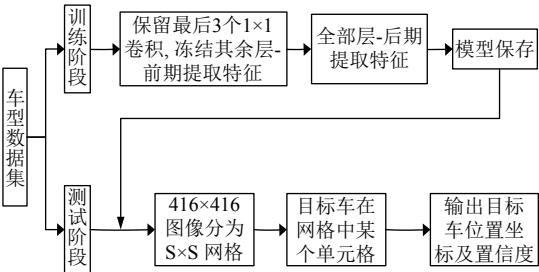


图 3 Dense-YOLOv3 模型流程图

算法具体步骤如下。

(1) 数据预处理

输入: BIT-Vehicle 原始数据集

输出: 已标注训练集 (目标物体车的位置、类别)

xml 文件和未标注的测试集

Step 1. 在 BIT-Vehicle 车型数据集 (如图 4) 中挑选具有代表性即不同光照强度和不同拍摄距离的图片, 随机划分训练集和测试集;

Step 2. 将训练集用 labelImg 工具进行标注形成 xml 文件;

Step 3. 利用代码形式训练集划分为两部分即 train 集和 val 集;

Step 4. 保留测试集图片。

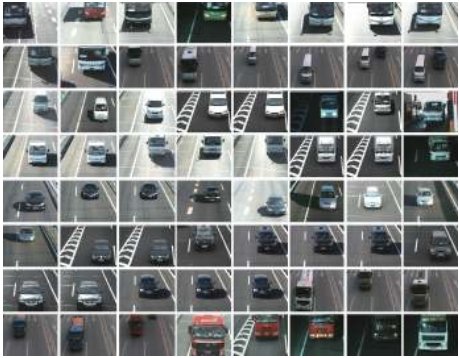


图 4 车型数据集

(2) 网络训练

输入: 已标注训练集 xml 文件与对应图片

输出: 训练权重模型

Step 1. 配置 cfg 文件, 设置学习率、类别数、batch 和 subdivisions 等超参数;

Step 2. 前期阶段, 冻结部分层进行训练;

Step 3. 一个 xml 文件经过卷积层、最大池化层和一系列密集卷积块与转移层提取车型特征;

Step 4. 将 Step3 中提取特征经过三个尺度的检测框进行特征局部交互;

Step 5. 计算出本文模型输出与真实值 y_true 之间的损失值;

Step 6. 返回 Step 3, 在训练过程中不断调整网络中的参数, 使得损失函数达到一个前期稳定的收敛值, 控制前期迭代次数结束前期训练;

Step 7. 开始后期阶段, 取消冻结层将进行全部层数的训练;

Step 8. 重复 Step 3~Step 5, 达到迭代次数停止训练.

(3) 网络测试

输入: 测试集、训练权重模型

输出: 目标车位置及类别

Step 1. 将测试集图片放入网络中;

Step 2. 进行前向传播, 得到预测框与相应的类别;

Step 3. 将网络预测框与真实框使用 IOU 技术得到接近真实框的预测框, 如图 5 所示.



图 5 IOU 技术展示

Step 4. Step 3 中的预测框与相应的类别求积得到置信度;

Step 5. 设置阈值, 过滤到低分的预测框, 对剩余的预测框进行 NMS 处理, 得到最后目标物体车结果;

Step 6. 循环 Step 1~Step 5, 直到预测完测试集即测试结束.

2 实验分析与结果

2.1 实验环境与数据集正文内容

硬件环境: 在系统类型为 Windows10 64 位操作系统和基于 x64 的处理器、计算机 CPU 型号为 Intel(R)

Xeon(R) W-2102 CPU@ 2.90 GHz、安装 RAM 的内存 16.0 GB, 以及使用 GPU device 0: GeForce RTX 2080 Ti, 设备上 68 个多处理器、全局内存总量为 11 264 MB, 每个线程块共享内存 48 KB、每个线程块最大线程数为 1024 上进行实验的运行.

软件环境: 采用基于 Theano 的深度学习框架 Keras, 是一个高度模块化的神经网络库, 简化了编程复杂度. 该框架有三大特性: 在操作方面建模简单; 配置简单; 相比其他框架, 对于不同模型算法, 微调步骤简单. 另外, 在算力上, 从 CPU 到 GPU 上进行加速不需要任何代码的改动, 且性能没有太多的损耗.

本文用于训练与测试数据集选取了少部分研究学者们经常实验用的 BIT-Vehicle 车型数据集, 一共有 6 种类型的车型. 该库中的图像均来自于实际路段的卡口高清图像, 与实际路况非常吻合, 且受拍摄距离、光照强度以及天气等的影响而拍摄到的多路况的车型图像, 更能突出深度卷积神经网络在检测效率上的优越性. 本次实验将每类车型数据集分为两组, 一组数据用来训练, 另一组用来测试. 在 BIT-Vehicle 车辆数据库中, 用于训练的车型集共有 5404 张, 按照类别 Bus、Microbus、Minivan、Sedan、SUV 和 Truck 划分为 810、700、748、1357、845 和 944 张; 用于模型测试有 3603 张车型集, 将其按类别排列顺序划分为 601、520、511、765、600 和 606 张车型图像. 具体车型数据集分配如表 2 所示.

表 2 实验数据分配

类别	训练集	测试集	总数
Bus	810	601	1411
Microbus	700	520	1220
Minivan	748	511	1259
Sedan	1357	765	2122
SUV	845	600	1445
Truck	944	606	1550
总数	5404	3603	9007

2.2 实验分析与对比

为了更好地验证 Dense-YOLOv3 在车型数据集上的网络性能和使用价值, 文中通过采用不同算法定性分析对比与实验数据的定量分析对比两部分进行研究.

2.2.1 定性分析

在本文中, 分别采用文献 [13]、YOLOv3 增加 4 层网络^[1]和 Dense-YOLOv3 模型在已经准备好的测试集上进行检测, 如表 3 所示, 其中, 选取具有代表性

即拍摄距离较远且相似车辆和光线不同的图片, 分别采用单通道直方图的重合度与差异值哈希法和亮度进行定量分析. 其中 YOLOv3 增加 4 层网络算法是在 Darknet-53 网络基础上再增加 12 层网络即 4 组卷积层+残差单元共同构成的网络结构. 理论上讲, 神经网络层数越大, 其表达能力越强即提取到的车型特征信息越多, 测试结果越显著, 但从表 3 中检测结果可以看出, 文献 [1] 检测效果并没有文献 [13] 那么明显.

从主观角度判断图片中目标物体车距离较远; 从客观的角度判断图片的相似度, 其由颜色和图像指纹相结合, 分别用单通道直方图的重合度与差异值哈希法 (dHash) 来进行表示. dHash 算法都是值越小, 相似度越高, 取值为 0-64, 即汉明距离中, 64 位的 hash 值有多少不同; 单通道直方图的值为 0-1, 值越大, 相似度越高. 两者进行结合, 更能判断不同算法检测效果. 从表 4 中看出, 所选取图片重合度为 0.604、0.798 和 0.602; dHash 值为 16、12 和 14.

图片亮度表示光线暗或者曝光程度如何, 一般规定一张图片的平均亮度为 128, 对于一张正常拍照拍出来的图片平均亮度应该是在 [100, 150]. 在表 4 中选取亮度为 33.99、20.31 和 166.95 的较暗与曝光的 3 张图片进行检测研究.

从表 3 可知, 对于检测同为距离较远的车, 本文模型相比于文献 [13]、文献 [1] 模型检测准确, 且检测出正确车型概率分别为 0.96、0.99 和 0.93, 而文献 [13]、文献 [1] 模型虽然检测出有目标物体存在, 但是未能正确检测出车型或者准确率比本文模型低. 光线较暗和曝光较强车型图片, 本文模型正确检测出车型且准确率高, 而文献 [13] 模型, 虽然正确检测出了车型, 但是准确率没有本文模型高, 从表 3 中看到, 检测出多余框, 准确率差; 文献 [1] 模型虽然在网络层数加大, 但是相较于文献 [13] 模型车型检测错误. 对于车型相似的图片, 文献 [13] 与文献 [1] 模型将 Minivan 错识别为 Truck, 而本文模型识别正确且准确率高. 总体来说, 本文模型不论在拍摄距离较远和光线较暗、较强情况下, 都能正确检测出目标物体车的类型且准确率高, 整体检测效果明显, 出现漏检的情况极少, 达到了预期的效果.

2.2.2 定量对比

为了进一步测试车型图片在本文模型中检测效果明显, 本文又分别利用文献 [13]、文献 [1] 和 Dense-

YOLOv3 网络模型在车型数据集上进行定量分析比较, 如表 4 所示.

在表 4 中通过采用性能指标平均精度 AP 、多类别平均精度 mAP 、准确率 $Precision$ 和召回率 $Recall$, 这些数值更能直观的反应出网络的性能如何. 准确率是指预测样本中的实际正样本数在所有正样本中的占比; 召回率是指预测样本中的实际正样本数占预测样本数的多少; 平均精度是由 Precision-Recall 曲线所围成的面积, 其公式为:

$$\int_0^1 p(r)dr = \sum_{k=1}^N P(k)\Delta r(k) \quad (2)$$

且本文算法各类的 $Recall$ 与 $Precision$ 之间的关系图如图 6 所示. 在式 (3) 中, 积分中 p 为准确率、 r 为召回率, N 代表此次实验所用到的测试集数即为 3603; 多类别平均精度 mAP 是将所有类的 AP 值进行平均, 公式为:

$$mAP = \frac{\sum_{q=1}^Q AP(q)}{Q} \quad (3)$$

在式 (4) 中, Q 代表实验数据集的类别数, 即为 6 类.

图 7 为表 4 中各个算法的 AP 与 mAP 直观图.

为了更直观综合评价算法好坏, 将精确率、召回率和 $F1$ 值这 3 个指标在不同算法上的对比图如图 8 所示. 从图 7 中的召回率和准确度可知, 本文算法相对于文献 [13]、文献 [1] 找到正确的 6 种车型图片要多, 并且准确率也相当高, 比传统的 YOLOv3 高达 3.8%, 说明本文模型综合利用密集卷积网络提取到更多的车型基本结构特征; 另外, 本文算法平均检测速度达到了 28 fps, 在与 DenseNet 网络集成后也保证了实时检测效果; 总体来看, 本文算法在不同类别的车型情况下, 各类的 AP 相对于其他文献 [13]、文献 [1] 两种算法占很大的优势, 且多类平均值高出 6.45%. 因此, 这些数据很好的验证了本文模型优越性和使用价值.

3 结论与展望

文中设计的 Dense-YOLOv3 模型通过将密集卷积神经网络作为主要车型特征的提取, 结合 YOLOv3 中的三尺度检测框进行局部信息的交互来实现目标车的

表3 不同算法的识别过程

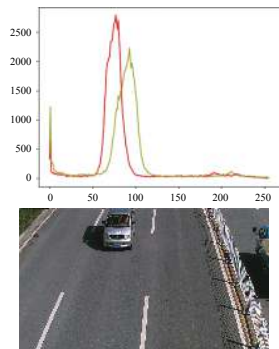
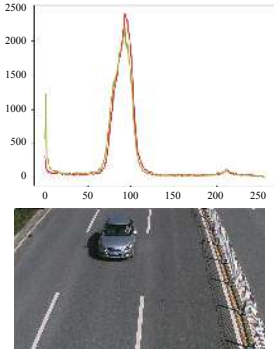
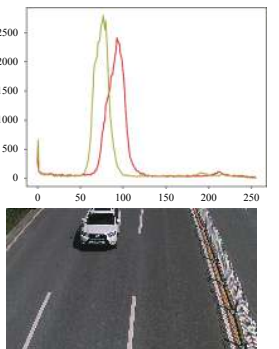

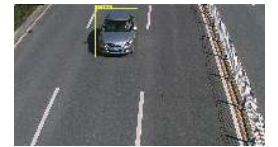



















示例图 (相似度)	<div>Microbus与SUV 直方图重合度: 0.604 dHash:16</div> 	<div>Microbus与Sedan 直方图重合度: 0.798 dHash:12</div> 	<div>Sedan与SUV 直方图重合度: 0.602 dHash:14</div> 	
	文献[13]			
文献[1]				
本文算法				
示例图 (亮度)	33.99	20.31	166.95	
				
	文献[13]			
	文献[1]			
	本文算法			

表 4 算法对比

算法	AP(%)						mAP(%)	Recall(%)	Precision(%)	FPS(fps)
	Bus	Microbus	Minivan	Sedan	SUV	Truck				
文献[13]	98.61	81.93	91.58	96.42	79.61	92.59	90.12	76.72	93.41	35
文献[1]	97.24	78.42	64.31	94.71	89.34	97.47	86.91	76.83	90.02	29
本文算法	98.88	95.77	95.72	95.80	95.51	97.75	96.57	93.30	97.21	28

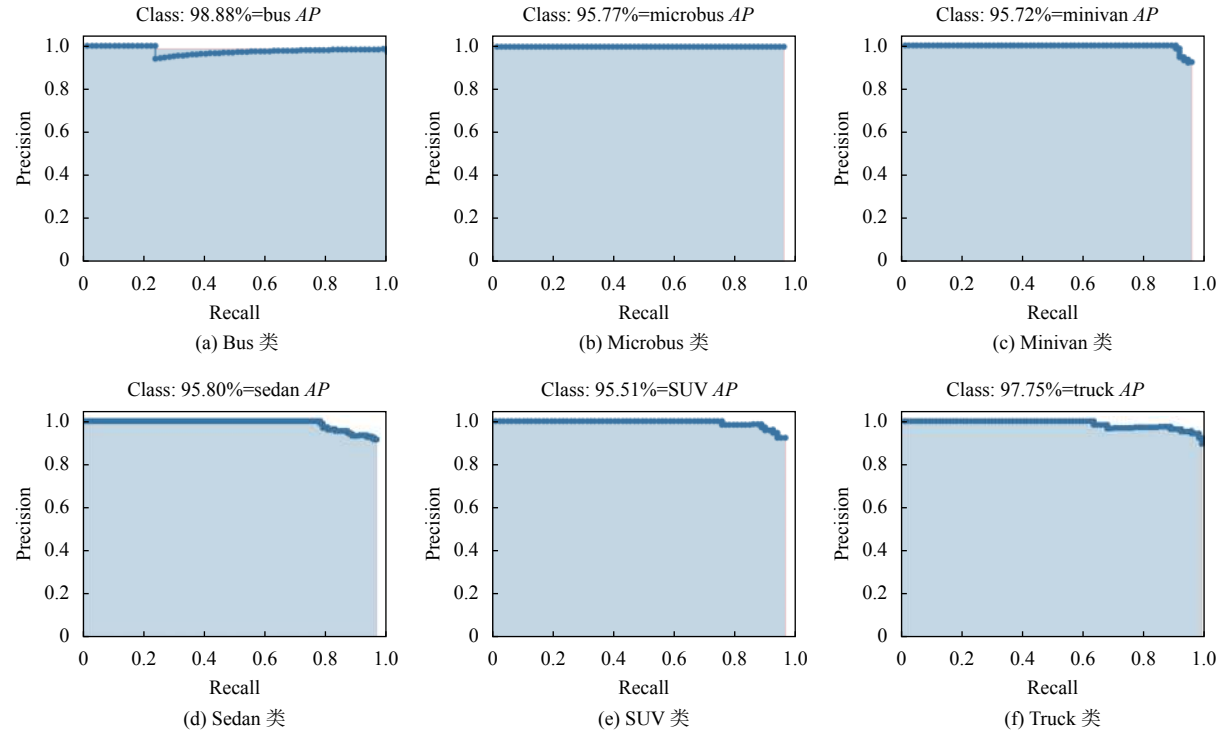


图 6 Precision-Recall 曲线

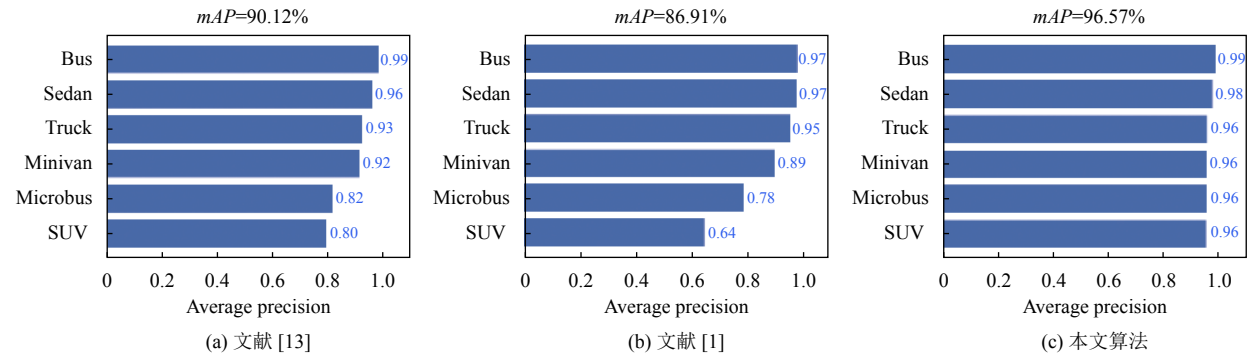


图 7 各算法 AP 与 mAP 图

检测,综合利用密集卷积网络的优势即在车辆存在异常环境下将提取到的车型特征信息进行重复利用,以此来提高了车型准确率,并且此模型在 BIT-Vehicle 车型数据集进行了实验验证.先采用 3 种不同的算法对拍摄距离较远和光线不同图片的检测过程进行定性分析;然后在车型数据集上利用目标检测指标性能参数

进行定量分析对比.通过定性和定量的实验分析,本文模型有较强的学习、抗过拟合能力,在实际应用中有很好的检测效果并且有一定的使用价值.但是,实验中的车型数据集有一些不足:车型数据集的种类较少、场景较为单一等的缺点.因此,在后续工作中将寻找更多样的、不同场景下车型数据集进行实验研究.

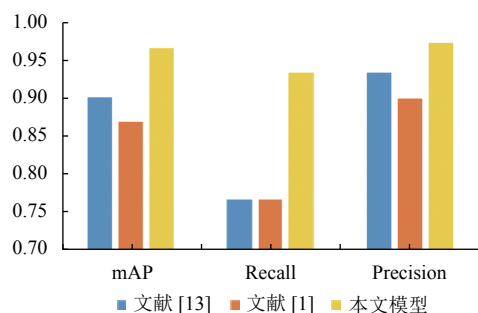


图8 各算法的准确率、召回率、F1值对比

参考文献

- 1 张富凯, 杨峰, 李策. 基于改进 YOLOv3 的快速车辆检测方法. 计算机工程与应用, 2019, 55(2): 12–20. [doi: 10.3778/j.issn.1002-8331.1810-0333]
- 2 石磊, 王亚敏, 曹仰杰, 等. 基于深度卷积神经网络的车型识别. 计算机科学, 2018, 45(5): 280–284.
- 3 何霞, 汤一平, 袁公萍, 等. 基于级联多任务深度学习的卡口识别引擎研究. 计算机科学, 2019, 46(1): 303–308. [doi: 10.11896/j.issn.1002-137X.2019.01.047]
- 4 王思元, 王俊杰. 基于改进 YOLOv3 算法的高密度人群目标实时检测方法研究. 安全与环境工程, 2019, 26(5): 194–200.
- 5 Girshick R, 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.
- 6 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, 2014, 37(9): 1904–1916.
- 7 Girshick R. Fast R-CNN. Proceedings of 2015 IEEE International Conference on Computer Vision. Santiago, Chile. 2015. 1440–1448.
- 8 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.
- 9 He KM, Gkioxari G, Dollár P, *et al.* Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 386–397.
- 10 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.
- 11 Liu W, Anguelov D, Erhan D, *et al.* SSD: Single shot MultiBox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam, the Netherlands. 2016. 21–37.
- 12 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.
- 13 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv: 1804.02767, 2018.
- 14 徐诚极, 王晓峰, 杨亚东. Attention-YOLO: 引入注意力机制的 YOLO 检测算法. 计算机工程与应用, 2019, 55(6): 13–23. [doi: 10.3778/j.issn.1002-8331.1812-0010]
- 15 王琳, 卫晨, 李伟山, 等. 结合金字塔池化模块的 YOLOv2 的井下行人检测. 计算机工程与应用, 2019, 55(3): 133–139. [doi: 10.3778/j.issn.1002-8331.1710-0236]
- 16 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.
- 17 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.