

一种高效的车体表面损伤检测分割算法

林少丹¹, 冯晨², 陈志德², 朱可欣³

(1. 福建船政交通职业学院信息与智慧交通学院, 福州 350007; 2. 福建师范大学数学与信息学院, 福州 350007; 3. 台湾中山大学资讯工程系, 高雄 80424)

摘要: 车体表面损伤检测是计算机视觉中的经典问题。车体表面损伤检测的主要瓶颈在于图像中损伤实例的不同尺度影响了分割的精度与效率。本文采用单阶段语义分割网络(YOLACT++)进行车体表面的损伤检测, 通过结合EfficientNet设计主干网络提高分割效率, 并通过改进损失函数优化YOLACT++中目标实例Mask的生成, 实验中用深度学习标注实验数据集进行训练测试。实验表明, 改进后的YOLACT++降低了Mask生成误差, 检测的实时帧率提高到35帧/s, 同时也提高了YOLACT++进行实例分割的精度。

关键词: YOLACT++; 主干网络; 高效网络; MaskIOU-NET; 损失函数

中图分类号: TP18 **文献标志码:** A

Efficient Damage Detection Segmentation Algorithm of Vehicle Body Surface

LIN Shaodan¹, FENG Chen², CHEN Zhide², ZHU Kexin³

(1. College of Information and Intelligent Transportation, Fujian Chuanzheng Communications College, Fuzhou 350007, China; 2. College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350007, China; 3. Department of Information Engineering, Sun Yat sen University of Taiwan, Kaohsiung 80424, China)

Abstract: Car body surface damage detection is a classic problem in computer vision. The main bottleneck of car body surface damage detection lies in the different scales of damage instances in the image, which affects the accuracy and efficiency of segmentation. In this paper, we use a single-stage semantic segmentation network (YOLACT++) for damage detection on the car body surface, combine EfficientNet to design a backbone network to improve segmentation efficiency, and improve the loss function optimization YOLACT++ to generate the target instance Mask in the experiment. Experimental data are marked by deep learning, and results show that the improved YOLACT++ detection frame rate is increased to 35 frame/s, which reduces the mask generation error and improves the instance segmentation accuracy of YOLACT++.

Key words: YOLACT++; backbone network; EfficientNet; MaskIOU-NET; loss function

引言

车体表面损伤检测可以用于车辆损害评估。传统车辆修复主要依赖于人工识别, 在面对车辆轻微

的划痕、裂缝或部分形变等车体表面的损伤时,完整检测的难度较大、效率较低。本文研究一种更为高效的车体表面损伤检测分割算法,能够大量应用到汽车车体表面美容、维修、车辆保养和车辆鉴定等涉及车体表面检测与识别方面的应用。

在物体目标图像识别领域,基于卷积神经网络的目标检测已经超越传统目标检测方法,成为当前目标检测的主流方法。单阶段目标检测将提取和检测合二为一,直接得到物体检测的结果,提高算法的效率,如YOLO、SSD等算法。而现有的实例分割算法,如:Fast-RCNN、Mask-RCNN等采用多阶段语义分割网络提高分割精度,但在实际车体表面损伤检测过程中,由于视角、遮挡和姿态等因素导致算法可用性受限。因此设计高准确率、高效率的车体表面损伤检测分割算法具有重大意义。

近年来,随着实例分割算法的不断推出,已有诸多研究将深度学习算法迁移至车辆检测领域中。Chen等^[1]使用YOLO的7个卷积层来检测单个类,解决了车牌检测问题,该算法车辆检测的准确率达到约98.22%,车牌识别的准确率达到78%;Hu等^[2]通过多任务学习在两个CNN中利用多种判别线索,提出了一种细粒度车辆类型识别的方法,将多个线索进行组合输入到卷积神经网络;吴迪等^[3]通过对颜色表示方法的深入研究,使用特殊的颜色空间合并与分解方法,提出了在室外正常光照条件下基于支持向量机的车辆颜色识别的方法;Zaarane等^[4]针对现有智能交通系统中视频车辆车型识别方法存在的误检率高、效率低的问题,提出一种基于二维离散小波变换(Two-dimensional discrete wavelet transform, 2D-DWT)的两阶段实时车辆检测算法;王威等^[5]提出了一个基于Mask-RCNN目标定位及掩膜生成算法;苏欣欣等^[6]针对传统无损检测技术在罐车内壁裂纹检测中效率低、抗干扰能力差等问题,提出一种基于卷积神经网络的热成像裂纹识别方法;Lu等^[7]提出了一种基于YOLO网络的视频对象实时检测算法,通过图像预处理消除图像背景的影响,训练快速的YOLO模型进行目标检测,获取目标信息,减少了参数的数目,大大缩短了目标检测的时间;叶玉婷等^[8]提出了一种基于侧面轮廓的实时车型分类系统,该分类系统分为3个部分:侧面轮廓处理、车型模板建立和实时车型匹配;Jamtsho等^[9]提出使用YOLO的实时车牌本地化,消除了看上去与LP相似的假阳性,在LP定位之前执行了车辆检测,单个卷积神经网络的整体平均精度为98.6%,而车辆和LP的训练损失为0.0231;Zhang等^[10]提出了一种基于轻量级CNN的车辆颜色识别方法,结果表明该方法识别率提高了0.7%以上,达到了95.41%,而特征向量的维数仅为18%,内存占用只有0.5%;Fu等^[11]提出了一种新的网络结构——基于残差学习的多尺度综合特征融合卷积神经网络(Multiscale comprehensive feature fusion convolutional neural network, MCFF-CNN),用于颜色特征提取;Huynh等^[12]提出了一种基于准自主视觉的方法来检测关键连接中的螺栓松动,并提出了一种基于区域卷积神经网络(Regional convolutional neural network, RCNN)的深度学习算法,以自动检测并裁剪连接图像中的可能螺栓。

在实际场景下的车体表面损伤检测过程中,所拍摄的多数相关图像存在背景复杂、车体表面形状差异较小以及拍摄角度造成相似度高等问题,因此在进行损伤检测时,传统的模型仍存在以下两方面问题:(1)效率较低,对于实际场景下的应用带来了诸多的不便;(2)目标查准率相对较低,检测效果较差。为了解决上述问题,本文提出了一种基于YOLACT⁺⁺的车体表面损伤检测算法,通过对YOLACT⁺⁺进行改进,生成一种单阶段语义分割网络。

1 相关工作

2019年Bolya等^[13]首次提出一种实时的实例分割算法——YOLACT,在MS-COCO数据集上检测帧率达到33帧/s的性能,作者通过实验对比研究了不同主干网络架构、原型数量以及图像分辨率等所导致的速度与性能之间的差异,并在此基础上提出了一种新颖的FAST-NMS方法,在性能损失可以忽

略不计的情况下,比传统NMS方法快12 ms。在YOLACT基础上,Bolya等^[14]提出了YOLACT++,它通过将可变形卷积合并到骨干网络中,并增加了一个快速掩码的评分计算分支——MASKIOU-NET, YOLACT++模型可以在MS-COCO数据集上以33.5帧/s的速度实现34.1 map,在保持准确率的同时也提升了算法的实时性。

1.1 YOLACT++模型

YOLACT++主要由BackBone、PredictionHeader、ProtoNet和SemanticSegmentation 4个部分组成,采用3通道550像素×550像素的图像作为输入到主干网络,通过FPN(Feature Pyramid network)输出到PredictionHeader和ProtoNet两个并行处理分支。ProtoNet分支使用全连接网络生成1组原图大小的原型掩膜,并与MaskIOU-Net生成的掩膜特征向量结合生成目标原型掩膜。PredictionHeader分支用于预测实例检测时生成的每个anchor的掩膜系数的向量、分类向量和anchor-box向量。最后,经过Fast-NMS处理后,将2个分支进行线性组合进行语义分割并生成目标掩膜。实验表明,将原型掩膜与掩膜模板系数线性组合可产生高质量的目标掩膜。

YOLACT++利用擅长产生语义向量的全连接层和擅长产生空间相干掩膜的卷积层来分别产生“掩膜系数”和“原型掩膜”,因为原型掩膜和掩膜系数都是独立计算的,所以主干网络的计算开销主要来自合成步骤,通过这种方式,可以在特征空间中保持空间一致性,同时使模型保持了单阶段的高效性,提高了算法的实时性,因此主干网络的优化是提升模型计算速度的有效优化方法。

1.2 YOLACT++的Loss函数

YOLACT++损失函数主要包含分类损失、边界框回归损失和Mask损失3部分损失函数相加所得,即有

$$L_{\text{YOLACT}} = L_{\text{class}} + L_{\text{box}} + L_{\text{mask}} \quad (1)$$

式中: L_{class} 、 L_{box} 和 L_{mask} 分别为分类损失函数、预测框损失函数和掩膜生成损失函数,其中掩膜生成损失函数由通过取得目标原型Mask与相对应的原型Mask系数相乘,加入非线性激励函数后, YOLACT++采用预测的Mask和真实的Mask两者像素集的二值交叉熵生成Mask的损失函数值,其求导过程为

$$-\sum_i p_i^* \log(p_i, p_i^*) = -\sum_i (p_i \times \log(p_i^*) + (1 - p_i) \log(1 - p_i^*)) \quad (2)$$

式中: p_i 为通过预测的Mask值; p_i^* 为原数据集中标注的Mask值(实际的Mask值)。生成Mask的损失值通过非线性激励函数sigmoid^[15]后所得,即

$$L_{\text{mask}} = \text{sigmoid}(-p_i^* \log(p_i, p_i^*)) \quad (3)$$

通过激励函数sigmoid得到Mask的损失值,其sigmoid激励函数为

$$\text{sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (4)$$

sigmoid函数曲线如图1所示。

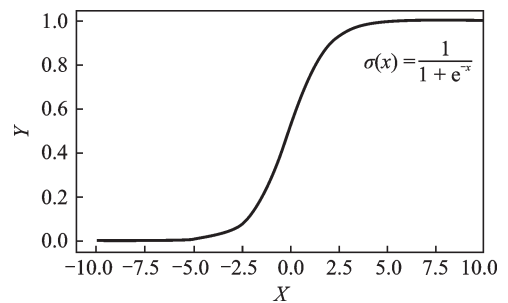


图1 Sigmoid激励函数曲线

Fig.1 Sigmoid activation function curve

2 基于改进YOLACT++的车体表面损伤检测

本文为解决车体表面的损伤检测的精度和效率问题,提出了一种改进的YOLACT++的车体表面的损伤检测模型,算法结构如图2所示。在该算法中采用了EfficientNet作为主干网络,使得YOLACT++网络层次变深,提高了效率和精度,同时为解决Mask生成梯度损失值较高、Mask精度较低等问题,通过对Mask损失函数进行优化,降低梯度损失值并提高生成Mask精度。

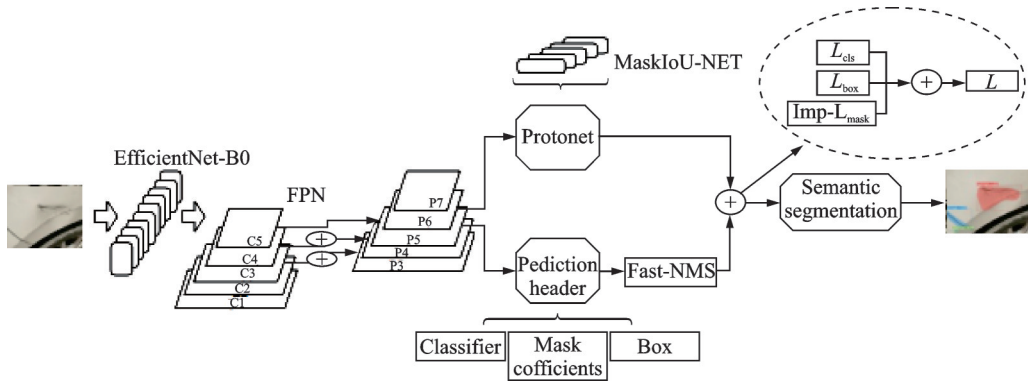


图2 基于改进YOLACT++的车体表面的损伤检测

Fig.2 Damage detection of car body surface based on improved YOLACT++

2.1 主干网络优化

在对车体表面损伤目标定位的YOLACT++结构中,根据越深的网络效果越好原则进行主干网络选择,采用EfficientNet-B0+FPN作为用于特征映射的基础主干网络,选择C1、C2、C3作为与EfficientNet的对接层,并设置每个层相应输入通道数,增加anchors数量,从而提高目标检测的准确率。

EfficientNet^[16]提出了尺度均匀缩放所有维度的方法,根据图像的深度、宽度、分辨率形成有效复合系数 ϕ ,动态调整网络深度和宽度,定义如下:depth: $d = \alpha^\phi$; width: $w = \beta^\phi$; resolution: $r = \gamma^\phi$ s.t. $\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2, \alpha \geq 1, \beta \geq 1, \gamma \geq 1$ 。

本算法采用EfficientNet的特性,形成适用YOLACT++的主干网络的深度和宽度参数,从而有效地提升YOLACT++的效率和准确性。实验中需要注意的是计算模型所对应的接入层数及参数,即有

$$L_{(s_1, s_2, s_3)} = \sum_{i \in \{1, 8\}} \{\text{ceil}(d_m \times l_i) - 1\} \quad (5)$$

式中: $L_{(s_1, s_2, s_3)}$ 代表3个阶段对应的层数; d_m 代表模型的深度系数; l_i 代表层数;ceil代表取整函数。可以通过宽度系数计算出各模型的输入通道数,即

$$I_m = b \times w_m \quad (6)$$

式中: I_m 代表模型输入通道数; w_m 代表模型的宽度系数; b 为EfficientNet-B0的输入通道数320,作为一个基线系数。

通过式(5,6)计算得到在YOLACT++时选择的接入层为[4, 10, 15],以及所需的EfficientNet-B0的EfficientNet深度系数、宽度系数分别为1.0和1.0,输入通道数为320。

2.2 Loss函数改进

YOLACT++采用sigmoid函数作为二值交叉熵的激励函数的优点在于,它的输出映射在(0,1)内单调连续,适合用作输出层且容易求导,但是因为软饱和性,一旦输入落入饱和区,导数就会变得接近于0,很容易产生梯度消失^[17]。当预测值与实际值相差较大时,采用交叉熵损失函数造成误差增大。

本算法采用交叉熵与 L_1 范数、 L_2 范数相结合的梯度计算方式,主要是针对当预测框与ground truth差别过大时,梯度值不至于过大;当预测框与ground truth差别很小时,梯度值也能够足够小。针对输入 $x < 1$ 的部分使用 L_1 范数(式(7)),使梯度更平滑且求导方便,针对输入 $x > 1$ 的部分使用 L_2 范数(式

(8)), 有效地避免梯度爆炸, 减少离群值的出现^[18]。

计算 Loss 前必须先设置 anchors 的正、负样本标定规则, 假设 anchor 对应的预测框与 GT 的重叠度 $\text{IoU} > 0.7$, 标记为正样本; 假设 anchor 对应的预测框与 GT 的重叠度 $\text{IoU} < 0.3$, 标记为负样本; 余下的样本既不属于正样本也属于负样本, 负样本不参与最终训练。

$$\text{smooth}_{L_1}(x) = 0.5 \times x^2 \quad |x| < 1 \quad (7)$$

$$\text{smooth}_{L_2}(x) = |x| - 0.5 \quad |x| > 1 \quad (8)$$

不妨定义 Mask 定位的损失函数为 $L_{\text{mask_loc}}$, 预测平移缩放参数为 t^u , 真实平移缩放参数为 v , 通过 2 个参数的比较来评估检测框定位的损失率。

$$L_{\text{mask_loc}}(t^u, v) = \sum_{i \in \{x, y, w, h\}} \text{smooth}_{L_{(1,2)}}(t_i^u - v_i) \quad (9)$$

在车辆表面的掩膜分类中仍存在两个缺点: 首先所有像素在评估损失函数中起着相同的作用, 这可能会忽略 p_i 的特殊位置信息。其次, 原损失函数更适合平衡正例/负例的情况, 而大多数车辆划痕的数据集可能无法满足此要求。针对上述问题, 本算法采用改进的交叉熵损失函数来考虑像素空间位置的影响以及严重的不平衡正负例的影响。改进 Mask 的损失率定义为

$$-\sum_i p_i^* \log(p_i, p_i^*) = -\sum_i (\alpha_1 p_i \times \log(p_i^*) + \alpha_2 e^{-g(l_i)} (1-p_i) \log(1-p_i^*)) \quad (10)$$

式中: p_i 表示预测的第 i 个锚框是一个目标物体的概率, 发现目标时 p_i 为 1, 否则 p_i 为 0。 $g(l_i)$ 为约束系数, 当 $l_i = 0$ 时, $g(l_i) = 0$; 当 $0 < l_i < T$ 时, $g(l_i) = \frac{l_i}{\max_{j \in I} \{l_j\}}$; 当 $l_i > T$ 时, $g(l_i) = \frac{T}{\max_{j \in I} \{l_j\}}$, l_i 表示第 i 个像素与划痕的欧氏距离, $T = 0.3 \max \{l_i\}$ 为像素与划痕距离的阈值。 α_1, α_2 为正负样本的比例。

最后通过式 (10) 和式 (9) 得到最终的损失, 即有

$$L(\{p_i\}, \{t_i\}) = \sum_i L_{\text{mask_loc}}(p_i, p_i^*) + \lambda \times -\sum_i p_i^* \log(p_i, p_i^*) \quad (11)$$

式中: t_i 为对应于正锚点的 GT 坐标; $L_{\text{mask_loc}}$ 为预测边界框和 GT 框的 $\text{smooth}_{L_{(1,2)}}$ 损失函数。 Mask 的总损失率归一化由 smooth_loss 、 $\text{binary_cross_entropy}$ 和 λ (平衡权重) 合并得到^[19], 本文将 λ 设置为 0.5, 这样设置可以使 Mask 的两部分 loss 值保持平衡, 最后通过训练获得一系列 Mask 特征区域。

2.3 车体表面损伤目标定位解决方案

本节介绍了车体表面损伤目标定位问题的解决方案^[20], 过程如图 3 所示, 主要分为 6 步:

(1) 图像输入。通过视频采集生成大小为 550 像素 \times 550 像素的车体表面损伤区域图像 (图 3(a)), 输入到主干网络中进行特征提取, 并将提取的特征张量送入 PredictionHeader 和 ProtoNet 两个并行处理分支中处理。

(2) 生成原型掩膜。通过 ProtoNet 对特征区域进行一系列卷积操作后生成原型掩膜 (图 3(b))。

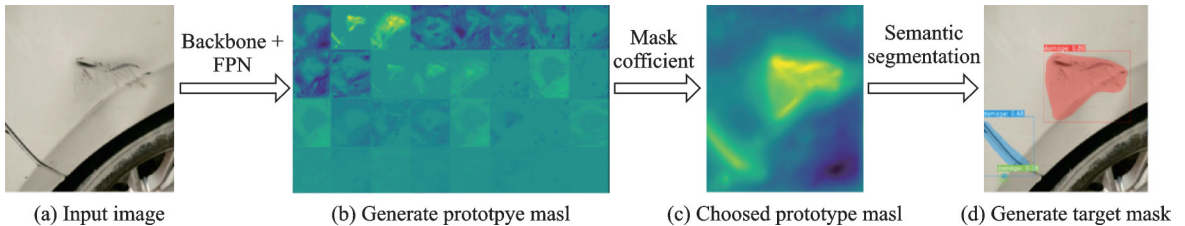


图 3 车体表面损伤目标定位过程

Fig. 3 Target location process of vehicle body damage

(3) 生成原型掩膜系数。通过 PredictionHeader 对一系列原型掩膜生成对应掩膜系数。

(4) 原型掩膜提取。通过原型掩膜张量与掩膜系数计算出最优的原型掩膜(图 3(c))。

(5) 目标分割。结合原型掩膜的目标定位,分割出损伤区域范围。

(6) 生成目标掩膜。在分割出的目标上生成相应的掩膜(图 3(d)),从而实现车体表面损伤区域识别定位的过程。

3 实验结果

3.1 数据集准备

本实验所用数据集是利用爬虫程序从网络上爬取的车辆损伤图片,由于 YOLACT⁺⁺对输入图片大小要求,将数据集图片改成 550 像素×550 像素大小,并使用 LabelMe 对图像中的车体表面损伤目标所在的区域进行标注。LabelMe 是由麻省理工学院计算机科学与人工智能实验室(MIT CSAIL)创建的一个开放的注释工具^[21]。通过 LabelMe 软件,能够方便地完成对图像中各种车体表面损伤区域目标边界的标注并保存图像。实验前首先使用 LabelMe 制作 MS-COCO 标准数据集,其中包括训练集 30 000 张,验证集 15 000 张。

3.2 实验参数

在本实验中训练周期设置为 127 次,每周期内迭代 100 步,训练步数通常根据数据集规模大小而定。采用上述改进的 YOLACT⁺⁺模型进行大规模的数据集训练,需要耗费大量的内存资源和时间,对硬件要求较高,用于训练的硬件环境参数为 2 颗 CPU: Intel(R) Xeon(R) Gold-6150 72 核,内存 1 TB, 3 颗 GPU, NVIDIA tesla v100;软件环境采用 Python3.6, pytorch1.1, cuda10.1。上述软硬件环境结合改进后的 YOLACT⁺⁺网络进行训练,较之前的 YOLACT⁺⁺架构在训练所用时间及梯度损失率等方面均有一定优化。

3.3 主干网络优化分析

基于原算法复杂的主干网络带来的效率问题,本算法对主干网络进行了优化,并对掩膜系数生成、原型掩膜生成以及掩膜生成 3 个过程中的激励函数进行改进,分别进行训练。首先,在不同主干网的对比实验发现,主干网络的更换提高了整个模型对目标识别平均准确度,在实验中分别采用 ResNet101/50、DarkNet53、VGG16、MobileNetV2 和 EfficientNet 等作为模型主干网络进行训练。当主干网络设置为 EfficientNet 时,有效地降低了 YOLACT⁺⁺的网络规模,同时采用加大训练规模和调整学习率等方法,有效地解决了大型网络的梯度冻结问题。在本算法中将 EfficientNet-B0 作为主干网络,生成模型总大小为 46.1 MB,小于采用 ResNet-50 的 129.9 MB, EfficientNet-B0 的参数大小为 10.93 MB,小于 ResNet-50 的 20.2 MB。经过 127 个周期的训练,达到 35.57 帧/s,且还未完全收敛。值得注意的是,生成的 YOLACT⁵⁵⁰⁺⁺-EfficientNet-B0 模型大小仅为 46.1 MB。表 1 为各主干网络训练参数对比。

3.4 损失函数优化分析

3.3 节中分析了采用 EfficientNet+FPN 作为主干网络所带来的模型压缩和训练效率上的优势,但主干网络的更换也使得损失值升高。为解决这一问题,本文采用改进的损失函数降低梯度损失值,提高了检测效果和精度。改进前后的梯度损失值对比如表 2 所示。

从表 2 中可以看出,基于改进的损失函数训练模型梯度损失值均小于原损失函数的梯度损失值,同时表 3 给出了损失函数改进前后的平均精度均值(Mean average precision, mAP)对比。实验结果表明,改进的损失函数对于模型精度的提升较为明显。

同时,与传统的主干网络相对比,损失函数的改进使得本模型对 Mask 的梯度损失率的优化效果更

表 1 不同主干网络 YOLACT⁺⁺模型参数表

Table 1 YOLACT⁺⁺ model parameter table of different backbone networks

Basic CNN	Parameter size/MB	Model size/MB	检测帧率/(帧·s ⁻¹)	预计到达时间
ResNet101	41.79	205.4	23.16	18 h 23 min 22 s
ResNet50	20.20	129.9	26.47	15 h 43 min 54 s
MobileNetV2	7.56	31.9	35.21	8 h 36 min 17 s
VGG16	60.39	253.3	34.95	11 h 28 min 0 s
DarkNet53	45.40	190.7	29.33	13 h 38 min 28 s
EfficientNet-B0	10.93	46.1	35.57	5 h 53 min 24 s

表 2 改进前后 YOLACT⁺⁺梯度损失值对比表

Table 2 Comparison table of YOLACT⁺⁺ gradient loss before and after improvement

模型	损失函数		
	Box-loss	Mask-loss	Class-loss
EfficientNet-B0	2.10	3.40	2.15
EfficientNet-B0(Loss improved)	2.08	2.03	2.14

表 3 采用不同损失函数的 mAP 比较表

Table 3 Comparison of mAPs using different loss functions

模型		IOU _{All}	IOU _{0.50}	IOU _{0.55}	IOU _{0.60}	IOU _{0.65}	IOU _{0.70}	IOU _{0.75}	IOU _{0.80}	IOU _{0.85}	IOU _{0.90}	IOU _{0.95}
EfficientNet-B0	Box	16.01	41.01	35.18	28.77	21.63	15.61	9.91	4.77	1.88	1.10	0.25
	Mask	9.14	29.95	23.27	16.77	10.73	6.00	2.75	1.37	0.33	0.25	0.00
EfficientNet-B0 (Loss improved)	Box	16.95	42.58	36.81	30.55	23.39	16.72	10.62	5.48	2.08	1.08	0.20
	Mask	9.25	31.82	23.61	17.03	10.27	5.43	2.22	0.99	0.50	0.40	0.20

为显著,生成梯度损失走势,其对比图如图 4 所示。从图 4 中可以看出,改进实验模型所产生的 Mask 梯度损失值最低,效果更好。

表 4 给出了改进的 YOLACT⁺⁺与其他不同主干网络的 YOLACT⁺⁺梯度损失值,比较结果展示了本实验在 Mask 上损失值最低,优化效果较为显著。

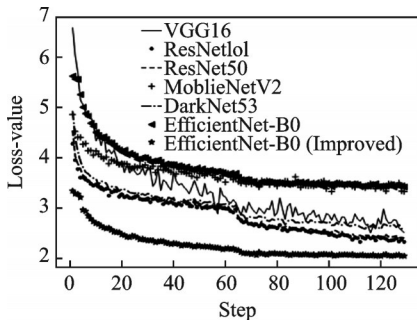


图 4 Mask 生成梯度损失率走势对比图

Fig.4 Comparison of trend of gradient loss rate generated by the Mask

表 4 不同主干网络的 Mask 损失值对比表

Table 4 Mask loss comparison of different backbone networks

Basic CNN	Mask-loss
ResNet101	2.31
ResNet50	2.39
MobileNetV2	3.32
VGG16	2.49
DarkNet53	2.56
EfficientNet-B0(Improved)	2.03

另外,在训练中将主干网络 VGG16、MobileNetV2 和 EfficientNet-B0 分别与 MaskIOU-Net 生成的方法相结合,并对训练结果的 mAP 进行对比,比较结果见表 5。

表 5 采用不同基础主干网络的 mAP 比较表
Table 5 Comparison of mAP using different basic backbone networks

Basic CNN		IOU _{All}	IOU _{0.50}	IOU _{0.55}	IOU _{0.60}	IOU _{0.65}	IOU _{0.70}	IOU _{0.75}	IOU _{0.80}	IOU _{0.85}	IOU _{0.90}	IOU _{0.95}
MobileNetV2	Box	6.55	20.32	15.89	11.43	7.93	4.68	2.67	1.48	1.07	0.03	0.01
	Mask	3.63	14.49	10.34	6.02	3.34	1.51	0.44	0.08	0.03	0.00	0.00
VGG16	Box	15.30	40.67	34.68	28.58	21.22	14.57	8.53	3.56	0.98	0.15	0.10
	Mask	9.27	29.16	22.99	16.92	11.24	6.76	3.37	1.26	0.50	0.50	0.03
EfficientNet-B0(Im- proved)	Box	16.95	42.58	36.81	30.55	23.39	16.72	10.62	5.48	2.08	1.08	0.20
	Mask	9.25	31.82	23.61	17.03	10.27	5.43	2.22	0.99	0.50	0.40	0.20

由表 5 可见,采用 EfficientNet 跟移动端的主流模型 MobileNetV2 和 VGG16 相比,识别精度有了显著提升。说明本模型在保证精准的识别效果的前提下,进一步削减了模型的规模,并显著地提高了模型的识别速度。特别是在移动端的应用上,比其他模型更快速、更精确,有着很好的应用前景。

3.5 查准率分析

从数据集中提取了 1 000 张图像作为测试集,将本算法与其他方法的多组对照实验结果进行比较。评估指标主要包括查准率(Precision)以及召回率(Recall),利用这两个指标来衡量模型对车体表面损伤目标定位效果,表达式为

$$\text{Precision} = \frac{TP}{TP + FP} \tag{12}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{13}$$

式中:TP 代表被期望为正样本的,实际识别也为正样本;FP 代表期望为正样本,但实际识别为负样本。TP 取准确定位出车体表面损伤目标的图像数,FP 取没有定位出车体表面损伤目标或定位出部分车体表面损伤目标的图像数,FN 取完全没有定位出车体表面损伤目标,通常采用测试精度低于某个比较低阈值的图像数。

当阈值取 0.80 时,TP 为识别率 ≥ 0.80 的数量,FP 为 $0.75 \leq \text{识别率} \leq 0.8$ 的数量,FN 为识别率 < 0.75 的数量;当阈值取 0.75 时,TP 为识别率 ≥ 0.75 的数量,FP 为 $0.70 \leq \text{识别率} \leq 0.75$ 的数量,FN 为识别率 < 0.70 的数量;根据以上取值范围,对改进前后的方法进行测试,实验采用低于 0.75 测试精度的作为 FN,图 5 为不同方法测试查准率对比。

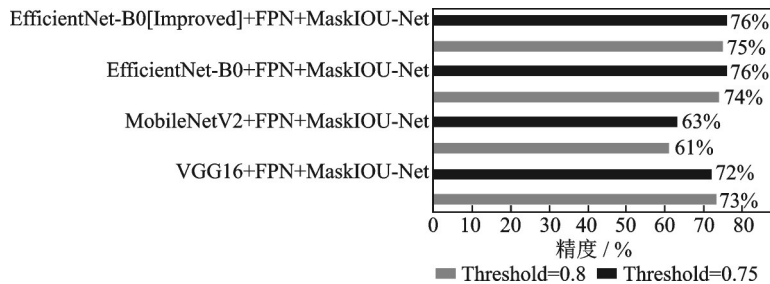


图 5 不同方法测试查准率对比

Fig.5 Comparison of test accuracy of different methods

从图5可知,采用EfficientNet-B0+FPN+MaskIOU-Net方法生成的模型测试查准率达76%,比采用MobileNetV2、VGG16作为基础主干网络的模型测试查准率提高了3%~13%,说明查准率已超过一些轻量级主干网络。

3.6 效率分析

通过与现有的实例分割模型的实时性和准确性对比可知,本算法不仅可以获得较高的查准率,同时也降低了模型的响应延迟,保持了实时性,如图6所示。从图6中可以看出,改进后的YOLACT++检测帧率达到35.57帧/s,相较于其他模型速度提升显著,而Mask mAP比其他3个模型相差无几。因此,可以证明改进后的YOLACT++具备快速分割定位的特点。将本算法运用在汽车整形维修、车辆快速理赔、二手车市场以及二手车抵押贷款等领域具有重要的理论价值和应用价值。

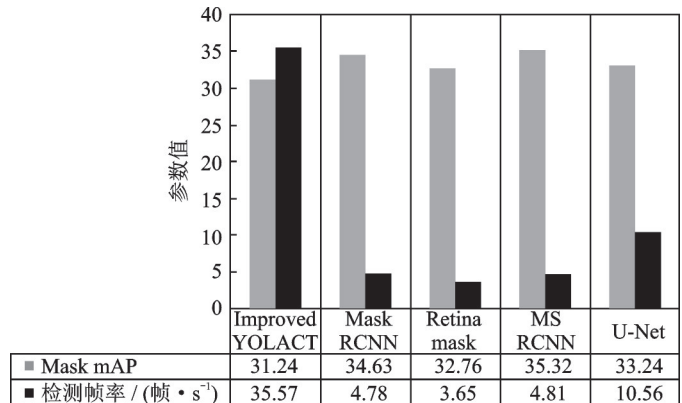


图6 不同模型实时性参数对比

Fig.6 Comparison of real-time parameters for different models

4 结束语

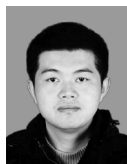
针对实际场景中车体表面损伤的目标检测效率与精度问题,本文改进YOLACT++中对原型掩膜生成的激励函数、掩膜系数生成的激励函数以及掩膜生成的激励函数,提出一种基于改进的YOLACT++的车体表面损伤检测算法。通过与ResNet-101、ResNet-50、VGG16、MobileNetV2以及EfficientNet-B0五种基础主干网的实验对比,证明了改进的YOLACT++算法对车体表面损伤识别的准确性和检测效率。同时测试表明,利用EfficientNet-B0作为主干网络,结合能加快掩膜生成的MaskIOU-Net,并改进掩膜生成损失函数的YOLACT++模型,可以大幅提高检测帧率,有效地提高了目标识别和定位的实时性。但是模型准确率仍有待提高,在后续的工作中将采用更高效的主干网络,结合采用更大样本库的方法进一步提升模型的准确率。

参考文献:

- [1] CHEN R C. Automatic license plate recognition via sliding-window darknet-YOLO deep learning[J]. *Image and Vision Computing*, 2019, 87: 47-56.
- [2] HU B, LAI J H, GUO C C. Location-aware fine-grained vehicle type recognition using multi-task deep networks[J]. *Neurocomputing*, 2017, 243: 60-68.
- [3] 吴迪,蔡晓东,华娜,等.基于HOG和SVM的级别自适应车型识别算法[J]. *桂林电子科技大学学报*, 2016, 36(1): 23-28.
WU Di, CAI Xiaodong, HUA Na, et al. Level adaptive vehicle recognition algorithm based on HOG and SVM[J]. *Journal of Guilin University of Electronic Science and Technology*, 2016, 36(1): 23-28.
- [4] ZAARANE A, SLIMANI I, HAMDOUN A, et al. Real-time vehicle detection using cross-correlation and 2D-DWT for feature extraction[J]. *Journal of Electrical and Computer Engineering*, 2019(1): 1-9.
- [5] 王威,李青,孙叶青,等.基于卷积神经网络的红外热成像罐车内壁裂纹识别[J]. *红外技术*, 2018, 40(12): 1198-1205.
WANG Wei, LI Qing, SUN Yeqing, et al. Identification of cracks on the inner wall of tank car by infrared thermal imaging based on convolution neural network[J]. *Infrared Technology*, 2018, 40 (12): 1198-1205.
- [6] 苏欣欣,蒋行国,蔡晓东.基于局部特征的车辆二次识别方法[J]. *桂林电子科技大学学报*, 2018, 38(6): 469-475.
SU Xinxin, JIANG Xingguo, CAI Xiaodong. Vehicle secondary recognition method based on local features[J]. *Journal of Gui-*

- lin University of Electronic Science and Technology, 2018, 38(6): 469-475.
- [7] LU S, WANG B, WANG H, et al. A real-time object detection algorithm for video[J]. Computers & Electrical Engineering, 2019, 77: 398-408.
- [8] 叶玉婷,王文鼎. 基于侧面轮廓的实时车型分类系统[J]. 微型机与应用, 2016, 35(10): 50-51, 54.
YE Yuting, WANG Wennai. Real-time vehicle classification system based on side profile[J]. Microcomputer and Application, 2016, 35(10): 50-51, 54.
- [9] JAMTSHO Y, RIYAMONGKOL P, WARANUSAST R. Real-time Bhutanese license plate localization using YOLO[J]. ICT Express, 2019, 6(2): 121-124.
- [10] ZHANG Q, ZHUO L, LI J, et al. Vehicle color recognition using multiple-layer feature representations of lightweight convolutional neural network[J]. Signal Processing, 2018, 147: 146-153.
- [11] FU H, MA H, WANG G, et al. MCFF-CNN: Multiscale comprehensive feature fusion convolutional neural network for vehicle color recognition based on residual learning[J]. Neurocomputing, 2020, 395: 178-187.
- [12] HUYNH T C, PARK J H, JUNG H J, et al. Quasi-autonomous bolt-loosening detection method using vision-based deep learning and image processing[J]. Automation in Construction, 2019, 105: 102844.
- [13] BOLYA D, ZHOU C, XIAO F, et al. YOLACT: Real-time instance segmentation[C]//Proceedings of the IEEE International Conference on Computer Vision. Davis, California: IEEE, 2019: 9157-9166.
- [14] BOLYA D, ZHOU C, XIAO F, et al. YOLACT++: Better real-time instance segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 99: 1.
- [15] PRETORIUS A M, BARNARD E, DAVEL M H. ReLU and sigmoidal activation functions[C]//Proceedings of South African Forum for Artificial Intelligence Research (FAIR2019). South Africa: North-West University, 2019: 37-48.
- [16] TAN M, LE Q V. Efficientnet: Rethinking model scaling for convolutional neural networks[EB/OL]. (2019-05-28)[2020-09-11]. <https://arXiv.org/abs/1905.11946v1>.
- [17] PARK S, SHIN Y G, KO S J. Contrast enhancement using sensitivity model-based sigmoid function[J]. IEEE Access, 2019, 7: 161573-161583.
- [18] SHAO DAN L, CHEN F, ZHIDE C. A ship target location and mask generation algorithms base on mask RCNN[J]. International Journal of Computational Intelligence Systems, 2019, 12(2): 1134-1143.
- [19] LIN K W E, BALAMURALI B T, KOH E, et al. Singing voice separation using a deep convolutional neural network trained by ideal binary mask and cross entropy[J]. Neural Computing and Applications, 2020, 32(4): 1037-1050.
- [20] ZHANG Q, CHANG X, BIAN S B. Vehicle-damage-detection segmentation algorithm based on improved mask RCNN[J]. IEEE Access, 2020, 8: 6997-7004.
- [21] BRYAN C, RUSSELL A T, KEVIN P, et al. LabelMe: A database and web-based tool for image annotation[J]. International Journal of Computer Vision, 2008, 77(1/2/3): 109.

作者简介:



林少丹(1979-),通信作者,男,副教授,研究方向:计算机视觉识别、人工智能, E-mail: linshaodan66@qq.com。



冯晨(1996-),男,硕士研究生,研究方向:网络安全。



陈志德(1976-),男,教授,研究方向:网络安全。



朱可欣(1995-),男,硕士研究生,研究方向:算法与数据结构。