计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(12):127−134 [doi: 10.15888/j.cnki.csa.008758]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于改进 YOLOv5 的车辆端目标检测①

黎国溥[1], 陈升东[1], 王　亮[2], 邹　凯[1], 袁　峰[1]

[1](广州软件应用技术研究院, 广州 511458)
[2](贵阳信息技术研究院, 贵阳 550081)
通信作者: 陈升东, E-mail: chenshengdong@gz.iscas.ac.cn

摘　要: 在自动驾驶应用场景下, 将 YOLOv5 应用于目标检测中, 性能较之前版本有明显的提升, 但在高运行速度情况下检测精度仍不够高, 本文提出一种基于改进 YOLOv5 的车辆端目标检测方法. 为解决训练不同数据集时需手动设计初始锚框大小, 引入自适应锚框计算. 在主干网络 (backbone) 添加压缩与激励模块 (squeeze and excitation, SE), 筛选针对通道的特征信息, 提升特征表达能力. 为了提升检测不同大小物体时的精度, 将注意力机制与检测网络融合, 把卷积注意力模块 (convolutional block attention module, CBAM) 与 Neck 部分融合, 使模型在检测不同大小的物体时能关注重要的特征, 提升特征提取能力. 在主干网络中使用空间金字塔池化 SPP 模块, 使得模型输入可以输入任意图像高宽比和大小. 在激活函数方面, 进行卷积操作后使用 Hardswish 激活函数, 应用于整个网络模型. 在损失函数方面, 使用 *CIoU* 作为检测框回归的损失函数, 改善定位精度低和训练过程中目标检测框回归速度慢的问题. 实验结果表明, 改进后的检测模型在 KITTI 2D 数据集上测试, 目标检测的精确率 (precision) 提高了 2.5%, 召回率 (recall) 提高了 5.1%, 平均精度均值 (mean average precision, mAP) 提高了 2.3%.

关键词: 目标检测; YOLOv5; 压缩与激励模块; 注意力机制; 卷积注意力模块; 激活函数; Hardswish

引用格式: 黎国溥,陈升东,王亮,邹凯,袁峰.基于改进 YOLOv5 的车辆端目标检测.计算机系统应用,2022,31(12):127–134. http://www.c-s-a.org.cn/1003-3254/8758.html

Vehicle-side Target Detection Based on Improved YOLOv5

LI Guo-Pu[1], CHEN Sheng-Dong[1], WANG Liang[2], ZOU Kai[1], YUAN Feng[1]

[1](Guangzhou Institute of Software Application Technology, Guangzhou 511458, China)
[2](Guiyang Academy of Information Technology, Guiyang 550081, China)

Abstract: In the application scenario of autonomous driving, YOLOv5 is applied to target detection, and the performance is significantly improved compared with that of previous versions. However, the detection accuracy is still low in the case of high running speed. This study proposes a vehicle-side target detection method based on improved YOLOv5. In order to address the issue of manually designing the initial anchor box size in training different datasets, an adaptive anchor box calculation is introduced. In addition, a squeeze and excitation (SE) module is added to the backbone network to screen the feature information for channels and improve the feature expression ability. In order to improve the accuracy of detecting objects of different sizes, the attention mechanism is integrated with the detection network, and the convolutional block attention module (CBAM) is integrated with the Neck part. As a result, the model can focus on important features when detecting objects of different sizes, and its ability in feature extraction is improved. The spatial pyramid pooling (SPP) module is used in the backbone network so that the model can input any image aspect ratio and size. In terms of the activation function, the Hardswish activation function is adopted for the entire network model after the convolution operation. In terms of the loss function, *CIoU* is used as the loss function of detection box regression to solve the problems of low positioning accuracy and slow regression of the target detection box during training.

① 基金项目: 黔科合重大专项 (ZNWLQC[2019]3012-1); 黔科合支撑 ([2021] 一般 297)
收稿时间: 2022-01-24; 修改时间: 2022-02-22; 采用时间: 2022-03-11; csa 在线出版时间: 2022-09-14

Experimental results show that the improved detection model is tested on the KITTI 2D dataset, and the precision of target detection, the recall rate, and the mean average precision (mAP) are increased by 2.5%, 5.1%, and 2.3%, respectively.

Key words: target detection; YOLOv5; squeeze and excitation (SE); attention mechanism; convolutional block attention module (CBAM); activation function; Hardswish

自动驾驶能够有效避免因驾驶技能、心理变化、疲劳程度等人为因素而导致的交通事故, 能够有助于合理管控道路交通流量以改善道路的通行能力, 还能够提供舒适友好的驾乘体验, 具有广阔的应用前景以及潜在的社会效益[1]. 自动驾驶系统的核心可以分为 3 个部分: 感知、规划和控制, 其中目标检测是自动驾驶领域环境感知系统的一个重要分支. 在自动驾驶环境感知领域, 从相机拍摄的视频中检测前方目标是近年来的研究热点[2]. 传统检测方法主要是人工提取图像中的颜色、形状和纹理等特征, 然后通过支持向量机和 AdaBoost 等分类器识别[3]. 与传统的人工基于特征的提取方法相比, 深度学习算法提高了检测的速度和准确率. 因此, 使用 2D 图像的深度学习算法已经成为自动驾驶中道路目标检测的最有力工具之一[2].

随着深度学习的广泛应用, 目标检测的精确率和效率都得到了较大提升, 但基于深度学习的目标检测仍面临改进与优化主流目标检测算法的性能、提高小目标物体检测精度、实现多类别物体检测等关键技术的挑战[4]. 基于深度学习的检测方法可以分为 one-stage 和 two-stage 两大类. 以 Faster R-CNN[5] 为代表的 two-stage 方法首先通过 RPN 找出图片中待检测物体的候选区域, 再对候选区域的特征进行分类和目标框检测, 该类方法具有精度高和速度较慢的特点. 以 YOLO 系列[6–9] 为代表的 one-stage 目标检测算法, 通过归回方式预测出物体的位置和类别, 经过单次检测能直接输出检测结果, 该类方法具有检测速度快的特点, 被广泛应用于目标检测任务中.

在自动驾驶应用场景下, 将 YOLOv5 应用于目标检测中, 性能较之前版本有明显的提升, 但在高运行速度情况下检测精度仍不够高. 因此本文提出一种基于改进 YOLOv5 的车辆端目标检测方法.

1 YOLOv5 基本原理

它的结构由 4 部分组成, 包括输入端 (input)、主干网络 (backbone)、颈部网络 (neck) 和输出端 (head). YOLOv5 本整体的网络结构如图 1 所示, 网络组件结构如图 2 所示.

1.1 输入端 (Input)

YOLOv5 在输入端会对图像数据进行自适应图像填充和 Mosaic 数据增强. 输入端还集成了自适应锚框计算, 使得在训练不同数据集前, 模型通过对数据集的标签框进行聚类获得初始锚框大小, 无需人工设定锚框的参数.

Mosaic 是一种基于 Cutmix[10] 的数据增强方法. 在 Cutmix 中组合了两张图像, 而 Mosaic 将 4 张训练图像组合成 1 张进行训练. 作用是增强了对正常背景之外的对象的检测. 在训练时每个批量数据包含大量的图像, 使用 Mosai 后是原来批量所包含图像数量的 4 倍, 因此减少了估计均值和方差时需要大批量的要求.

自适应图片缩放操作仅在模型推理阶段执行, 首先根据原始图片大小与输入到网络图片大小计算缩放比例. 然后根据原始图片大小与缩放比例计算缩放后的图片大小. 最后缩放到适合模型输入大小的图片.

在 YOLOv2[7]、YOLOv3[8]、YOLOv4[9] 版本中, 其中通过聚类提取先验框尺度, 针对不同的数据集都需要设定特定长宽的锚点框. 训练不同的数据集时, 都是通过单独的程序运行来获得初始锚点框. YOLOv5 版本将自适应锚框计算功能嵌入到代码中, 每次训练时根据数据集的名称自适应的计算出最佳的锚点框.

1.2 主干网络 (Backbone)

主干网络能从图像中提取到特征, 在 YOLOv5 中主要使用了 C3 模块和 SPPF 模块. 其中使用 C3 模块能减少模型计算量和提高推理速度, 使用 SPPF 模块能对同一个特征图进行多尺度特征提取, 有利于提升模型的精度.

C3 模块其包含了 3 个标准卷积层以及多个瓶颈模块 (bottleneck), 其中瓶颈模块数量由配置文件. YAML 的 n 和 depth_multiple 参数乘积决定. 其结构如图 2 所示. C3 模块是对残差特征进行学习的主要模块, 其结构分为两个分支, 一分支使用了上述指定多个 Bottleneck 堆叠和 3 个标准卷积层, 另分一支仅经过一个基本卷

积模块, 最后将两支进行 Concat 操作. 其中在 Concat 操作后的标准卷积模块中的激活函数是 *SiLU*. 它是一种按元素应用 *Sigmoid* 线性单元的函数, 其特点是处处可导、连续光滑和并非一个单调的函数, 有助于表示非线性特征. *SiLU* 激活函数定义如下:

$$SiLU(x) = x \times Sigmoid(x) \tag{1}$$

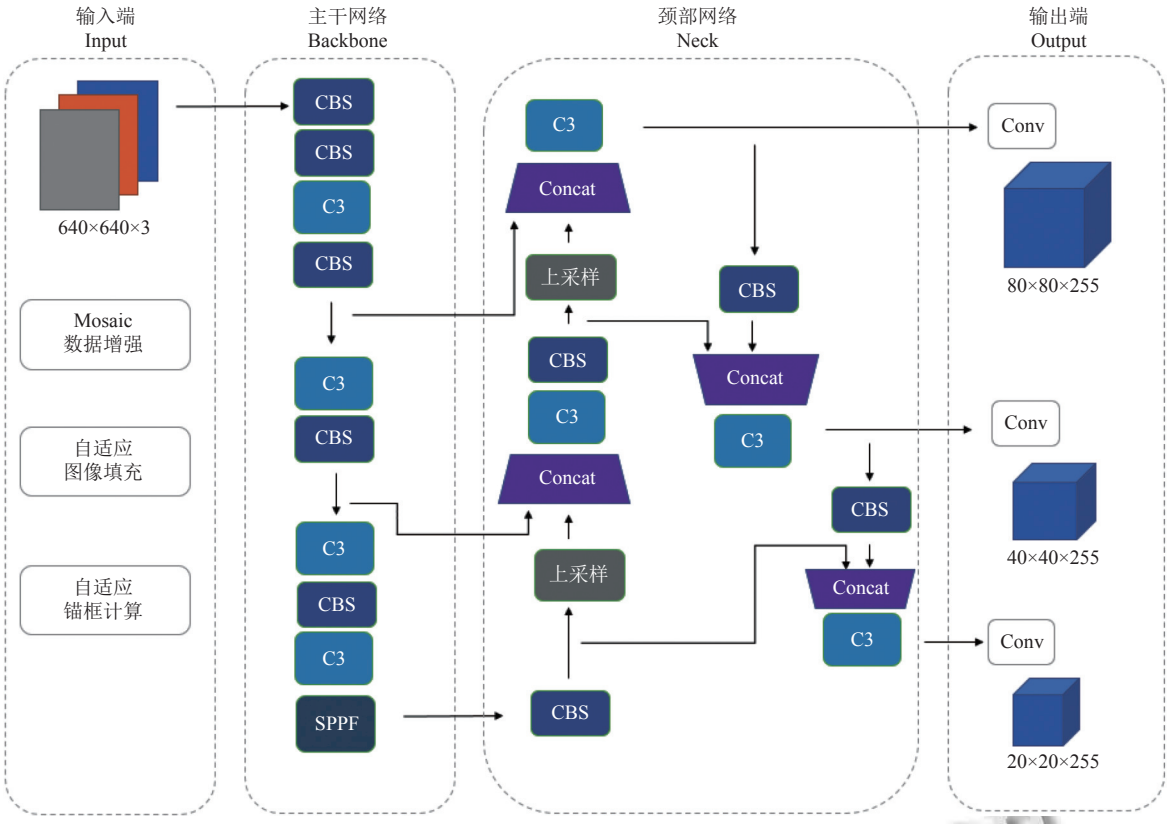

图 1 YOLOv5 网络结构

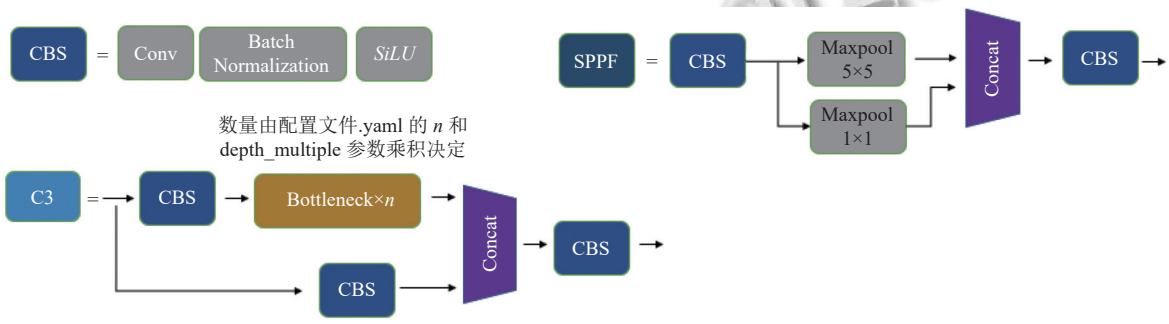

图 2 YOLOv5 网络组件结构

SPPF 模块原理和空间金字塔池化 (spatial pyramid pooling, SPP)[11] 基本一致, 但用到的池化核设计不一样. SPP 在 YOLOv5 中默认使用 4 个池化核, 分别是: 5×5、9×9、13×13 和 1×1. SPPF 在 YOLOv5 中默认使用两个池化核, 分别是: 5×5 和 1×1. 两者对比下 SPPF 在提取特征时速度会更快.

空间金字塔池化[11] 能融合不同尺度大小的特征图, 对任意尺寸的特征图直接进行固定尺寸的池化, 来得到固定数量的特征. 然后将每个池化得到的特征合起来即得到固定长度的特征个数, 接着输入到全连接层中进行训练网络. SPP 增加感受野, 解决了 CNN 输入图像大小必须固定的问题, 从而可以使得输入任意

图像的高宽比和大小.

1.3 颈部网络 (Neck)

由特征金字塔 (feature pyramid network , FPN) [12] 和路径聚合结构 (path aggregation network, PAN) [13] 组成. 其中 FPN 同时使用低层特征高分辨率和高层特征的语义信息, 在网络中自上而下传递语义信息. PAN 是自下而上传递定位信息, 使低层信息更容易传播到顶层. PAN 能融合不同尺寸特征图的特征信息, 有助于提升模型对不同形状大小物体的检测能力.

1.4 输出端 (Head)

输出端作为模型的检测部分, 主要是对提取到的多尺度特征图进行预测不同大小的物体. 输出端的锚框机制通过聚类提取先验框尺度, 并约束预测边框的位置.

模型输出 3 种尺度的张量, 第 1 种是对相对输入图像做了 8 倍下采样的输出, 其感受野较小, 保存了低层高分辨的特征, 对于检测小物体是很有帮助的. 第 2 种是相对输入图像做了 16 倍下采样的输出, 其感受野中等, 对于检测中等物体是有帮助的. 第 3 种是相对输入图像做了 32 倍下采样的输出, 其感受野较大等, 适合检测大物体.

1.5 损失函数 (Loss)

损失函数方面, 边框信息的回归损失计算采用了 *CIoU* 函数[14]. 它能同时考虑检测框和目标框重叠面积、边界框中心距离和边界框宽高比, 加速训练过程中目标检测框回归速度, 提高边界框的定位精度.

CIoU 公式如下:

$$CIoU = IoU - \frac{\rho^2(b,b^{gt})}{c^2} - av \tag{2}$$

其中, $\rho^2(b,b^{gt})$分别代表了预测框和真实框的中心点的欧式距离. c代表能够同时包含预测框和真实框的最小闭包区域的对角线距离. a是权重函数, v用来度量长宽比的相似性, a和v的公式如式 (3) 和式 (4):

$$\alpha = \frac{v}{1 - IoU + v} \tag{3}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{4}$$

CIoU 的损失公式如式 (5) 和式 (6):

$$Loss_{\mathrm{CIoU}} = 1 - CIoU \tag{5}$$

$$Loss_{\mathrm{CIoU}} = 1 - IoU + \frac{\rho^2(b,b^{gt})}{c^2} + av \tag{6}$$

2 改进 YOLOv5

改进后模型简称 YOLOv5-TR, 其网络结构如图 3 所示, 组件结构如图 4 所示. 在原始模型的主干网络添加压缩与激励模块 (squeeze and excitation, SE)[15], 筛选针对通道的特征信息, 提升特征表达能力.为了提升检测不同大小物体时的精度, 把卷积注意力模块 (convolutional block attention module, CBAM)[16,17] 与 Neck 部分融合, 使模型更加注意检测目标相关的特征, 提升模型特征提取能力. 在激活函数方面, 卷积操作后使用 Hardswish 激活函数[18], 应用于整个网络模型.

2.1 主干网络 (Backbone) 改进

在自动驾驶场景下进行目标检测, 由于复杂的环境会使模型学习到较多背景特征, 这不利于目标区域的特征学习, 进而影响目标检测的精度.

压缩与激励模块主要包含压缩 (squeeze) 和激励 (excitation) 两部分. SE 模块对输入的特征信息先经过压缩操作, 然后经过激励操作, 最终得到模块的输出.它能使模型更加关注目标区域的通道特征, 而抑制不重要的通道特征[15].

本文在主干网络中的 SPP 模块后面, 加入一个 SE 模块, 改进后的主干网络如图 3 所示. 原因是主干网络能提取图像大量的特征, 加入的 SE 模块能使模型更加关注目标区域的通道特征, 并能抑制不重要的通道特征, 提升模型对关键特征的提取能力, 从而提高目标检测的精度.

2.2 颈部网络 (Neck) 改进

在自动驾驶场景下, 检测的类别可分为“Car”“Van”“Truck”“Tram”“Pedestrian”“Person_sitting”“Cyclist”“Tram”[18], 类别之间的大小是有较大差异的. 同时由于相机成像模型, 导致距离相机近的物体在图像中显示较大, 距离相机远的物体在图像中显示较小. 检测的目标按形状大小可分为小目标、一般目标和大目标. 模型以高精度和高检测速度情况下, 同时对不同大小的目标进行检测是有困难的.

卷积注意力模块 CBAM 主要由两部分组成, 包括通道注意力模块 (channel attention module, CAM) 和空间注意力模块 (spatial attention module, SAM). CBAM 同时关注了空间信息和通道信息, 对网络中间的特征图进行重构, 使模型更加关注重要的特征, 提升模型的特征提取能力[16].

本文在颈部网络中融合两个 CBAM 模块, 改进后的颈部网络如图 3 所示, 同时对输入的特征图进行 CBAM

模块处理时, 先进行通道注意力操作, 然后进行空间注意力操作, 最终得到输出结果. 原因是特征图通过注意力机制处理后能获注意力信息, CBAM 模块能同时提取空间注意力信息和通道注意力信息. 将 CBAM 模块融合在颈部网络, 突出特征图中的重要信息, 经过后面的进一步特征提取, 预测输出不同大小特征图的目标检测结果. 使得模型在检测不同大小的物体时, 能更关注重要的特征和提升特征提取能力, 从而提升物体的检测精度.

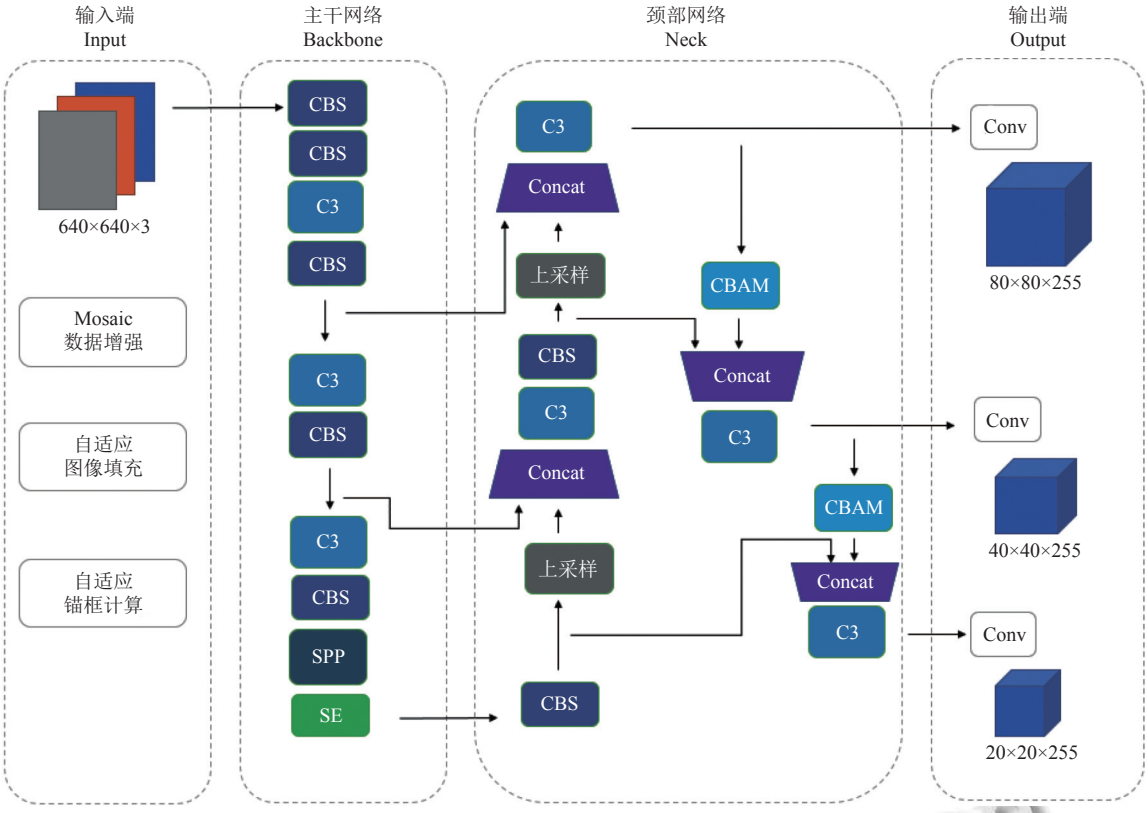

图 3 YOLOv5-TR 网络结构

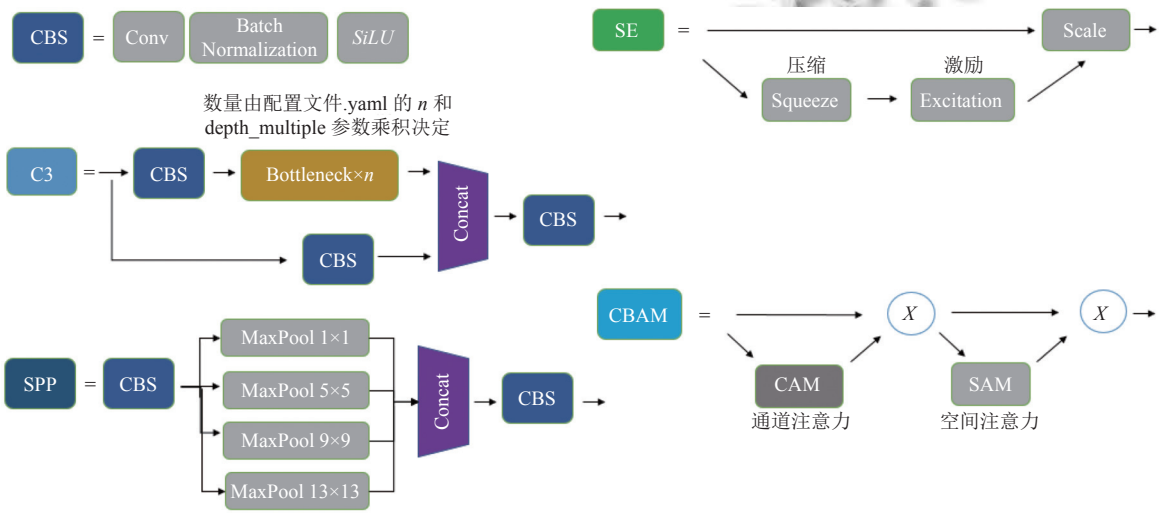

图 4 YOLOv5-TR 网络组件结构

2.3 激活函数改进

为了提高模型的表达能力, 同时考虑模型计算的速度, 选择合适的激活函数尤为重要.

Hardswish 在 MobileNetV3 架构[19] 中被提出, 具

有无上界、有下界、平滑和非单调等特点, 可使神经网络层具有更丰富的表达能力.

本文使用 Hardswish 函数替换 *SiLU* 函数. 原因是相较于 *SiLU* 函数, 它用分段线性模拟代替了计算成本高的 *Sigmoid* 处理, 具有数值稳定性好和计算速度快的优点, 同时使模型具有丰富的表达能力. 其中, Hardswish 激活函数公式定义如式 (7), 它分别考虑 3 种输入情况, x是输入值.

$$Hardwish(x)=\begin{cases}0, \text{ if } x\leqslant -3\\ x, \text{ if } x\geqslant +3\\ \dfrac{x(x+3)}{6}, \text{ otherwise}\end{cases} \tag{7}$$

3 实验分析

改进后的检测模型在 KITTI 2D[18] 目标检测数据集上测试, 其中评估指标包括精确率、平均精度均值、帧率和召回率. 在实验结果分析中还包括边界框回归损失和类别损失等指标. 最终使用不同改进策略后, 进行多组对比实验, 得到检测结果.

3.1 数据集

KITTI 2D 目标检测数据集, 采用左侧相机图像, 包含 7 481 张带有标签的图片, 其中图像是彩色并保存为 png 格式. 按照 7:2:1 的比例划分, 其中训练集 5 241 个样本, 验证集 1 500 个样本和测试集 750 个样本. 标签中包括 7 种类别, 分别是“Car”“Van”“Truck”“Tram”“Pedestrian”“Person_sitting”“Cyclist”“Tram”.

对训练数据进行数据增强, 包括图片 HSV 色域变换、随机位移 (translation)、大小变换 (scale)、随机左右翻转和 Mosaic 数据增强等, 具体参数如表 1. 通过数据增强增加训练样本的多样性, 提高模型的泛化能力.

表 1 数据增强参数

参数名称	大小
HSV-Hue	0.015
HSV-Saturation	0.7
HSV-Value	0.4
Translation	0.1
Scale	0.5
Flip left-right	0.5
Mosaic	1.0

3.2 实验环境

实验环境分为训练环境和测试环境. 训练环境使用 2 块 NVIDIA GeForce GTX 1080 8 GB 显存的 GPU, 内存是 32 GB, CPU 是 Intel(R) Core(TM) i7-6700K . 加载软件环境有 VSCode、Python 3.8、Cuda 11.4. 模型在 Linux 系统, 通过 PyTorch 深度学习框架搭建.

测试环境使用 1 块 NVIDIA GeForce GTX 1080 8 GB 显存的 GPU, 其余环境因素和训练环境一致.

3.3 训练模型

模型参数 depth_multiple 表示模型深度倍数, 设置其值为 0.33, 参数 width_multiple 表示模型宽度倍数, 设置其值为 0.50. 改进后的检测模型在 KITTI 2D 目标检测数据集上训练, 模型具体训练参数如表 2.

表 2 训练参数

参数名称	大小
输入尺寸	[640, 640]
批次大小	64
初始学习率	0.01
最终学习率	0.001
训练轮数	300
*IoU*训练阈值	0.50
优化器	SGD
优化器权重衰减	0.000 5
IoU Loss	*CIoU* Loss

3.4 评估指标

本文实验采用 4 个指标, 包括精确率 (Precision, *P*)、类别的平均精度均值 (mean average precision, *mAP*)、帧率 (frames per second, *FPS*) 和召回率 (recall, *R*).

计算公式如式 (8)–式 (11) 所示:

$$P=\frac{TP}{TP+FP} \tag{8}$$

$$mAP=\frac{1}{N}\sum_{i=1}^{N}AP_i \tag{9}$$

$$FPS=\frac{FrameNum}{ElapsedTime} \tag{10}$$

$$R=\frac{TP}{TP+FN} \tag{11}$$

其中, *TP* 为正样本预测正确的数量, *FN* 为负样本预测错误的数量, *FP* 为正样本预测错误的数量, *TN* 为负样本预测正确的数量. AP_i为第 i 类检测准确率, N 为类别数量. *FrameNum* 是指运行帧数, *ElapsedTime* 是指 1 s 时间内.

3.5 实验结果与分析

改进后的模型 YOLOv5-TR, 在主干网络中添加压缩与激励模块 (SE), 把卷积注意力模块 (CBAM) 与 Neck 部分融合, 同时进行卷积操作后使用 Hardswish 激活函数, 应用于整个网络模型.

模型训练 300 轮, 并在验证集上测试, 观察改进后的模块和原模型的边界框回归损失 (box_loss)、目标

置信度损失 (obj_loss) 和类别损失 (cls_loss) 的变化, 具体如图 5 所示.

如图 5 所示, 改进后的模型 YOLOv5-TR 和 YOLOv5 相比较下, 其边界框回归损失和目标置信度损失更低, 模型收敛速度更快. YOLOv5-TR 模型在迭代 300 轮后, 边界框回归损失曲线渐趋平缓, 最后达到 0.023 8 左右不再降低. 目标置信度损失达到 0.035 1 左右不再降低. 类别损失达到 0.002 1 左右不再降低.

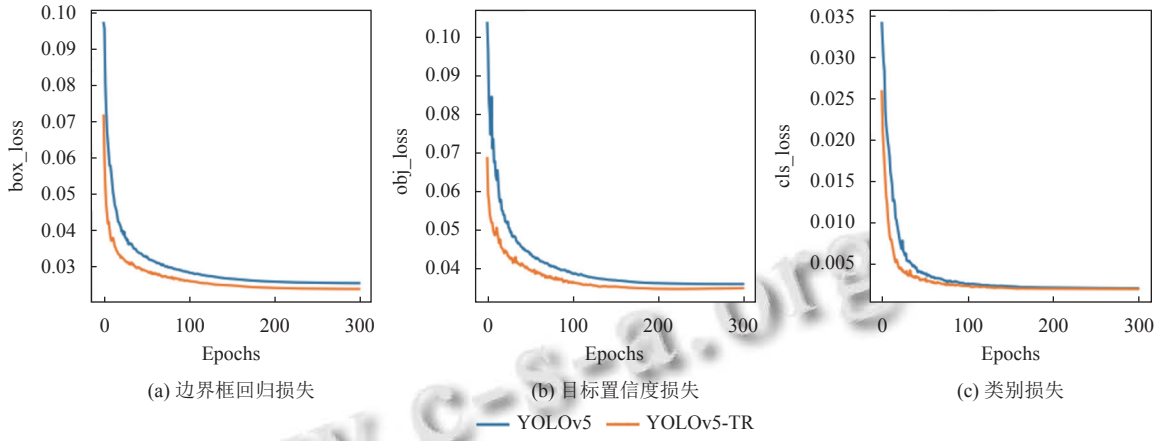

图 5 损失变化图

如图 6 所示, 在验证集上测试, 观察改进后的模块和原模型的精确率 (*P*)、召回率 (*R*) 和当 *IoU* 为 0.5 时平均精度均值 (*mAP*) 的变化, 具体数据如表 3. YOLOv5-TR 的整体性能比 YOLOv5 高, 在模型迭代 300 轮后, 精确率达到 0.945, 召回率达到 0.875, 平均精度值到达 0.929.

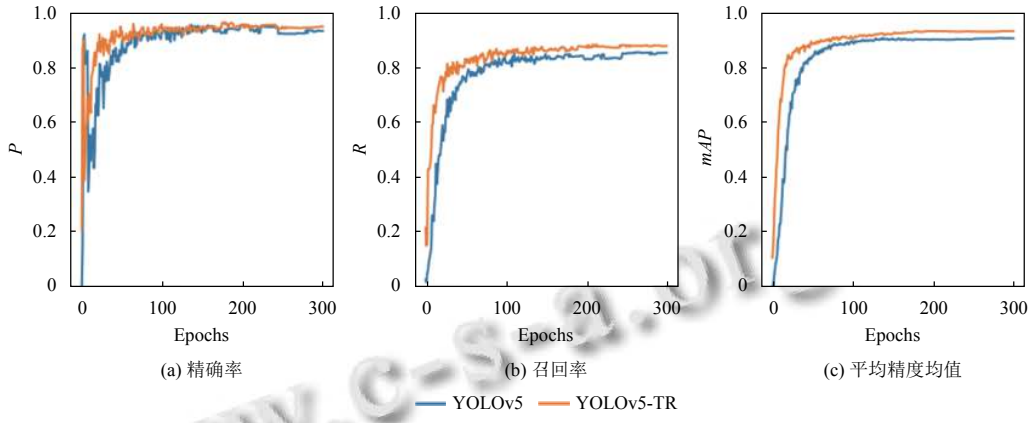

图 6 在验证集上各指标变化的对比图

表 3 在验证集上 YOLOv5-TR 和 YOLOv5 性能指标对比

评估指标	YOLOv5-TR	YOLOv5
P	**0.945**	0.931
R	**0.875**	0.846
mAP@0.5	**0.929**	0.904
FPS	277	**285**

表 4 在测试集上 YOLOv5-TR 和 YOLOv5 性能指标对比

评估指标	YOLOv5-TR	YOLOV5
P	**0.939**	0.914
R	**0.892**	0.841
mAP@0.5	**0.928**	0.905
FPS	277	**285**

在测试集上测试, 观察改进后的模块和原模型的评估指标, 具体数据如表 4. YOLOv5-TR 的整体性能比 YOLOv5 高, 在模型迭代 300 轮后, 精确率达到 0.939, 提高了 2.5%. 召回率达到 0.875, 提高了 5.1%. 当 *IoU* 为 0.5 时的平均精度值到达 0.929, 提高了 2.3%.

在测试集上测试, 观察不同改进带来的模型指标提升结果. 其中改进内容包括压缩与激励模块 (SE)、卷积注意力模块 (CBAM) 和激活函数 (Hardswish). 从表 5 可以观察到, 对精确率提升起到关键作用的是压缩与激励模块, 让精确率提升了 2.2%. 对召回率提升

起到关键作用的是压缩与激励模块和卷积注意力模块, 分别让召回率提升了 2.7% 和 2.2%. 对平均精度均值提升起到关键作用的是压缩与激励模块和卷积注意力模块., 分别让平均精度均值提升了 0.8% 和 1.1%.

表 5 在测试集上不同改进带来的模型指标变化

名称	*P*	*R*	*mAP*@0.5
YOLOv5	0.914	0.841	0.905
YOLOv5+Hardswish	0.916	0.843	0.909
YOLOv5+Hardswish+SE	0.938	0.870	0.917
YOLOv5+Hardswish+SE+CBAM	**0.939**	**0.892**	**0.928**

YOLOv5-TR 在 KITTI 2D 数据集中的 7 种类型的测试数据如表 6.

表 6 在测试集上 YOLOv5-TR 性能指标

类别	*P*	*R*	*mAP*@0.5
All	0.939	0.892	0.928
Car	0.957	0.945	0.976
Van	0.958	0.956	0.980
Truck	0.958	0.971	0.982
Pedestrian	0.915	0.803	0.874
Person_sitting	0.987	0.778	0.821
Cyclist	0.901	0.845	0.909
Tram	0.897	0.949	0.954

4 结论与展望

本文提出一种基于改进 YOLOv5 的车辆端目标检测方法, 在主干网络添加 SE 模块, 筛选针对通道的特征信息, 提升特征表达能力. 为了提升检测不同大小物体时的精度, 将注意力机制与检测网络融合, 把卷积注意力模块 CBAM 与 Neck 部分融合, 使模型在检测不同大小的物体时, 能更关注重要的特征, 和提升特征提取能力. 在整个网络模型中的卷积操作后使用 Hardswish 激活函数. 实验结果表明, 目标检测的精确率、召回率和平均精度值均得到了提升.

参考文献

1 章军辉, 陈大鹏, 李庆. 自动驾驶技术研究现状及发展趋势. 科学技术与工程, 2020, 20(9): 3394–3403. [doi: 10.3969/j.issn.1671-1815.2020.09.005]

2 Fan JQ, Huo TJ, Li X. A review of one-stage detection algorithms in autonomous driving. Proceedings of the 2020 4th CAA International Conference on Vehicular Control and Intelligence (CVCI). Hangzhou: IEEE, 2020. 210–214.

3 肖雨晴, 杨慧敏. 目标检测算法在交通场景中应用综述. 计算机工程与应用, 2021, 57(6): 30–41. [doi: 10.3778/j.issn.1002-8331.2011-0361]

4 赵永强, 饶元, 董世鹏, 等. 深度学习目标检测方法综述. 中国图象图形学报, 2020, 25(4): 629–654. [doi: 10.11834/jig.190307]

5 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. [doi: 10.1109/TPAMI.2016.2577031]

6 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: IEEE, 2016. 779–788.

7 Redmon J, Farhadi A. YOLO9000: Better, faster, stronger. Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 6517–6525.

8 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv:1804.02767v1, 2018.

9 Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv:2004.10934, 2020.

10 Yun S, Han D, Chun S, *et al*. CutMix: Regularization strategy to train strong classifiers with localizable features. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul: IEEE, 2019. 6022–6031.

11 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, 2015, 37(9): 1904–1916. [doi: 10.1109/TPAMI.2015.2389824]

12 Lin TY, Dollár P, Girshick R, *et al*. Feature pyramid networks for object detection. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu: IEEE, 2017. 936–944.

13 Li HC, Xiong PF, An J, *et al*. Pyramid attention network for semantic segmentation. British Machine Vision Conference 2018. Newcastle: BMVA Press, 2018.

14 Zheng ZH, Wang P, Liu W, *et al*. Distance-IoU Loss: Faster and better learning for bounding box regression. Proceedings of the 34th AAAI Conference on Artificial Intelligence. Palo Alto: AAAI Press, 2020.

15 Hu J, Shen L, Albanie S, *et al*. Squeeze-and-excitation networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(8): 2001–2023.

16 Woo S, Park J, Lee JY, *et al*. CBAM: Convolutional block attention module. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018. 3–19.

17 李永上, 马荣贵, 张美月. 改进 YOLOv5s+DeepSORT 的监控视频车流量统计. 计算机工程与应用, 2022, 58(5): 271–279. [doi: 10.3778/j.issn.1002-8331.2108-0346]

18 Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite. Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Providence: IEEE, 2012. 3354–3361.

19 Howard A, Sandler M, Chen B, *et al*. Searching for MobileNetV3. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul: IEEE, 2020. 1314–1324.

(校对责编: 牛欣悦)