面向路侧交通监控场景的轻量车辆检测模型

郭宇阳[1],胡伟超[2,4],戴 帅[3],陈艳艳[4]
1.中国人民公安大学 交通管理学院,北京 100038
2.公安部道路交通安全研究中心 科研管理组,北京 100062
3.公安部道路交通安全研究中心 交通政策规划研究室,北京 100062
4.北京工业大学 城市交通学院,北京 100124

摘 要:针对路侧交通监控场景和智能交通管控需要,提出轻量型的车辆检测算法GS-YOLO,解决现有模型检测速度慢、占用内存多的问题。GS-YOLO借鉴GhostNet思想将传统卷积分为两步,利用轻量操作增强特征,降低模型的计算量。在主干特征提取网络中引入注意力机制,对重要信息进行选择,提高模块的检测能力。另外参考SqueezeNet结构,使用Fire Module和深度可分离卷积减少模型参数,模型大小从244 MB降低到34 MB,内存占用降低了86%。使用Roofline模型对实验数据和模型实际性能进行分析,结果表明GS-YOLO的精确度(AP)达到85.55%,相比YOLOv4提升了约0.45%。由于受计算平台带宽影响,GS-YOLO在GPU上检测速度提升7.3%,但在CPU上检测速度提高了83%,更适用于算力资源不足的小型设备。

关键词:图像处理;目标检测;轻量化;GhostNet;深度可分离卷积
文献标志码:A **中图分类号:**TP391.4 **doi:**10.3778/j.issn.1002-8331.2109-0516

Lightweight Vehicle Detection Model for Roadside Traffic Monitoring Scenarios

GUO Yuyang[1], HU Weichao[2,4], DAI Shuai[3], CHEN Yanyan[4]
1.School of Traffic Management, People's Public Security University of China, Beijing 100038, China
2.Scientific Research Management Department, Road Traffic Safety Research Center of the Ministry of Public Security, Beijing 100062, China
3.Transportation Policy Planning Research Office, Road Traffic Safety Research Center of the Ministry of Public Security, Beijing 100062, China
4.School of Metropolitan Transportation, Beijing University of Technology, Beijing 100124, China

Abstract: To satisfy the needs of roadside traffic monitoring scenarios and intelligent traffic control, a lightweight vehicle detection model GS-YOLO is proposed to solve the problems of low detection efficiency and high memory consumption of existing models. Following the structure of GhostNet, GS-YOLO divides the vanilla convolution into two steps and uses cheap operations to enhance features and reduces the resource consumption of the model. An attention mechanism is introduced in the backbone feature extraction network to select important information and improve the detection capability of GS-YOLO. The model parameters are reduced by using Fire Module and depthwise separable convolution with reference to the SqueezeNet, and the model size is reduced from 244 MB to 34 MB, with an 86% reduction in memory usage. Using Roofline theory to analyze the experimental data and the actual performance of the model. The experimental results show that the accuracy(AP) of GS-YOLO reaches 85.55%, which is about 0.45% higher than that of YOLOv4. Due to the bandwidth impact of computing platform, GS-YOLO has a 7.3% improvement in image processing speed on GPU and 83% on CPU, which is more suitable for small devices with insufficient computing power resources.

Key words: image processing; object detection; lightweight; GhostNet; depthwise separable convolution

基金项目:国家重点研发计划(2020YFB1600304);公安部技术研究计划(2019JSYJB07);中央级公益性科研院所基本科研业务费专项(11104100000180001201101)。

作者简介:郭宇阳(1998—),女,硕士研究生,研究方向为计算机视觉、图像处理,E-mail:gyy1067181224@163.com;胡伟超(1987—),男,博士研究生,副研究员,研究方向为智能交通技术。

收稿日期:2021-09-30 **修回日期:**2021-11-10 **文章编号:**1002-8331(2022)06-0192-08

目标检测技术已经在人脸识别、车辆检测等多个领域内广泛应用。基于深度学习的目标检测算法主要分为一阶目标检测与两阶目标检测，前者采用端到端的方式直接获得物体位置并进行类别预测，计算速度快，但精度略有损失，典型算法有SSD[1]、RetinaNet[2]和YOLO[3-5]系列。两阶目标检测算法相比于一阶网络可以更充分地提取特征，得到目标物体的精准位置和分类，代表算法有Fast R-CNN[6]、Cascade R-CNN[7]。此外CornerNet[8]和CenterNet[9]这些依靠关键点来检测目标的算法也受到了研究人员和业界的广泛关注。

基于深度学习的目标检测算法在车辆检测领域的应用一直是研究的热点。曹诗雨等[10]使用Faster-RCNN网络检测车辆，用卷积特征代替人工提取特征，避免了传统检测问题中设计手工特征的问题。李松江等[11]使用Cascade RCNN算法进行目标检测，引入空洞卷积来减少下采样过程中的特征丢失，可以有效地检测出小目标和遮挡目标，改进后的网络准确率有所提高，但在速度方面略有损失。杜金航等[12]将Darknet53改进为30个卷积层的卷积神经网络，使用 K-means聚类选取车辆锚框，可以实现道路车辆的实时检测。金旺等[13]使用深度残差网络作为主干网络，利用软化非极大抑制解决车辆尺度变化大以及遮挡问题。顾恭等[14]提出的Vehicle-YOLO模型是在YOLOv3的基础上采用7次深度残网络输出五种不同大小的特征图，对潜在车辆的边界框进行提取，从而提升车辆检测的精度和普适性。鲁博等[15]将YOLOv3-tiny的主干网络与BiFPN特征金字塔结构相结合，并提出了一种新的上采样结构，来解决采样过程中信息丢失的问题。李宇昕等[16]和李汉冰等[17]分别使用残差网络和反残差网络作为基础模型，均使用Focal loss改进损失函数，平衡正负样本，提高了目标检测的准确度。

目前，车辆检测多用于智能网联和自动驾驶汽车感知交通环境信息，大量研究使用KITTI[18]数据集完成模型训练与测试。通常这类数据集基于车载设备、从平视或小倾角俯视角度采集，与交通管理场景下的路侧监控图像视角存在明显差异，相关模型不能很好适应大倾角俯视视角下的车辆检测任务。本文在UA-DETRAC[19]数据集基础上，针对交通流检测场景和边缘计算需求，提出一种基于YOLOv4[20]改进的轻量型车辆检测算法GS-YOLO。使用GhostNet结构思想，利用轻量操作降低通道，增强特征，在主干网络中增加注意力机制，筛选重要特征，提高模型的检测能力。由于多尺度预测部分卷积操作内存占用多，计算复杂，借鉴SqueezeNet[21]的Fire Module结构和深度可分离卷积，对加强特征提取网络和预测网络进行优化，有效地解决原有模型内存占用大和特征提取不充分的问题。

1　GhostNet网络

深度卷积神经网络会产生许多相似的特征图，这些特征图通常会被当作冗余信息，一些轻量化网络就是利用特征冗余性，通过裁剪一部分的冗余特征使得模型轻量化。而Han等[22]则认为卷积网络强大的提取功能与这些冗余特征图成正相关，并提出GhostNet算法，将少量传统卷积计算和轻量冗余特征生成相结合，在保证检测精度的情况下，减少模型参数量。

1.1　Ghost卷积和Ghost Bottleneck模块

Ghost模块的卷积操作将传统卷积分为两部分：生成少量特征图的传统卷积层和利用廉价的线性运算生成冗余特征图的轻量级线性变化层。Ghost模块使用更少的参数来生成冗余特征。

传统卷积和Ghost卷积的对比如图1所示。Ghost卷积过程第一部分利用传统卷积生成少量固有特征图，此处的卷积操作可以自定义卷积核大小，使用1×1卷积，也可以是3×3和5×5的卷积。第二部分使用轻量操作和恒等变换对第一部分生成的特征图进行线性操作处理，从而增加通道和扩充特征。轻量操作既可以是深度卷积，也可以是分组卷积等其他方式的卷积，以低成本的方式，保留冗余信息。最后将此部分的特征图和第一部分的固有特征图进行恒等变换拼接在一起，作为Ghost模块的输出特征图。

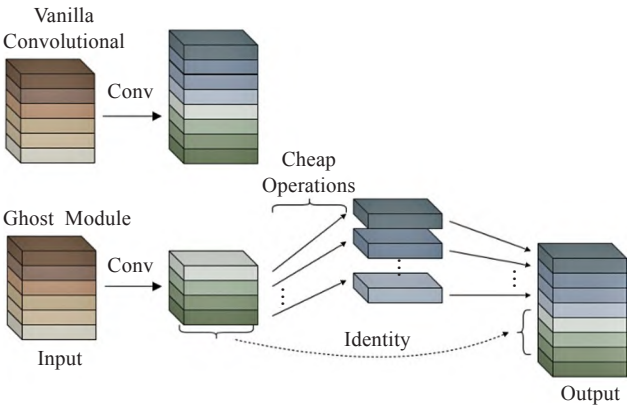


图1　传统卷积和Ghost卷积

Fig.1　Vanilla convolution and Ghost module

假设输入特征图 $H\times W$，输入通道数为 C_{in}，输出特征图尺寸为 $H'\times W'$，输出通道数为 $m\times radio$，卷积核尺寸为 $k\times k$，那么传统卷积的计算量为：

$$(m\times radio)\times H'\times W'\times C_{\text{in}}\times k\times k \tag{1}$$

为了使Ghost模块输出的特征图数量和传统卷积保持一致，Ghost模块中卷积操作的卷积核尺寸、步长和padding需要和传统卷积相同。假设Ghost模块的第一部分卷积生成 m 个特征图，每个特征图通过映射生成 $(radio-1)$ 个新特征图，则第二部分共获得 $m\times(radio-1)$ 个特征图。

Ghost模块第一部分的计算量为：

$$m\times H'\times W'\times C_{in}\times k\times k \tag{2}$$

Ghost模块第二部分的计算量为：

$$m\times(radio-1)\times H'\times W'\times k\times k \tag{3}$$

Ghost模块总计算量为：

$$m\times H'\times W'\times C_{in}\times k\times k+m\times(radio-1)\times H'\times W'\times k\times k \tag{4}$$

输出相同通道数和特征图尺寸，传统卷积和Ghost模块的计算量之比为：

$$\frac{(m\times radio)\times H'\times W'\times C_{in}\times k\times k}{m\times H'\times W'\times C_{in}\times k\times k+m\times(radio-1)\times H'\times W'\times k\times k}=\frac{C_{in}\times radio}{C_{in}+radio-1} \tag{5}$$

一般情况下 $C_{in}\gg radio$，则：

$$\frac{C_{in}\times radio}{C_{in}+radio-1}\approx radio \tag{6}$$

经过上述计算可以看出使用Ghost模块进行卷积操作可以在一定程度上降低卷积过程的计算量。

和ResNet的基本残差块(basic residual block)结构相类似，GhostNet Bottleneck的架构集成了多个卷积层和shortcut。每个GhostNet Bottleneck由两个Ghost模块堆叠而成。第一个模块作为扩展层，扩充通道数，第二个模块用于减少特征的通道数，和shortcut路径相匹配。和MobileNetV2的结构类似，在第一个Ghost模块后面加入批量归一化操作和ReLU激活函数，第二个模块只添加批量归一化操作。根据步长不同，GhostNet Bottleneck分为两类，如图2所示。其中步长为2的GhostNet Bottleneck主要用于压缩特征层尺寸，需在两个Ghost模块之间加入深度卷积完成下采样操作，同时在残差边添加步长为2的深度卷积和1×1的传统卷积。

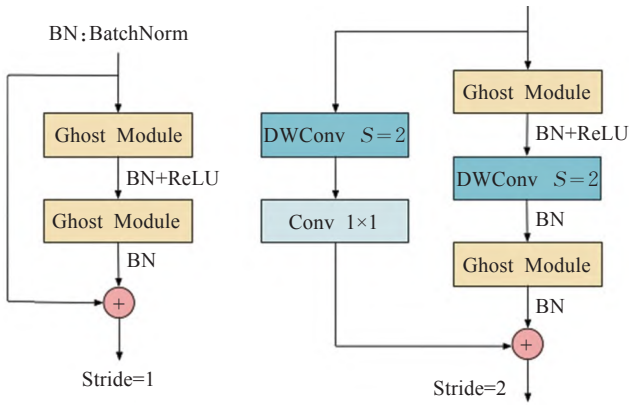


图2　GhostNet Bottleneck结构示意图

Fig.2　GhostNet Bottleneck structure

1.2　GhostNet

GhostNet网络架构的搭建灵感源自MobileNetV3基本体系结构，使用GhostNet Bottleneck代替MobileNetV3中的bottleneck结构，并将Hard-Swish激活函数更改为ReLU，各层网络结构参数如表1所示。

表1　GhostNet结构参数

Table 1　Structure parameters of GhostNet

输入	操作	输出通道数	SE	stride
416×416×3	Conv 3×3	16	×	2
208×208×16	Ghost Bottleneck	16	×	1
208×208×16	Ghost Bottleneck	24	×	2
104×104×24	Ghost Bottleneck	24	×	1
104×104×24	Ghost Bottleneck	40	√	2
52×52×40	Ghost Bottleneck	40	√	1
52×52×40	Ghost Bottleneck	80	×	2
26×26×80	Ghost Bottleneck	80	×	1
26×26×80	Ghost Bottleneck	80	×	1
26×26×80	Ghost Bottleneck	80	×	1
26×26×80	Ghost Bottleneck	112	√	1
26×26×112	Ghost Bottleneck	112	√	1
26×26×112	Ghost Bottleneck	160	√	2
13×13×160	Ghost Bottleneck	160	×	1
13×13×160	Ghost Bottleneck	160	√	1
13×13×160	Ghost Bottleneck	160	×	1
13×13×160	Ghost Bottleneck	160	√	1
13×13×160	Ghost Bottleneck	160	×	1
13×13×160	Conv 1×1	960	×	1
13×13×960	AvgPool 7×7	—	×	×
1×1×960	Conv 1×1	1 280	×	1
1×1×1 280	FC	1 000	×	×

GhostNet的第一层是卷积核为3×3、步长为2的传统卷积层，将分辨率为416×416的图片转换为通道数为16、大小为208×208的特征图，再通过一系列的GhostNet Bottleneck逐渐压缩特征图尺寸，扩大通道数。根据输入特征图的大小，可以将GhostNet Bottleneck分为6个阶段，在前5个阶段中除了每个阶段的最后一个GhostNet Bottleneck的步长为2，其余各部分GhostNet Bottleneck的步长均为1。此外，在部分的GhostNet Bottleneck中添加了注意力机制squeeze-and-excitation模块，根据特征重要程度来增强或削减特征图的权重，如图3所示。

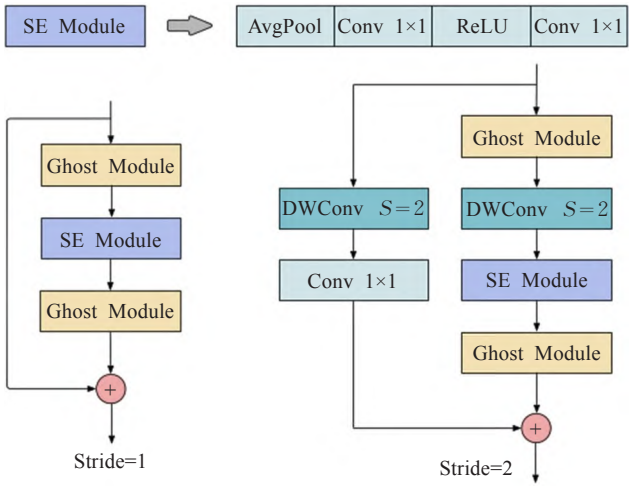


图3　包含SE模块的Bottleneck

Fig.3　Bottleneck with squeeze-and-excitation module

经过初步特征提取获得三个不同尺度的有效特征层，像素大小分别为52×52、26×26和13×13。

2　YOLOv4模型改进

2.1　YOLOv4简介

YOLOv4主要有主干特征提取网络、加强特征提取网络和预测网络三部分组成。主干特征提取网络使用CSPDarknet53提取输入数据的特征，CSPDarknet53由五个残差块构成，每个残差块分别包含了1、2、8、8、4个残差单元，并在Darknet53的基础上融入了CSPnet结构，缓解了网络深化造成的梯度消失的问题。

加强特征提取网络由SPPNet和PANet两部分构成，将主干特征提取网络得到的三个初步特征层进行特征融合，从而获得更有效的三个特征层。当主干特征提取网络的最后一个特征层完成三次卷积操作后，SPPNet对最后一个特征层进行四个不同尺度的最大池化操作，来增加感受野，分离出显著的上下文特征。随后将SPPNet加强特征提取得到的特征层与主干网络得到的另两个特征层送入PANet，执行自下而上和自上而下的双向特征融合，实现特征的反复提取。

预测网络利用从PANet获得的多尺度特征进行回归和分类，最终输出的维度包含样本类别值、预测框在x轴和y轴的偏移量、预测框的高度和宽度和置信度。YOLOv4的整体结构如图4所示。

2.2　深度可分离卷积

由于加强特征提取网络和预测网络存在较多的卷积过程，计算量较大，为了进一步减少模型参数，本文在此处引入深度可分离卷积操作。

深度可分离卷积可以将传统卷积分解为逐通道卷积和逐点卷积，逐通道卷积中的各个通道相互独立，缺少通道间的特征融合，导致输入输出的通道数相同，因此后续还需连接一个逐点卷积，用于改变通道数和融合各通道间的特征。

对于通道数为C_{in}、大小为$H\times W$的输入特征图，使用3×3的卷积核进行点乘求和，得到通道数为C_{out}、大小为$H\times W$的输出特征图。标准卷积的计算量为：

$$F_{\text{s}}=C_{\text{in}}\times 3\times 3\times H\times W\times C_{\text{out}} \tag{7}$$

深度可分离卷积是由3×3的逐通道卷积和1×1的逐点卷积构成，逐通道卷积的计算量为：

$$F_{\text{dw}}=C_{\text{in}}\times 3\times 3\times H\times W \tag{8}$$

逐点卷积的计算量为：

$$F_{\text{pw}}=C_{\text{in}}\times 1\times 1\times H\times W\times C_{\text{out}} \tag{9}$$

综合两步，可以得到深度可分离卷积与标准卷积的计算量之比：

$$r=\frac{F_{\text{dw}}+F_{\text{pw}}}{F_{\text{s}}}=\frac{C_{\text{in}}\times 3\times 3\times H\times W+C_{\text{in}}\times 1\times 1\times H\times W\times C_{\text{out}}}{C_{\text{in}}\times 3\times 3\times H\times W\times C_{\text{out}}}=\frac{1}{C_{\text{out}}}+\frac{1}{9}\approx\frac{1}{9} \tag{10}$$

由上式可知，深度可分离卷积虽然将一步卷积扩展为两步，但凭借其轻量化的卷积方式，总体计算量约为传统卷积的1/9，极大程度地降低了卷积过程的计算量。

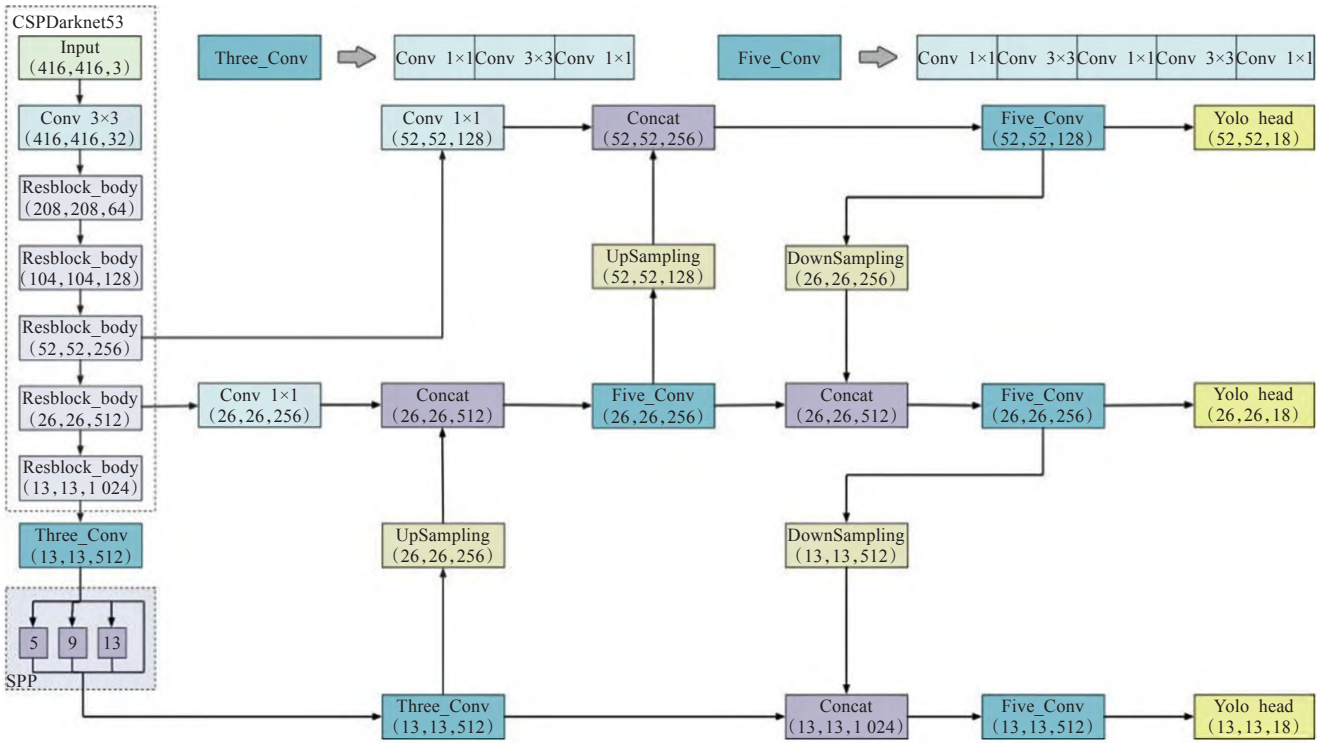


图4　YOLOv4网络结构

Fig.4　YOLOv4 network structure

2.3 Fire Module

Fire Module的结构如图5所示。输入特征图依次经过Squeeze层和Expand层,然后进行融合处理,将两个特征图进行通道拼接,作为最终输出。Squeeze层使用1×1卷积进行降维,特征图的尺寸保持不变,输出通道数为 s 。在Expand层,并行地使用1×1卷积和3×3卷积获得不同感受野的特征图,达到扩展的目的。

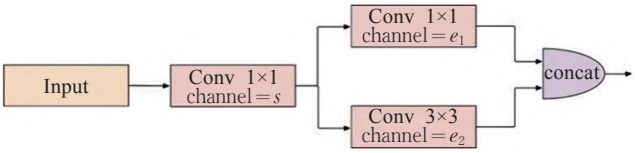


图5　Fire Module结构

Fig.5　Fire Module

主干特征提取网络输出三个不同尺度的特征图,其中尺寸为13×13的特征图需要先通过SPPNet结构,再传入PANet网络中。从SPPNet输出的特征图通道数为2 048,对其进行卷积操作会产生大量的参数,并且增加计算时间。公式(11)是使用Fire Module之后的参数量,其中 C_{in} 是输入特征图的通道数。

$$1\times1\times C_{in}\times s+1\times1\times s\times e_1+3\times3\times s\times e_2 \tag{11}$$

Fire Module具有三个可调参数: s、e_1、e_2 ,即Squeeze层和Expand层的输出特征图的通道数,通过设置三个参数可以有效地降低模型参数量。在GS-YOLO中Fire Module的三个参数设置为: $s=\frac{1}{8}C_{in}$, $e_1=e_2=\frac{1}{2}C_{in}$,使其在进行3×3卷积操作前将通道数压缩为原来的1/8。

2.4 GS-YOLO

本文在YOLOv4的基础上提出GS-YOLO车辆检测网络,其整体结构如图6所示。GS-YOLO在保证检测精度基本不变的情况下,一定程度上简化了网络结构和计算量,具体改进如下:

(1)主干特征网络选用GhostNet对输入图像进行初步特征提取,其中Ghost Module采用卷积核为1×1的传统卷积和深度卷积的轻量操作。通过SE模块对融合的特征进行过滤,增强重要特征权重,抑制无用特征,得到52×52、26×26和13×13的输出特征图。

(2)在多尺度检测部分,使用深度可分离卷积代替加强特征提取网络和预测网络中的传统卷积,使参数量大量减少,具体改动涉及:PANet网络中五次卷积块的3×3卷积、下采样过程中使用的3×3卷积以及预测网络Yolo-head中的3×3卷积。

(3)Fire Module可以在不过多增加计算开销的同时,加深网络深度,在SPPNet结构的输入输出部分引入Fire Module,可在一定程度上提高检测速度和准确度。

3 实验及分析

3.1 实验数据和平台

本文采用UA-DETRAC车辆数据集,该数据集拍摄于北京和天津的24个过街天桥,以俯视角度拍摄共100个交通场景,并标注了8 250个车辆和121万个目标对象。实验在全部场景中随机抽取80%作为训练集,剩余20%的样本作为测试集。

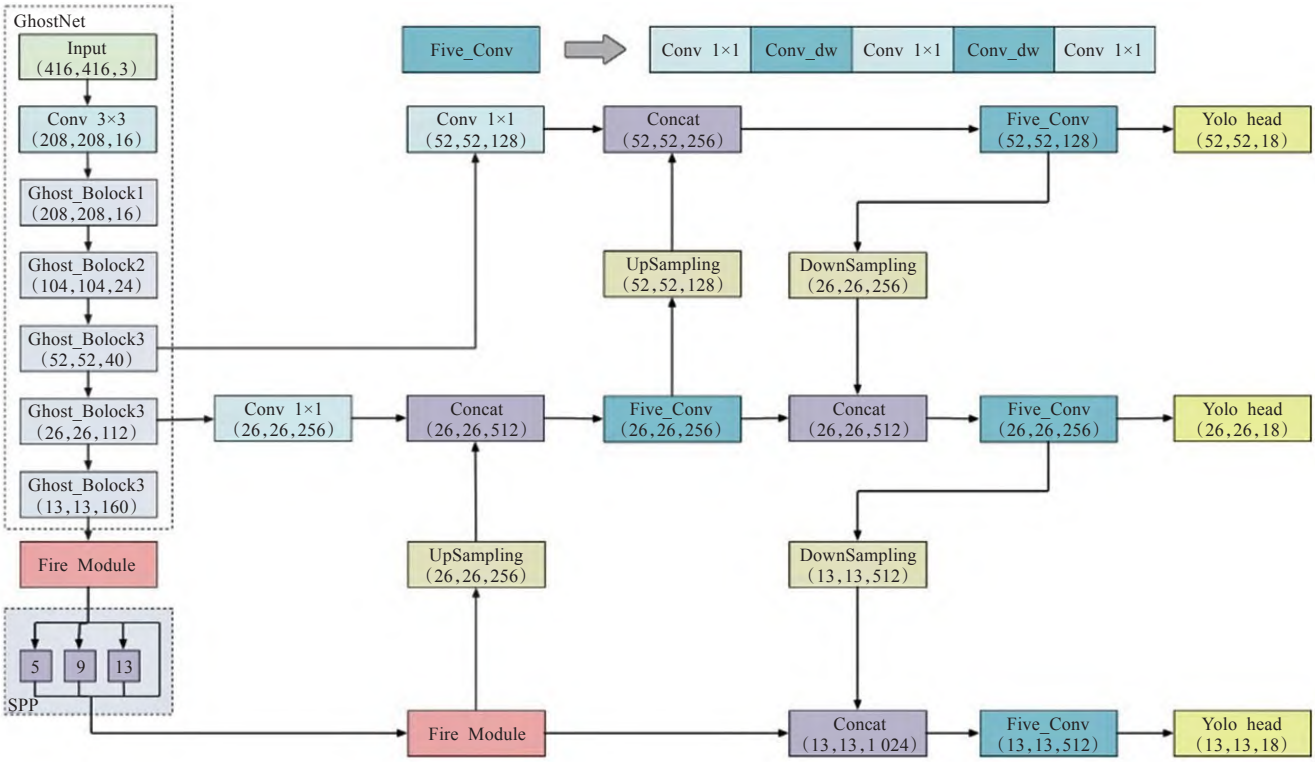


图6　GS-YOLO网络结构

Fig.6　GS-YOLO network structure

本实验在Windows10操作系统下进行,使用Python3.7进行编译和测试,对应开发工具为PyCharm2019.3.4,CPU为Intel Xeon Silver 4126,GPU为NVIDIA TITAN RTX。实验参数设置如表2所示,其中初始学习率为0.001,并在50 epoch后降低为0.000 1。

表2 实验参数选取

Table 2 Selection of experimental parameters

迭代次数	批量	动量	权重衰减	指数平滑	学习率
150	16	0.9	0.000 5	0.01	0.001/0.000 1

3.2 模型性能评价指标

模型的实际表现性能受到模型本身性能和计算平台性能的共同影响,如图7所示。计算平台的计算能力与算力和带宽有关,算力是指单位时间内平台倾尽全力能完成的浮点运算数,代表计算平台的性能上限,单位是FLOP/s;带宽是指单位时间内计算平台倾尽全力所能完成的内存交换量,代表计算平台的带宽上限,单位是Byte/s。

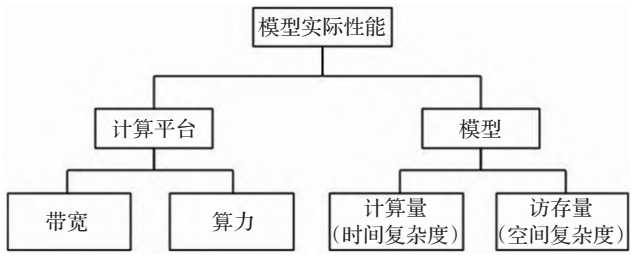


图7 模型性能影响因素

Fig.7 Factors of model performance

计算强度上限 $I_{\max}$ 是算力 π 和带宽 β 的比值,单位是FLOPs/Byte,表示的是单位内存交换最多可以完成多少次浮点运算,具体关系为:

$$I_{\max}=\frac{\pi}{\beta} \tag{12}$$

影响模型计算能力的两个指标是模型计算量和访存量。

计算量指的是模型的浮点运算次数,使用FLOPs衡量,即时间复杂度。时间复杂度影响模型的训练和预测时间,模型越复杂,模型训练和预测时间耗费时间越多,也就无法快速地预测和验证想法。整个卷积神经网络的时间复杂度表示为:

$$Time\sim O\left(\sum_{l=1}^{D}M_l^2\cdot K_l^2\cdot C_{l-1}\cdot C_l\right) \tag{13}$$

其中,D 是神经网络的卷积层数,C_{l-1} 和 C_l 指的是第 l 个卷积层的输入和输出通道数,M 是输出特征图边长,K 是卷积核边长。

访存量指的是单个样本在模型中进行一次前向传播发生的内存交换总量,即空间复杂度,访存量由模型每层参数权重的内存占用和输出特征图的内存占用构成。空间复杂度决定模型的参数量,模型参数越多,需要的训练数据量越大,由于实际生活中数据集通常不大,模型参数较多会导致训练过程容易发生过拟合现象。整个卷积神经网络的空间复杂度表示为:

$$Space\sim O\left(\sum_{l=1}^{D}K_l^2\cdot C_{l-1}\cdot C_l+\sum_{l=1}^{D}M_l^2\cdot C_l\right) \tag{14}$$

上式的第一个求和表达式代表总参数量与卷积核大小 K、通道数 C 和层数 D 相关,第二个求和表达式说明输出特征图的空间占比是输出特征图尺寸与通道数的乘积。

模型的计算强度 I 是模型计算量与访存量的比值,单位是FLOPs/Byte,表示模型在计算过程中单位内存交换完成浮点计算次数。模型计算强度越大,内存使用效率越高。

3.3 实验结果分析

表3是网络模型使用深度可分离卷积和Fire Module前后的参数量对比,其中模型YOLOv4和YOLOv5是未做任何改动的初始模型,Ghost-A仅将YOLOv4的主干网络替换为GhostNet,Ghost-B是在Ghost-A的基础上,对强特征提取网络和预测网络所有的3×3卷积中使用深度可分离卷积的模型,GS-YOLO是本文改进后的模型。通过相关轻量化操作,模型参数量由6 394万降至899万,降低约86%。

表3 不同模型的参数量对比

Table 3 Comparison of parameter numbers of different models

模型	主干网络	深度可分离卷积	Fire Module	参数量/10^4
YOLOv4	CSPDarknet53	×	×	6 394
YOLOv5	CSPDarknet53	×	×	4 143
Ghost-A	GhostNet	×	×	3 926
Ghost-B	GhostNet	√	×	1 100
GS-YOLO	GhostNet	√	√	899

为了验证这些改进点对模型性能的影响,将GS-YOLO与表3中的模型进行消融实验,根据表2的参数配置训练参数,输入尺寸均为416×416,结果如表4所示。

表4 不同模型的检测结果对比

Table 4 Comparison of detection results of different models

模型	AP/%	F1	检测速度CPU/ms	检测速度GPU/ms	模型大小/MB
YOLOv4	85.10	0.85	3 717	60.3	244.40
YOLOv5	83.55	0.80	632	58.0	366.93
Ghost-A	85.24	0.85	1 895	59.5	150.24
Ghost-B	84.26	0.83	631	58.7	42.50
GS-YOLO	85.55	0.85	626	55.9	34.77

模型大小是指模型占用存储空间字节数。由表4可知,使用GhostNet替换主干网络CSPDarknet53模型

后,模型占用内存降低38.53%,精度增加0.14%,CPU检测速度提高49.02%。在多尺度检测部分,使用深度可分离卷积代替传统卷积进一步减少了71.71%的内存占用空间,CPU和GPU检测速度分别提升66.70%和1.34%,精度损失仅为0.98%。在加强特征提取网络引入Fire Module后,检测精度提升1.29%,同时模型大小降低至34.77 MB,相较于原始YOLOv4占用内存244.40 MB,减小了约86%的内存空间,极大地减少了设备的资源消耗,提升了模型性价比。

根据Roofline Model[23]理论,模型实际测试性能会受到计算平台的算力和带宽影响,当模型计算强度I小于平台计算强度上限I_{max}的时候,模型性能受到平台带宽限制,处于带宽瓶颈区域(Memory-Bound);当模型计算强度I大于平台计算强度上限I_{max}的时候,模型处于计算瓶颈区域(Compute-Bound),虽受到平台算力限制,但可以充分利用平台的全部算力。

根据3.2节相关内容和NVIDIA官网提供的数据可知,NVIDIA TITAN RTX的算力π为16.3 TFLOP/s,带宽β为672 GB/s,平台计算强度上限I_{max}约为22.6 FLOPs/Byte。表5为各模型的理论性能及计算强度。

表5　性能参数及计算强度

Table 5　Performance parameters and operational intensity

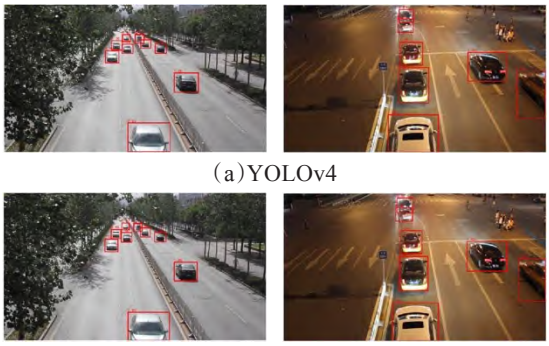
模型	特征图内存/MB	访存量/MB	计算量/GFLOPs	计算强度/(FLOPs/Byte)
YOLOv4	605.77	849.67	29.88	33.54
YOLOV5	252.75	410.81	18.17	42.18
Ghost-A	233.91	383.69	12.80	31.82
Ghost-B	262.14	304.11	3.26	10.23
GS-YOLO	253.91	288.21	2.92	9.67

经过对比发现GS-YOLO在GPU上运行时处于带宽瓶颈区域,性能受到平台带宽限制,无法像YOLOv4一样完全利用平台的全部算力,所以推理时间减少效果不明显。但在CPU处理器上运行时,GS-YOLO的性能不再受带宽限制,速度提升约83%,说明GS-YOLO更适合用于CPU或资源不足的设备上。

为了更直观地展示检测结果,使用GS-YOLO和YOLOv4进行车辆检测的结果对比如图8所示,(a)为YOLOv4的结果,(b)为GS-YOLO的结果。测试图像分别选取白天和夜间不同场景,从图中标出的目标框可以看出模型轻量化操作前后,目标检测准确率几乎没有差异。

4　结束语

本文提出一个适用于路侧交通监控场景的轻量型车辆检测模型GS-YOLO。GS-YOLO使用GhostNet作为主干网络,将SE模块和检测网络相融合,极大地压缩了模型占用内存大小。在多尺度检测部分使用深度可分离卷积代替传统卷积,降低大量卷积操作产生的计算量。此外,在网络中添加Fire Module也带来了精度和速度的提升。使用Roofline Model理论对GS-YOLO不同资源条件下的计算性能进行分析,结果表明,GS-YOLO在精度相对YOLOv4提升0.45%的同时,模型占用内存从244 MB降低到34.77 MB,CPU和GPU上的检测速度分别提升83.2%和7.3%,解决了YOLOv4检测效率低的问题,对资源不足、计算能力低的设备十分友好。



(a)YOLOv4

(b)GS-YOLO

图8　YOLOv4和GS-YOLO检测结果

Fig.8　Detection results of YOLOv4 and GS-YOLO

参考文献:

[1] LIU W,ANGUELOV D,ERHAN D,et al.SSD:single shot multibox detector[C]//European Conference on Computer Vision,Amsterdam,Oct 11-14,2016.Berlin:Springer,2016:21-37.

[2] LIN T Y,GOYAL P,GIRSHICK R,et al.Focal loss for dense object detection[C]//Proceedings of the IEEE International Conference on Computer Vision,Venice,Oct 22-29,2017.Washington D C:IEEE,2017:2980-2988.

[3] REDMON J,DIVVALA S,GIRSHICK R,et al.You only look once:unified,real-time object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,Las Vegas,Jun 26-Jul 1,2016.Piscataway:IEEE,2016:779-788.

[4] REDMON J,FARHADI A.YOLO9000:better,faster,stronger[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,Hawaii,Jul 21-26,2017.Piscataway:IEEE,2017:7263-7271.

[5] FARHADI A,REDMON J.Yolov3:an incremental improvement[C]//Computer Vision and Pattern Recognition,Salt Lake City,Jun 18-22,2018.Piscataway:IEEE,2018.

[6] REN S,HE K,GIRSHICK R,et al.Faster R-CNN:towards real-time object detection with region proposal networks[J].Advances in Neural Information Processing Systems,2015,28:91-99.

[7] CAI Z,VASCONCELOS N.Cascade R-CNN:delving into high quality object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,

Salt Lake City,Jun 18-22,2018.Piscataway:IEEE,2018:6154-6162.

[8] LAW H,DENG J.Cornernet:detecting objects as paired keypoints[C]//Proceedings of the European Conference on Computer Vision,Munich,Sept 8-14,2018.Berlin:Springer,2018:734-750.

[9] DUAN K,BAI S,XIE L,et al.Centernet:keypoint triplets for object detection[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision Long Beach,Jun 16-20,2019.Piscataway:IEEE,2019:6569-6578.

[10] 曹诗雨,刘跃虎,李辛昭.基于Fast R-CNN的车辆目标检测[J].中国图象图形学报,2017,22(5):671-677.
CAO S Y,LIU Y H,LI X Z.Vehicle detection method based on fast R-CNN[J].Journal of Image and Graphics,2017,22(5):671-677.

[11] 李松江,吴宁,王鹏,等.基于改进Cascade RCNN的车辆目标检测方法[J].计算机工程与应用,2021,57(5):123-130.
LI S J,WU N,WANG P,et al.Vehicle target detection method based on improved Cascade RCNN[J].Computer Engineering and Applications,2021,57(5):123-130.

[12] 杜金航,何宁.基于改进的YOLOv3道路车辆实时检测[J].计算机工程与应用,2020,56(11):26-32.
DU J H,HE N.Real-time road vehicles detection based on improved YOLOv3[J].Computer Engineering and Applications,2020,56(11):26-32.

[13] 金旺,易国洪,洪汉玉,等.基于卷积神经网络的实时车辆检测[J].计算机工程与应用,2021,57(5):222-228.
JIN W,YI G H,HONG H Y,et al.Real-time vehicle detection based on convolutional neural network[J].Computer Engineering and Applications,2021,57(5):222-228.

[14] 顾恭,徐旭东.改进YOLOv3的车辆实时检测与信息识别技术[J].计算机工程与应用,2020,56(22):173-184.
GU G,XU X D.Real-time vehicle detection and information recognition technology based on YOLOv3 improved algorithm[J].Computer Engineering and Applications,2020,56(22):173-184.

[15] 鲁博,瞿绍军.融合BiFPN和改进Yolov3-tiny网络的航拍图像车辆检测方法[J].小型微型计算机系统,2021,42(8):1694-1698.
LU B,ZHAI S J.Vehicle detection method in aerial images based on BiFPN and improved Yolov3-tiny network[J].Journal of Chinese Computer Systems,2021,42(8):1694-1698.

[16] 李宇昕,杨帆,刘钊等.基于改进残差网络的道口车辆分类方法[J].激光与光电子学进展,2021,58(4):384-390.
LI Y X,YANG F,LIU Z,et al.Classification method of crossing vehicle based on improved residual network[J].Laser & Optoelectronics Progress,2021,58(4):384-390.

[17] 李汉冰,徐春阳,胡超超.基于YOLOV3改进的实时车辆检测方法[J].激光与光电子学进展,2020,57(10):332-338.
LI H B,XU C Y,HU C C.Improved real-time vehicle detection method based on YOLOV3[J].Laser & Optoelectronics Progress,2020,57(10):332-338.

[18] GEIGER A,LENZ P,STILLER C,et al.Vision meets robotics:the kitti dataset[J].The International Journal of Robotics Research,2013,32(11):1231-1237.

[19] WEN L,DU D,CAI Z,et al.UA-DETRAC:a new benchmark and protocol for multi-object detection and tracking[J].Computer Vision and Image Understanding,2020,193:102907.

[20] BOCHKOVSKIY A,WANG C Y,LIAO H Y M.Yolov4:optimal speed and accuracy of object detection[EB/OL].(2020-04-23)[2021-07-05].http://arxiv.org/abs/2004.10934.

[21] IANDOLA F N,HAN S,MOSKEWICZ M W,et al.SqueezeNet:AlexNet-level accuracy with 50x fewer parameters and<0.5MB model size[EB/OL].(2016-11-04)[2021-08-21].http://arxiv.org/abs/1602.07360.

[22] HAN K,WANG Y,TIAN Q,et al.Ghostnet:more features from cheap operations[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,Washington D C,Jun 14-19,2020.Piscataway:IEEE,2020:1580-1589.

[23] WILLIAMS S,WATERMAN A,PATTERSON D.Roofline:an insightful visual performance model for multicore architectures[J].Communications of the ACM,2009,52(4):65-76.