

唐坤俊, 宁媛, 刘聂天和. 基于 YOLOv8s-CGSA 交通标志检测算法[J]. 智能计算机与应用, 2025, 15(5): 142-148. DOI: 10.20169/j.issn.2095-2163.250520

基于 YOLOv8s-CGSA 交通标志检测算法

唐坤俊¹, 宁媛¹, 刘聂天和²

(1 贵州大学 电气工程学院, 贵阳 550025; 2 贵州电网有限责任公司贵阳花溪供电局, 贵阳 550025)

摘要: 随着辅助驾驶的快速发展, 交通标志检测精度要求日益提高, 但在嵌入式设备部署的实时性仍然具有挑战。于是提出一种基于 GhostNet 改进的 YOLOv8s 目标检测算法 YOLOv8s-CGSA。首先, 采用 HardSwish 激活函数的 GhostConv 替代网络 Neck 部分的全部 Conv 模块, 降低模型参数量、提升表达能力的同时降低计算成本、提高推理速度。其次, 采用改进的 C2fGhost 替代原 C2f 模块, 进一步减少模型参数量、提高模型性能。最后, 引入 SA 注意力机制, 增强语义信息和位置信息的融合, 提高模型特征融合和检测性能。实验结果表明, 在中国道路交通标志检测数据集 TT100K 上, 相对于 YOLOv8s 原始模型 $mAP@0.5$ 提高了 3.4%, 模型参数减少了 26%, 模型减小了 4.1 mb, 在嵌入式设备 Jetson Xavier NX 上, FPS 提高了 39%, 实现了在嵌入式设备上对交通标志实时且准确的检测。

关键词: YOLOv8s; GhostNet; 嵌入式部署; SA 注意力机制

中图分类号: TP391.41

文献标志码: A

文章编号: 2095-2163(2025)05-0142-07

Traffic signs detection algorithm based on YOLOv8s-CGSA

TANG Kunjun¹, NING Yuan¹, LIUNIE Tianhe²

(1 College of Electrical Engineering, Guizhou University, Guiyang 550025, China;

2 Guiyang Huaxi Power Supply Bureau of Guizhou Power Grid Co., Ltd., Guiyang 550025, China)

Abstract: With the rapid development of assisted driving, traffic sign detection accuracy requirements are increasing day by day, but real-time deployment in embedded devices is still a challenge. The paper proposes an improved YOLOv8s target detection algorithm YOLOv8s-CGSA based on GhostNet. Firstly, the paper uses GhostConv of HardSwish activation function to replace all Conv modules in the Neck part of the network, reducing the number of model parameters and improving expression ability while reducing computational costs and improving inference speed. Secondly, the paper uses the improved C2fGhost to replace the original C2f module to further reduce the number of model parameters and improve model performance. Finally, the SA attention mechanism is introduced to enhance the fusion of semantic information and location information, and improve the model feature fusion and detection performance. Experimental results show that in the Chinese road traffic sign detection data set TT100K, compared with the YOLOv8s original model $mAP@0.5$ is increased by 2.7%, model parameters are reduced by 26%, the model is reduced by 4.1 mb, and FPS is improved on the embedded device Jetson Xavier NX. A real-time and accurate detection of traffic signs on embedded devices is achieved.

Key words: YOLOv8s; GhostNet; embedded deployment; SA attention mechanism

0 引言

随着无人驾驶与辅助驾驶系统的兴起, 道路交通标志的检测变得越来越重要。尽管大多数驾驶员熟悉常见的交通标志, 但当面对陌生的交通标志时, 驾驶员可能会出现短暂的迟疑, 从而导致潜在的交通事故风险。因此, 基于深度学习的交通标志检测方法具有广阔应用潜力, 不仅可以为驾驶员提供便

利, 减少由于不熟悉交通标志而引发的事故, 还可以在辅助驾驶系统中提前提示驾驶员, 从而提高道路交通的安全性。然而, 要实现这一目标, 需要建立一个高效、快速且易于部署的实时目标检测模型。

目标检测算法的发展可以分为 2 个主要阶段: 传统算法和深度学习算法^[1]。传统的目标检测算法需要使用滑动窗口对输入图像进行多次扫描以选择候选区域, 然后进行特征提取和分类, 这种方法的

基金项目: 贵州省科技计划基金(黔科合 ZK2022135)。

作者简介: 唐坤俊(1998—), 男, 硕士研究生, 主要研究方向: 深度学习目标检测, 模型轻量化部署; 刘聂天和(1998—), 男, 硕士研究生, 主要研究方向: 电网技术, 边缘计算。

通信作者: 宁媛(1968—), 女, 教授, 硕士生导师, 主要研究方向: 计算机视觉, 图像处理。Email: ee_yning@gzu.edu.cn。

收稿日期: 2023-10-23

哈尔滨工业大学主办 ◆ 专题设计与应用

计算复杂度高,不适合实时检测和边缘设备上的部署。相反,如果使用深度学习卷积神经网络逐级提取图像特征并自主学习获取特征信息,可以得到精度更高并且更适合边缘设备部署的模型,所以近年来多使用深度学习进行目标检测领域。

深度学习目标检测方法通常可分为 two-stage 和 one-stage 两类。基于 two-stage 的方法,例如 R-CNN^[2]、Fast R-CNN^[3]和 Faster R-CNN^[4]等需要首先生成候选区域,然后对这些区域进行目标预测和识别,尽管这类方法在检测准确性上表现出色,但速度较慢^[5],不适合部署在边缘设备上。与之相反,基于 one-stage 的方法,例如 SSD^[6]和 YOLO 系列^[7-10]通过目标检测网络直接进行目标的定位和分类,具有更快的检测速度和更高的效率。在这些方法中,YOLOv8s 作为最新的模型,不仅提供更高的准确性,还对网络结构进行了改进以提高推理速度。因此,本文的主要目标是改进 YOLOv8s 网络结构,以实现了对道路交通标志的快速准确检测,并将其有效地部署到边缘设备上。

在交通标志目标检测领域,先前的研究中 Lee 等学者^[11]提出了基于 SSD(Single Shot Multibox Detector)的目标检测算法,使用 VGG-16 模型进行交通标志的位置和边界估计,最高平均准确率(Mean Average Precision, *mAP*) 达到了 88.4%。然而,由于 VGG-16 的参数数量较大,网络结构复杂,因此在边缘设备上的性能表现较差。熊恩杰等学者^[12]在 YOLOv8n 的结构上

增加小目标检测头,增加了检测精度和小目标识别,但是模型计算复杂性有所增加,实际使用需要更多的计算资源。为了解决这一问题,本文提出了以下改进措施:

- (1)采用轻量级的 GhostConv 卷积神经网络以减少计算量,提高运行速度。
- (2)改变 GhostConv 模块的激活函数以增强模型的表达能力。
- (3)更改模型 C2f 模块中的 BottleNeck 为 GhostBottleNeck,减少参数数量和计算量。
- (4)在 Neck 部分引入 Shuffle Attention 注意力模块,以改善模型对不同区域的关注能力,从而提高检测性能。

1 YOLOv8 介绍

YOLOv8 是由 Ultralytics 团队在 2023 年 1 月提出,有 YOLOv8-n/s/m/l/x 几个版本,其中 YOLOv8s 网络对比之前 YOLO 的 s 网络,精度更高,推理速度更快。原模型相对之前的模型主要做了以下的改进:提出了全新的 C2f 模块,实现了进一步的轻量化,继续沿用 SPPF 模块;同时使用 anchor-free 机制直接预测对象的中心而不是已知锚框的偏移量,加速了非极大值抑制 *NMS*^[13],推理速度得到提升;还使用了 Decoupled-Head,即通过 2 个头分别输出 cls 与 reg;使用 DFL Loss + CIoU Loss 作为分类损失。YOLOv8 原始结构如图 1 所示。

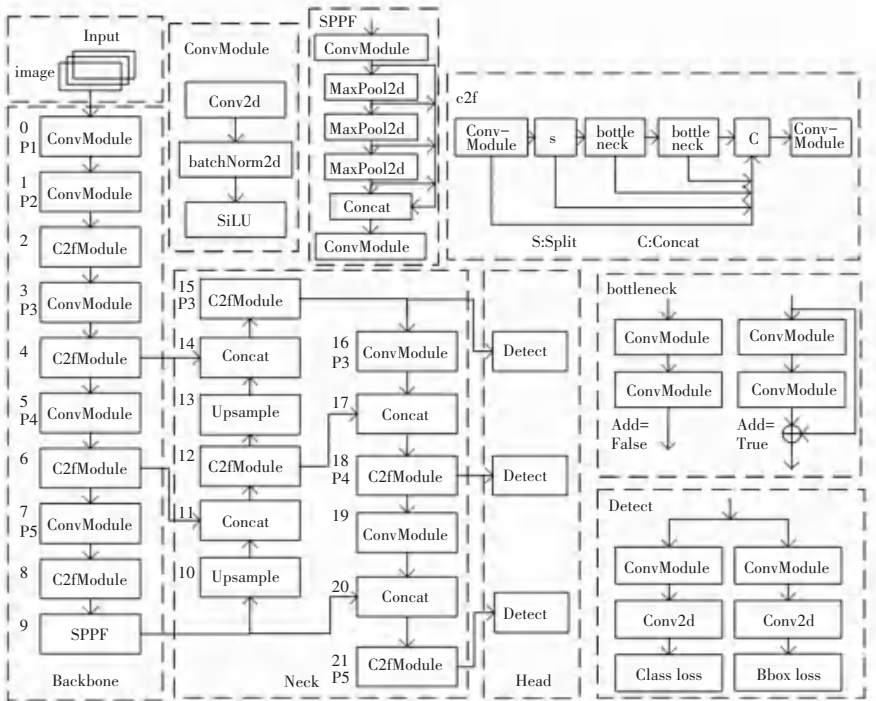


图 1 YOLOv8 结构图

Fig. 1 Original architecture of YOLOv8

Backbone 部分在目标检测模型中发挥着特征提取的重要作用,包括多个关键模块,如 ConvModule、C2f 以及 SPPF。其中,ConvModule 模块的主要功能是对输入图像执行一系列操作,包括卷积(Conv)、批归一化(Batch Normalization),以及激活函数 SiLU。这一过程有助于从原始图像中提取具有高级语义信息的特征。C2f 结构是 YOLOv8 一个全新的模块,在主干部分中扮演了学习残差特征的关键角色。SPPF 模块具备将任意大小的特征图转换成固定大小的特征向量的能力。这一特性使得模型能够处理不同尺寸的目标,同时保持对目标的全局感知。通过这些不同模块的组合,主干部分能够有效地提取图像特征,为后续的目标检测任务提供了关键的信息。

Neck 网络在目标检测模型中的主要任务是对多尺度的特征进行融合,从而生成特征金字塔,以增强 Backbone 部分提取的特征信息。YOLOv8 的 Neck 部

分采用了特征金字塔网络(FPN)和金字塔注意力网络(PAN)的结构,这两者的结合使得网络在不同尺度上能够更好地捕捉语义信息和定位信息。

Head 输出端作为最后的预测部分,根据不同尺寸的特征图来获取不同大小目标物体的类别和位置信息。

2 YOLOv8s 改进策略

本文主要基于 GhostNet^[14] 进行改进。GhostNet 是 2020 年华为诺亚实验室提出的轻量级卷积神经网络,GhostConv 是 GhostNet 网络中的一个卷积模块,这里用于替换原有网络 Neck 部分 Conv 模块,并更改 GhostConv 模块的激活函数为 HardSwish;改进 C2f 模块的 BottleNeck 为 GhostBottleNeck;最后引入 Shuffle Attention 注意力机制,改进后的网络结构如图 2 所示。

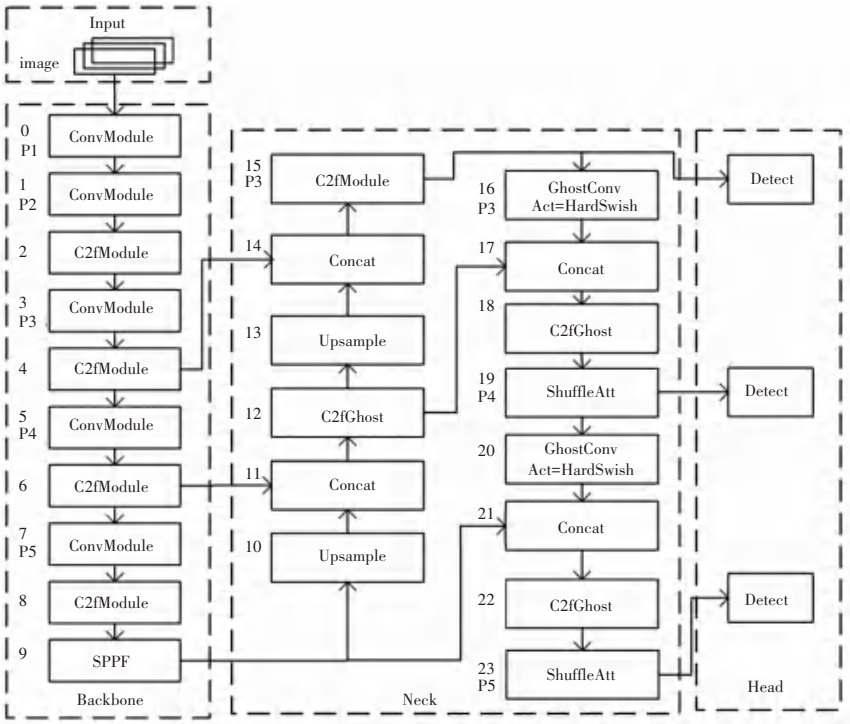


图 2 YOLOv8s-CGSA 网络结构图

Fig. 2 YOLOv8s-CGSA network architecture

2.1 基于 GhostNet 的设计

2.1.1 GhostConv 模块

Cheap linear operation 是原论文中提出降低运算量和参数量的操作,首先使用原卷积一半大小的卷积生成一半的特征图,接着继续经过一个 5×5 卷积核、步长为 1 的 Cheap linear operation 得到另一半特征图,最后通过 Concat 操作将 2 部分特征图拼接成完整的特征图。其中,ConvModule 的激活函数由

原来的 ReLU 改为更适合边缘设备的 HardSwish。GhostConv 模块如图 3 所示。

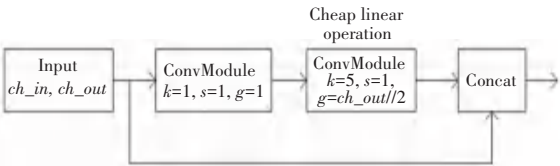


图 3 GhostConv 模块

Fig. 3 GhostConv module

首先 *Swish* 激活函数代替 *ReLU*,显著提高了神经网络的准确性,其具体定义为:

$$Swish(x) = x \cdot Sigmoid(x) \quad (1)$$

虽然这种非线性提高了精度,然而 *Sigmoid* 函数是由指数构成的,在边缘设备上的计算成本要高得多。*Sigmoid* 激活函数可以用分段线性函数 *HardSigmoid* 拟合:

$$HardSigmoid(x) = \begin{cases} 0, & x \leq -3 \\ 1, & x \geq 3 \\ \frac{x}{6} + \frac{1}{2}, & \text{otherwise} \end{cases} \quad (2)$$

由此,用 *HardSigmoid* 替代 *Sigmoid* 可以大大减少运算成本。*HardSwish* 的计算公式为:

$$HardSwish(x) = x \cdot HardSigmoid(x) = \begin{cases} 0, & x \leq -3 \\ x, & x \geq 3 \\ x \cdot (\frac{x}{6} + \frac{1}{2}), & \text{otherwise} \end{cases} \quad (3)$$

经过实验发现,该函数和 *Swish* 的效果几乎没有明显差别,但是对于部署来说具有很多的优势,可以减少内存的访问数量,大幅降低延迟成本。

2.1.2 基于 GhostBottleneck 改进 C2fGhost

GhostBottleneck 模块如图4所示,这个网络结构通过一系列层次化的操作,包括通道增加、通道减少、深度可分离卷积、正则化和 *SiLU* 激活函数的选择,以及残差连接的特征融合,来提高模型的性能和有效性。首先,通过第1个 *GhostConv* 模块,增加输入特征图的通道数量。接下来,根据 *Stride* 的数值,可以选择采用深度可分离卷积(*DWConv*)或正则化和 *SiLU* 激活函数,以适应不同的特征提取需求。然后,经过第2个 *GhostConv* 模块,将输出特征图的通道数量减少,使其与输入通道数量相匹配。最后,将上一步得到的特征图与残差相加,实现特征融合。这个设计可以帮助网络更好地理解图像中的语义信息,并提高模型对复杂场景的处理能力。

拥有了 *GhostBottleneck* 模块后,设计全新模块 *C2fGhost*,将原网络 *C2f* 模块中的 *Bottleneck* 全部替换为 *GhostBottleneck*。这个结构采用了跨阶段特征融合策略和截断梯度流技术,以增强不同网络层之间学习特征的可变性。这样做的主要目标是减少冗余梯度信息对模型的影响,从而提高网络的学习能力。通过引入 *GhostConv* 和 *C2fGhost* 模块,减少了原始结构中大量的 3×3 普通卷积操作。这一改进显著减小了网络的模型大小,降低了参数数量和计

算复杂度。使得模型可以轻松部署在嵌入式设备上,更容易实现交通标志的边缘计算检测任务。以 *GhostBottleneck* 数量为3的示意如图4所示。研发得到的 *C2fGhost* 模块如图5所示。

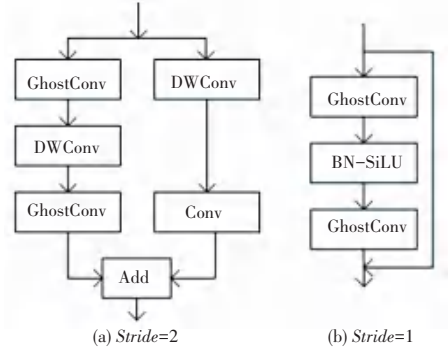


图4 GhostBottleneck 模块

Fig. 4 GhostBottleneck module

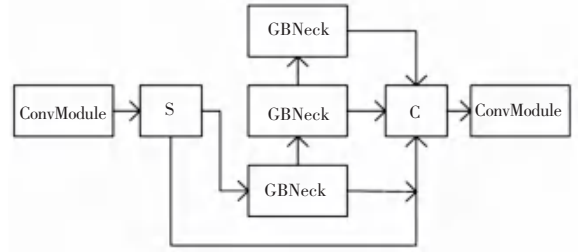


图5 C2fGhost 模块

Fig. 5 C2fGhost module

2.2 Shuffle Attention 注意力

Shuffle Attention 是一种用于计算机视觉的注意力机制,主要聚焦于通道级别的特征注意力,通过学习不同通道之间的关系来选择性地强调图像特征中的重要信息,同时引入通道重排以增强特征之间的交互性。这个机制允许模型自适应地学习通道权重,以适应不同任务和输入情境,减少了冗余梯度信息的影响,提高了模型检测性能,同时保持了较低的计算复杂性,适用于嵌入式环境。*Shuffle Attention* 模块结构如图6所示。

首先,将待检测的特征图传递给 *Shuffle Attention* 模块。接着,*Shuffle Attention* 会对特征图中的通道进行智能选择,以便更好地捕获不同通道之间的关联性。这种通道级别的权重分配有助于突出重要的特征,提高了模型的鲁棒性。此外,*Shuffle Attention* 还引入了通道重排操作,以促进通道之间的信息流动,从而增强了特征图的表达能力。通过这种方式,能够更好地利用不同通道的信息,进一步提高了模型的性能。最终,本文的实验结果表明,通过在网络 Neck 部分 *C2fGhost* 模块的后面应用 *Shuffle Attention*,本次研究有效提高了模型在交通标志目标检测中的性能。

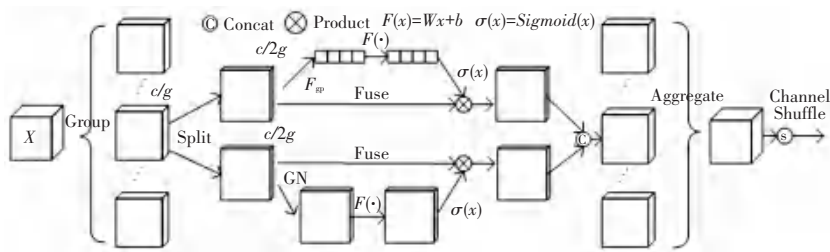


图 6 Shuffle Attention 模块

Fig. 6 Shuffle Attention module

3 实验设计与结果分析

3.1 数据集与实验环境

TT100K 数据集是由清华大学与腾讯实验室联合发布的基于真实场景的交通标志公开数据集^[15], 整个数据集一共出现了 151 种不同的类别, 各类别实例数存在严重的不均衡现象, 争对出现的情况, 本文筛选出类别大于 100 的交通标志, 并添加部分没有标签的图, 增加泛化性, 最后总共得到 45 类, 其中 6 793 张作为训练集, 1 949 张作为验证集。

训练环境: 所使用环境操作系统为 Windows10, 计算资源为 1 张 NVIDIA RTX 2060 显卡, 深度学习框架为 Pytorch1. 8. 1。训练时参数设置: 优化器选用随机梯度下降 (Stochastic Gradient Descent, SGD), 动量设置为 0. 937, 初始学习率为 0. 01, 共训练 100 个 epochs。

3.2 评价指标

本实验使用精确率 (Precision)、召回率 (Recall)、平均精度均值 (mAP@0. 5)、Params 和 GFLOPs 对模型进行评价。其中, 精确率是指正确预

测为正确 (TP) 的占全部预测 (TP + FP) 的比例, 可用如下公式进行计算:

$$Precision = \frac{TP}{TP + FP}$$
 (4)

召回率是指正确预测为正确 (TP) 的占实际 (TP + FN) 的比例, 计算公式具体如下:

$$Recall = \frac{TP}{TP + FN}$$
 (5)

Params 和 GFLOPs 用于做实际部署的参考。n 个类别的平均准确率计算公式如下:

$$mAP = \frac{1}{n} \sum_{i=1}^n \int_0^1 Precision(Recall) d(Recall)$$
 (6)

模型对比结果见表 1。根据表 1, YOLOv8s 相比 YOLOv5s 虽然参数量有所增加, 但是精度更高, 并且经过对模型的加速, 使得更适合在嵌入式上部署。同时为了验证改进模块是否有效做了消融实验, 实验结果见表 2。根据表 2 中消融实验结果, GhostConv 不是越多越好, 如果在网络的 Backbone 部分添加了改进的 GhostConv, 将会因为参数量低造成模型提取信息能力不足, 导致模型整体性能变差。

表 1 模型对比

Table 1 Model comparison

Model	P/%	R/%	mAP@0. 5/%	Params/ M	GFLOPs
Faster-RCNN	72. 3	71. 0	74. 2	28. 5	369. 1
YOLOx-s	73. 2	67. 2	70. 2	9. 3	35. 1
YOLOv5s	63. 7	65. 9	69. 1	7. 1	15. 8
YOLOv8s-C3Ghost	72. 0	62. 4	69. 0	5. 2	13. 2
YOLOv8s-C2fGhost-SE	79. 7	65. 2	75. 7	9. 0	24. 4
YOLOv8s-CGSA(ours)	80. 6	69. 6	78. 2	8. 9	24. 4

表 2 消融实验

Table 2 Ablation experiment

Model	P/%	R/%	mAP@0. 5/%	Params/ M	GFLOPs
YOLOv8s	76. 9	65. 4	74. 8	11. 1	28. 5
YOLOv8s-C2fGhost-Neck	77. 8	68. 9	75. 5	8. 9	24. 4
YOLOv8s-GhostConv-ALL	75. 5	65. 1	73. 4	8. 0	21. 6
YOLOv8s-CGSA(ours)	80. 6	69. 6	78. 2	8. 9	24. 4

3.3 模型部署

模型部署的一般流程涵盖了多个关键步骤,部署流程如图 7 所示。首先将训练得到的模型保存为通用的中间格式 ONNX (Open Neural Network Exchange),这个开放文件格式在工业界得到广泛使用。ONNX 的优点在于可以轻松转换来自不同训练框架的模型,使得模型迁移和部署变得更加方便。接下来,将 ONNX 模型转换为 TensorRT(NVIDIA 的深度学习推理加速库)对应的 engine 文件,以实现模型的加速部署。在这个过程中,需要进行模型性能测试,以确保在设备上的推理速度和效率达到预期。在性能评估中,使用 2 个关键指标来衡量模型的实际性能 *mAP* 和 *FPS*。其中,*mAP* 表示模型的检测准确度, *FPS* 表示模型在单位时间内处理的帧数,这 2 个指标共同反映了模型的性能水平。

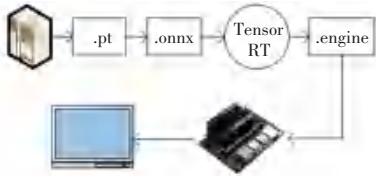


图 7 部署流程
Fig. 7 Deployment process

表 3 模型部署表现
Table 3 Model deployment performance

Model	ONNX- <i>mAP</i> @0.5/ %	TRT-fp32 <i>mAP</i> @0.5/ %	TRT-fp16 <i>mAP</i> @0.5/ %	<i>fps</i> / (f · s ⁻¹)
YOLOv8s	75.8	75.8	75.4	17
YOLOv8s-C2fGhost	75.5	75.5	75.2	20
Yolov8s- C2fGhost-SE	72.3	72.3	71.8	21
YOLOv8s-CGSA(本文)	78.2	78.2	78.0	24



图 8 实验检测结果
Fig. 8 Experimental results

4 结束语

为了在嵌入式设备 Jetson Xavier NX 上实现高效的交通标志检测,本文采用了最新的目标检测算法 YOLOv8,并对其进行了一系列改进。这些改进主要包括使用 GhostNet 对原本较大的 C2f 模块进行轻量化设计,创造了更加精简的 C2fGhost 模块;还替换了原本的卷积模块为 GhostConv,并将模块的激

在实际的模型部署过程中,选择合适的推理精度对于性能和效率至关重要。尽管 INT8 推理在理论上可以减少模型的存储和计算开销,但实际应用中,由于 CPU 和 GPU 之间的通信和反量化操作,会导致推理时间显著延长,同时可能会引起模型精度的较大下降。考虑到这些问题,本文选择了采用 FP16 进行推理,这种方法减半了模型文件中浮点数的精度,但几乎不会对检测模型的精度造成显著影响。这使得模型具有更广泛的适用性,同时提高了推理速度,使本文研发的模型能够轻松部署在边缘设备上,并实现实时的检测效果。

经过对 engine 模型进行测试,半精度量化仅损失了本文模型约 0.3 个百分点的性能,但在推理速度方面却获得了显著的提升。这使得本文研发的模型能够满足边缘设备上实时检测的需求,为实际应用提供了更快速和高效的解决方案。

模型部署表现见表 3。由表 3 可知,改进后的网络模型比原模型的速度得到提升,在保证检测精度的条件下,大幅提升了检测速度,实现了模型的轻量化,可以将该模型部署到边缘设备进行实时检测。实验检测结果示例如图 8 所示。

活函数改为 *HardSwish*,以更好地适应边缘设备的推理计算需求;此外,在网络的 Neck 部分引入了注意力机制 SA,以提高模型的检测精度。最终,使用 TensorRT 加速改进后的模型,从而成功部署在边缘设备上,实现了高效的交通标志识别。

然而,本文仍然存在一些改进空间。例如,模型仍需要进一步提升在小目标检测方面的性能,因为在实际行驶过程中,交通标志可能会非常小,需要及时识别,尤其是在远距离情况下。此外,数据集的标签数量不均匀也会影响模型的鲁棒性和泛化性能。最后,由于仅针对中国道路上的交通标志进行了研究,模型的跨国使用可能会面临一些挑战。为了解决这些问题,未来的研究将根据项目需求,收集不同国家特定的交通标志,并扩充数据集,以不断提升检测精度,从而实现更广泛的应用。

参考文献

- [1] HOESER T, KUENZER C. Object detection and image segmentation with deep learning on earth observation data: A review – part II: Evolution and recent trends[J]. Remote Sensing, 2020, 12(10): 1667.
- [2] AGRAWAL P, GIRSHICK R, MALIK J. Analyzing the performance of multilayer neural networks for object recognition [C]// Proceedings of 13th European Conference on Computer Vision (ECCV 2014). Cham: Springer, 2014: 329–344.
- [3] GIRSHICK R. Fast R – CNN [C]//Proceedings of the IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE, 2015: 1440–1448.
- [4] REN Shaoqing, HE Kaiming, GIRSHICK R, et al. Faster R – CNN: Towards real – time object detection with region proposal networks[J]. arXiv preprint arXiv, 1506.01497, 2016.
- [5] 张寅, 朱桂熠, 施天俊, 等. 基于特征融合与注意力的遥感图像小目标检测[J]. 光学学报, 2022, 42(24): 2415001.
- [6] LIU Wei, ANGUELOV D, ERHAN D, et al. SSD: Single shot multibox detector[C]// Proceedings of 14th European Conference on Computer Vision(ECCV 2016). Cham: Springer, 2016: 21–37.
- [7] 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. Piscataway, NJ: IEEE, 2016: 779–788.
- [8] REDMON J, FARHADI A. YOLO9000: better, faster, stronger [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 7263–7271.
- [9] REDMON J, FARHADI A. Yolov3: An incremental improvement [J]. arXiv preprint arXiv, 1804.02767, 2018.
- [10] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv, 2004.10934, 2020.
- [11] LEE H S, KIM K. Simultaneous traffic sign detection and boundary estimation using convolutional neural network[J]. IEEE Transactions on Intelligent Transportation Systems, 2018, 19(5): 1652–1663.
- [12] 熊恩杰, 张荣芬, 刘宇红, 等. 面向交通标志的 Ghost-YOLOv8 检测算法[J]. 计算机工程与应用, 2023, 59(20): 200–207.
- [13] HOSANG J, BENENSON R, SCHIELE B. Learning non – maximum suppression [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 4507–4515.
- [14] HAN Kai, WANG Yunhe, TIAN Qi, et al. Ghostnet: More features from cheap operations [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 1580–1589.
- [15] ZHU Zhe, LIANG Dun, ZHANG Songhai, et al. Traffic – sign detection and classification in the wild [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 2110–2118.