分类号： U471.15 学校代码:10363
密 级： 公开 学 号： 2210120168

安徽工程大学
Anhui Polytechnic University

硕士学位论文

题目 基于改进 YOLOv5 的交通标志识别研究

论 文 作 者 郑昆明
校 内 导 师 张荣芸(副教授)
校 外 导 师 赵夕长(高级工程师)
专业学位类别 机 械
研究方向（领域） 智能驾驶及环境感知

论文提交日期： 2024 年 5 月 25 日

分类号：U471.15　　单位代码：10363
密 级：公开　　学　　号：2210120168

基于改进 YOLOv5 的交通标志识别研究

Research on Traffic Sign Recognition Based on Improved YOLOv5

学　生　姓　名：郑昆明
校　内　导　师：张荣芸(副教授)
校　外　导　师：赵夕长(高级工程师)
专 业 学 位 类 别：机　械
研究方向　(领域)：智能驾驶及环境感知

论 文 答 辩 日 期：2024 年　5 月 25 日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名：郑昆明

日期： 2024 年 5 月 31 日

安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 □，在　　年解密后适用本版权书。

本学位论文属于

不保密☑。

学位论文作者签名：郑昆明　　指导老师签名：张荣芝

日期：　2024 年　5 月 31 日　　日期：2024 年 5　月 31 日

基于改进 YOLOv5 的交通标志识别研究

摘 要

在智能辅助驾驶系统和自动驾驶技术中，交通标志识别扮演着至关重要的角色。它能帮助驾驶员或自动驾驶车辆遵守交通规则并理解道路环境。通过准确地识别交通标志，自动驾驶汽车可以及时响应道路交通情况，确保行驶安全。传统的交通标志检测方法由于其较慢的检测速度和低精度已无法满足当今智能驾驶的需求。尽管目前存在许多基于卷积神经网络的检测算法，它们提高了交通标志检测算法的速度，但检测精度却有所下降，同时模型参数量过大，难以在移动设备上进行部署。因此，本文以道路交通标志为背景，以实时性较好的 YOLOv5 算法为基础开展研究。主要的工作总结如下：

(1) 针对智能辅助驾驶系统中当前交通标志识别技术存在的实时性和准确性不足问题，提出了一种 CBAM-TSR 算法。该算法采用 ACONC 激活函数对 SiLU 激活函数进行了改进，提升了模型的收敛性和鲁棒性；并通过利用 SIoU 损失函数对 GIoU 损失函数进行了改进，达到了优化训练模型的目的，提高了识别精度。最后利用 TT100K 数据集对本文提出的检测算法进行了训练，结果表明，在 TT100k 数据集上，通过在主干网络中融入卷积注意力模块的方法，可以在特征图的卷积过程中发现目标对象后聚焦于目标对象，并抑制特征图中的无关特征信息，从而实现对小目标区域的有效检测。其中本文提出的算法检测精度较未改进 YOLOv5 方法提升了 6.2%，检测速度提升了 3.54 帧/秒。

(2) 为解决交通标志算法存在模型参数量大，检测精度低的问题，本文提出了一种轻量化的 GCC-TSR 算法。首先，通过采用 Ghost 卷积和 CBAM 模块级联的结构来优化 YOLOv5 主干特征提取网络，从而降低模型复杂度，提高了交通标志的检测速度。其次，在特征融合层引入 CARAFE 上采样模块，该模块能够更好地捕获小目标特征信息，并恢复检测目标的局部细节。最后，采用 WIoU 作为模型损失函数，以提高模型精度并加速模型的收敛速度。结果表明，改进后的 YOLOv5 模型，mAP 提升 1.8%，为 91.9%，权重下降为原来的 65.1%，参数量下降为原来的 58.8%，检测速率提升了 8.6 帧/秒。

关键词：交通标志，YOLOv5，轻量化，卷积注意力机制，激活函数，损失函数

RESEARCH ON TRAFFIC SIGN RECOGNITION BASED ON IMPROVED YOLOV5

ABSTRACT

Traffic Sign Recognition (TSR) plays a crucial role in both intelligent driver assistance systems and autonomous driving technologies. It assists drivers in adhering to traffic regulations and understanding the road environment. Accurate recognition of traffic signs enables autonomous vehicles to promptly respond to traffic situations, ensuring safe driving. Traditional methods of traffic sign detection, due to their slow detection speed and low accuracy, are inadequate to meet the demands of modern intelligent driving. Despite the existence of many convolutional neural network-based detection algorithms, while they enhance the speed of traffic sign detection, there is a decrease in detection accuracy, and the models have excessive parameters, making deployment on mobile devices challenging. Therefore, this paper focuses on research conducted against the backdrop of road traffic signs, employing the YOLOv5 algorithm, known for its real-time capabilities, as the foundation. The main contributions and findings of this work are summarized as follows:

(1) In response to the insufficient real-time performance and accuracy of current traffic sign recognition technology in intelligent assisted driving systems, a CBAM-TSR algorithm is proposed. This algorithm improves the convergence and robustness of the model by enhancing the SiLU activation function with the ACONC activation function. Additionally, by utilizing the SIoU loss function to

refine the GIoU loss function, the algorithm achieves the goal of optimizing the training model and enhancing recognition accuracy. Finally, the proposed detection algorithm is trained on the TT100K dataset. Results indicate that by incorporating convolutional attention modules into the backbone network, the algorithm focuses on target objects during the convolution process of feature maps and suppresses irrelevant feature information, thereby effectively detecting small target regions. The proposed algorithm improves detection accuracy by 6.2% compared to the unmodified YOLOv5 method and increases detection speed by 3.54 frames per second.

(2) To address the issues of large model parameter size and low detection accuracy in traffic sign algorithms, a lightweight GCC-TSR algorithm is proposed in this paper. Firstly, by employing a structure that cascades Ghost convolution and CBAM modules to optimize the YOLOv5 backbone feature extraction network, model complexity is reduced, thereby increasing the detection speed of traffic signs. Secondly, introducing the CARAFE upsampling module at the feature fusion layer enables better capture of feature information for small targets and restores local details of detected targets. Finally, WIoU is adopted as the model loss function to enhance model accuracy and accelerate model convergence speed. The results demonstrate that the improved YOLOv5 model achieves a 1.8% increase in mAP, reaching 91.9%, with a weight reduction to 65.1% of the original and a parameter reduction to 58.8% of the original. The detection speed is increased by 8.6 frames per second.

KEY WORDS: traffic sign, yolov5, lightweight, convolutional block attention module, activation function, loss function

目 录

第 1 章 绪论

1.1 课题研究背景及意义

随着城市化的发展，各种车辆的数量逐年增加，由此引起的一些交通问题也日益严重，其中之一就是导致道路交通拥堵问题日益严重[1,2]。图 1-1 为国内近十年民用汽车保有量。城市道路的交通流量不断增加引发的交通拥堵现象频发，因此，确保交通安全已经成为社会中的重要任务[3,4]。道路交通拥堵凸显了交通标志识别的重要性，因为准确识别交通标志能够帮助自动驾驶系统及时调整行驶策略，提高交通流畅度，减少拥堵发生的可能性，从而提高道路通行效率并确保行车安全。交通标志在指挥车辆行驶和传递交通规则方面起着重要作用，它们设置在道路两侧，通常包含着非常重要的信息。这些标志的作用不仅仅局限于指导车辆，还涉及到交通拥堵带来的诸多负面影响，包括经济损失、时间浪费，以及对环境、健康、生活质量和安全等多个方面的影响，对城市和个人的发展都造成了不利影响。其中警示标志反映当前路况信息，如弯道、陡坡等，对路况进行预警，禁止超车、禁止转弯、限速标志等禁止标志规范驾驶员行为，引导交通有序进行[5]。然而，由于各种因素的影响，交通标志经常受到污损、损坏或遮挡，导致驾驶员无法准确识别，从而增加了交通事故的风险，降低了交通管理的效率。

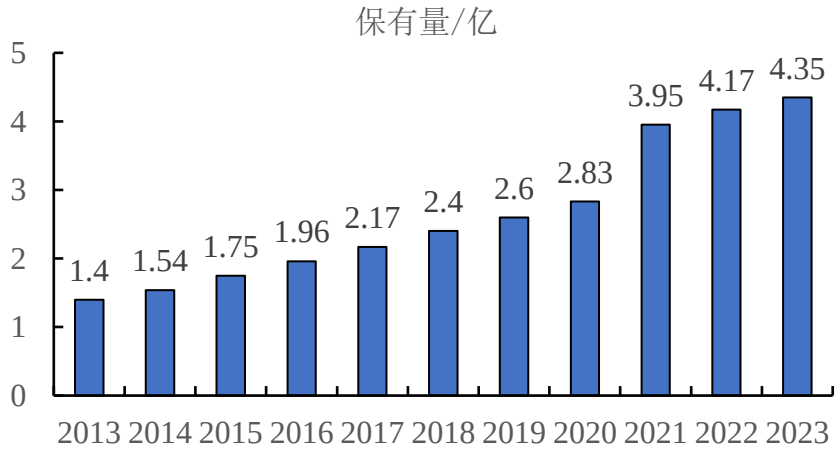


图 1-1 国内近十年民用汽车保有量

因此，对交通标志识别技术的研究成为解决这一难题的重要途径[6]。传统的交通标志检测方法主要依赖于手工设计的特征提取器和分类器，但检测速度较低。随着卷积神经网络 (Convolutional Neural Network，CNN) 在目标检测中的广泛应用，交通标志检测领域也迎来了新的发展机遇。CNN 能够自动学习和提取图像中的特征，具有较强的适应性和泛化能力，可以对道路上的交通标志进行快速、准确的识别，通过实时监测和识别道路上的交通标志，交通管理部门可以及时调整交通信号灯、引导交通流动，优化交通路线规划，从而缓解交通拥堵[7]。

随着人工智能、感知技术和车辆通信技术的发展，车辆智能辅助驾驶系统能够借助先进的传感器系统，如激光雷达、摄像头等，实时感知周围环境，迅速做出决策并执行精准操控[8]。智能辅助驾驶技术的发展不仅为解决交通问题提供了新的可能性，也为提升出行体验和交通效率带来了巨大的潜力。通过实现车辆之间的智能协同和对交通流的精确预测，智能辅助驾驶系统有望为出行提供更加安全、高效和便捷的选择[9]。在智能辅助驾驶领域，交通标志检测系统扮演着关键角色。它能够收集并提取车辆在行驶过程中道路上交通标志的关键信息，并及时向驾驶员发送警示信息，从而为实现车辆环境感知、分析和决策提供了基础[10,11]，同时降低事故发生的可能性[12-14]。智能驾驶车辆中的交通标志识别任务属于基于视觉的目标检测范畴，对模型的设计提出了鲁棒性、高精度、低计算资源消耗和快速处理等方面的要求，这是一个具有挑战性的领域[15,16]。

综上所述，交通标志检测的研究具有重要的理论和实践意义。通过不断提升交通标志检测技术的准确性、鲁棒性和实用性，可以更好地应对交通安全和交通管理方面的挑战，推动智能辅助驾驶系统的发展。

1.2 交通标志识别国内外研究现状

交通标志识别是在给定特定分类的情况下，从数字图像或视频图像中识别交通标志的位置并进行检测[17]。其方法基本上都采用视觉信息，例如交通标志的形

状和颜色。然而，在真实道路场景中，常常出现物体与真实的交通标志相似的情况，这可能导致检测效果不佳。虽然传统方法可以用于检测交通标志，但其过程较为繁琐，提取特征的能力较差，无法满足端到端的智能驾驶的要求[18]。

交通标志识别技术的发展历程涵盖了传统方法和基于深度学习方法两个显著的阶段[19]。在传统方法中，研究人员主要依赖尺度不变特征变换 (Scale-Invariant Feature Transform, SIFT)[20]和方向梯度直方图 (HOG)[21]等手工特征提取算法来获取交通标志的特征。传统的交通标志识别方法具有一些特点和局限性。首先，它们主要依赖于手动设计的特征，对光照条件、遮挡和复杂背景的变化敏感[22]。其次，这些方法通常难以适应多样化的交通标志数据集和真实世界的场景[23]。虽存在一些局限性，但传统方法为早期交通标志识别研究奠定了基础，并在受控条件下表现出了合理的性能[24-26]。深度学习方法引入了区域提议网络 (Region Proposal Network, RPN)，例如 Fast R-CNN 和 Faster R-CNN，通过生成候选目标区域，使得模型能够集中精力学习具体目标的特征。同时，单发多框检测器 (SSD) 等模型在提高检测速度和保持准确性方面取得了巨大进展。如 YOLO (You Only Look Once, YOLO)和 EfficientDet[27]，大大提高了目标检测的准确性和效率。

在上世纪 80 年代末，国内外学者开始探索基于传统手工特征的检测算法，旨在深入研究复杂特征提取网络，以实现对输入图像的准确识别[28-29]。当一张图片上存在多个物体时，该算法通过提取候选区域，将整个图像分割成多个部分，以尽可能准确地截取目标物体。因此，候选区域的准确性对于模型的性能至关重要，准确的候选区域能够显著提高模型的精确率和召回率。图 1-2 为传统检测算法的流程图。

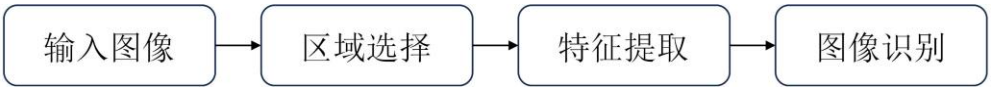


图 1-2 传统检测算法流程图

在这一流程中，首先通过候选区域提取的步骤，算法会对整个图像进行分割，获得多个候选区域。接着，对每个候选区域进行特征提取，以捕获目标的关键信息。最后，通过分类器的训练，模型能够对提取的特征进行分类，从而识别图像中的不同目标。然而，这种传统方法也存在一些限制。首先，手工设计的特征提取网络可能无法充分捕获复杂图像中的信息，从而导致模型的泛化能力较弱。其次，对于不同任务和场景，需要调整和修改特征提取网络，增加了算法的复杂性。

在 2014 年之前，手动提取特征的传统方法仍然是主导的目标检测手段。早期的 TSM 方法通常依赖于手工设计的特征，如 SIFT 或加速稳健特征 (Speeded Up Robust Feature, SURF)，以提取局部描述符并执行相似性匹配[30]。在文献[31]中，Ren 等人提出了一种用于通用交通标志识别的方法(针对新西兰道路标志进行了测试)，使用 Hough 变换检测特殊形状的潜在标志，最后通过使用特征匹配方法(如 SIFT 或 SURF 特征)。在文献[32]中，He 等人提出了一种基于视觉检查的交通标志自动识别算法。为了提高视觉检查的准确性，通过内容分析和关键信息识别设计了一个感兴趣区域提取方法，此外，还开发了一种基于方向梯度直方图的图像检测方法，以防止投影失真，最后基于 CapsNet 创建了一个交通标志识别学习架构，该架构依赖于神经元来表示目标参数。在文献[33]中，Tabernik 等人提出了一种基于 Mask R-CNN 检测器的检测和识别方法，该方法包含了基于几何和外观失真分布的数据增强技术以及类间差异较小且类内变异性较大的 DFG 交通标志数据集，同时为系统学习大量类别提供了高效且快速检测的深度网络。在文献[34]中，Kumar 等人提出了一种新颖的基于 Goat-base 卷积神经卡尔曼框架的方法来检测城市道路中的交通标志。首先，通过卡尔曼函数对输入的交通标志图像进行噪声过滤，并提取用于识别过程的相关特征。然后，通过将这些特征与训练集进行匹配并使用 Goat 适应度函数进行分类来识别交通标志。

在文献[35]中，Zhang 等人提出了一种检测自然场景图像中道路交通标志的新方法。首先，基于像素向量的自适应颜色分割方法将彩色图像分割成二值图像并突出显示交通标志区域，这可以减少光照条件对图像分割的影响。其次，为了提高交通标志检测中的形状识别能力，采用中心投影变换计算不同候选区域的形状特征向量，将这些形状特征输入到概率神经网络中以区分真实的交通标志和候选标志。在文献[36]中，Qu 等人提出了一种基于知识蒸馏的改进型 SEDG-Yolov5 交通标志检测方法。首先，采用切片辅助超推理方法作为模型训练的本地离线数据增强方法。其次，为了解决高维特征信息损失和高模型复杂度的问题，提出了带有融合注意力机制的倒残差结构 ESGBlock，并基于此构建了轻量级特征提取骨干网络，同时在特征融合层引入 GSConv 以进一步降低模型的计算复杂度。最后，提出了一种改进的基于响应的物体性知识蒸馏方法，以重新训练交通标志检测模型，以弥补由于轻量化而导致的检测精度下降。Min 等人[37]提出了一种新的基于密集连接风格、多尺度特征融合和改进的 K-means++算法的多尺度密集连接对象检测器。然后，通过过滤场景结构模型外的虚假候选对象，找到了可信的交通标志。在文献[38]中，Zhang 等人提出了一种复杂环境下交通标志识别算法。算法包括两个模块，首先，提出了一种基于渐进多尺度残差网络的图像去雨算法，该算法利用多尺度残差结构提取不同尺度的特征，以提高算法对信息的利用率，并结合卷积长短时记忆网络以增强算法提取雨水痕迹特征的能力。其次，使用 CoT-YOLOv5 算法在恢复的图像上识别交通标志。最后在特征提取模块中用 Contextual Transformer 模块替代了 3 × 3 卷积，弥补了卷积神经网络全局建模能力的不足。

综上所述，这些研究方法在解决交通标志识别问题上提出了不同的思路和技术方案，为交通安全和智能辅助驾驶系统的发展做出了重要贡献。然而，当前的研究还存在一些挑战，如复杂天气条件下的识别准确性、小尺寸和遮挡交通标志的检测等，需要进一步的研究和改进。

1.3 交通标志识别与检测当前存在的难点

如前所述，尽管目前出现了许多优秀的用于交通标志检测和识别任务算法，并且在当前可用的数据集上取得了较好的识别率，但这些算法仅仅只在有限的数据集上进行了实验，没有考虑到现实情况下可能存在的各种复杂、不确定和无法预知的情况。由于拍摄光线的变化、拍摄环境以及数据集样本容量的限制，仍然存在一些影响交通标志识别的因素，具体表现在以下几个方面：

(1) 复杂场景和遮挡： 在城市环境中，交通标志常常受到建筑物、树木、车辆等遮挡，导致图像中的标志部分或全部被遮挡。解决这个问题需要开发能够识别部分遮挡标志或在复杂背景中精确定位标志的算法。

(2) 光照变化： 不同的天气条件和光照水平可能会导致图像中的交通标志外观发生变化，从而增加了识别的难度。研究者需要设计能够在各种光照条件下鲁棒性强的模型，以确保在不同环境下的可靠性。

(3) 小样本问题： 一些特殊类型的交通标志可能在数据集中的样本数量较少，或者在特定地区的标志可能没有足够的样本进行训练，目前国内公开的数据集仅有清华大学和腾讯共同发布的 TT100k 数据集和长沙理工大学的 CCTSDB 数据集，解决这个问题需要使用小样本学习、迁移学习等技术，以提高模型在新标志或特殊标志上的性能。

(4) 实时性要求： 在自动驾驶和智能交通系统中，交通标志识别实时性要求较高。设计高效实时识别系统，保持高准确性是一个具有挑战性的任务，尤其是在计算资源有限的嵌入式设备上。

针对以上问题，本文主要研究内容是解决检测精度低、实时性不足、模型尺寸大等问题。设计了高效实时识别系统，保持了高准确性，对边缘设备限制进行了考虑，并对模型进行了优化，以满足硬件资源的限制。

1.4 课题主要研究内容和结构安排

1.4.1 研究内容

通过大量文献的阅读，对于交通标志检测与识别中常用的技术方法有了充分的了解。针对当前存在的问题，如检测精度低、速度慢、模型尺寸大等，本文分析了交通标志检测中存在的问题，并提出了基于深度学习的交通标志检测与识别的方法。具体而言，本文的研究分为两个主要部分。

第一部分介绍了一种改进的 YOLOv5 方法，其中利用 SIoU 损失函数替换了YOLOv5模型中的GIOU损失函数，以优化训练模型。同时，在主干网络的CSP1_3模块中融入了卷积注意力模块，形成了新的 CSP1_3CBAM 模块。此外，还引入了ACONC 作为 YOLOv5 的激活函数，以提升模型的收敛性和鲁棒性。这一部分的改进可以增强模型的特征提取能力，从而提高对小目标的检测精度。第二部分是提出了一种基于 Ghost 卷积和注意力机制级联的结构来优化 YOLOv5 主干特征提取网络，从而减少模型的参数量，提高交通标志的检测速度，采用轻量级通用上采样算子 (Content-Aware ReAssembly of Features, CARAFE) 恢复特征分辨率，该部分可以使网络变得轻量化。

1.4.2 章节安排

第 1 章：绪论。本章介绍了课题研究的背景，并对比了传统的交通标志检测方法与基于深度学习的算法的优势和不足，进一步说明了基于深度学习的交通标志研究方法是众多学者研究的趋势。最后，该章节介绍了国内外交通标志研究的现状。

第 2 章：介绍并总结了交通标志检测与识别的相关理论概述，其中包括卷积神经网络、RNN 系列算法以及 YOLO 系列算法。另外，还涵盖了现有的交通标志检测数据集和算法性能评估指标。在详细阐述了卷积神经网络及系列目标检测算法的内容和特点之后，着重介绍了各种检测算法的优势与不足。

第 3 章：改进 YOLOv5 算法，首先介绍了 YOLOv5 网络的组成结构，随后以该网络为基础提出了一种新的改进方法。针对当前许多检测算法运行速度较快但检测精度较低的问题，对 YOLOv5 进行了主干网络、损失函数和激活函数的改进。实验结果表明，改进后的 YOLOv5 算法在精度上得到了提升，具备更好的收敛性，并提高了模型的鲁棒性。

第 4 章：轻量化 YOLOv5 算法。针对第 3 章提出的交通标志检测识别算法具有庞大的网络参数和模型尺寸，不太适用于移植到驾驶辅助系统中的问题。本章提出了一种轻量化的 YOLOv5 算法，并详细对比了三种轻量化网络，最后由实验结果可知，本章提出的轻量化算法相对于其他算法更易于移植到移动设备端，具有模型尺寸小，检测实时性高等优点。

第 5 章：总结与展望。本章总结了本文的工作和创新性，并讨论了基于深度学习的交通标志检测算法目前仍存在的难点，以及未来进一步的研究方向。

第 2 章 交通标志检测与识别相关理论概述

人工手动设计的特征提取模型对道路交通标志特征提取和拟合能力有限，在捕捉数据特征和融合方面表现较弱，导致检测效果不尽如人意。此外，传统方法的模型潜力有限，难以适应更快的检测速度和更复杂的环境。相比之下，卷积神经网络作为深度学习技术的主要推动力，在人工智能领域发挥着不可或缺的作用。随着硬件技术的不断发展，深度学习取得了突破性进展，特别是卷积神经网络在自主学习方面的能力日益增强，这使得它们能够更有效地利用大量数据提取特征并融合，以适应不同的用途。

本章首先介绍了卷积神经网络的理论基础和一些经典神经网络模型，然后对部分国内外主流的交通标志数据集进行了比较，并详细描述了本文所选用的数据集，最后介绍了深度学习模型常用的评价指标。

2.1 卷积神经网络

卷积神经网络是深度学习中最为常见的网络，被广泛应用于计算机视觉领域中，因在图像识别、物体检测、图像分割等任务上表现出色而闻名。其中以循环神经网络 (Recurrent Neural Network，RNN)[39]为代表的算法在自然语言处理领域取得了显著进展，在目标检测领域也表现出色。相较于传统方法，卷积神经网络可以直接从原始数据中学习输入与输出之间的映射关系，简化了模型的设计过程。同时在特征学习、适应性、参数共享、局部感受野和端到端的学习等方面相较于传统方法具有明显优势，在各种任务中取得了显著的性能提升。卷积神经网络由以下几部分组成。

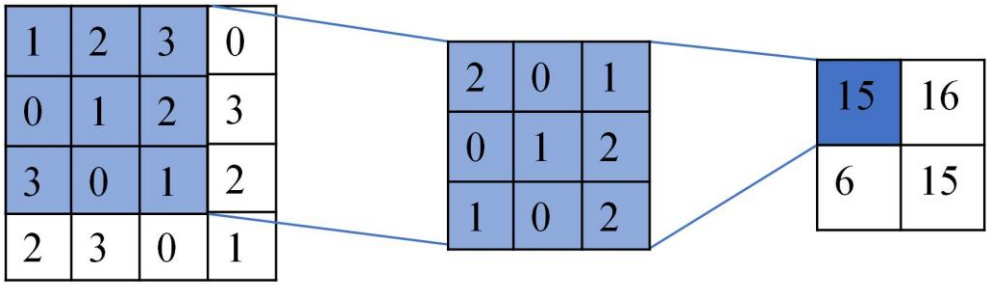
2.1.1 输入层

在卷积神经网络中，输入层是网络的第一层，负责接收原始输入数据。输入层的主要功能是将原始数据转换为网络可以处理的格式，并将其传递给下一层进

行进一步的处理。输入层的设计反映了 CNN 在处理图像数据时的特殊性质。输入层接收图像数据，通常以三维张量的形式表示，其中包括图像的高度、宽度和通道数。

2.1.2 卷积层

卷积层是网络中至关重要的部分，其作用是从输入数据中提取局部特征，可以将卷积层类比为人眼中的感受野，卷积核越大，一次性看到的信息就越多。在计算机视觉任务中，卷积操作是深度学习中的核心操作之一，通过卷积核在输入图像上滑动并进行逐元素相乘和相加，实现对图像的特征提取。这种操作具有局部连接、权值共享和平移不变性等重要特性，能够有效地减少参数数量、提高计算效率，并帮助神经网络逐步提取和组合图像的抽象特征，实现对图像的高级理解和分析。这些变种卷积操作通过灵活的卷积核设计，更好地适应不同的任务和数据结构。



(a) 输入图像　　(b) 卷积核　　(c) 输出特征图

图 2-1 卷积运算过程

卷积运算的过程如图 2-1 所示，其中输入图像如图 2-1(a)所示，特征图的大小为 4×4，卷积核大小为 3×3，这些尺寸在图中以下标数字的形式表示出来。在卷积操作中，卷积核在输入特征图上按照从左到右、从上到下的顺序进行移动，步长设定为 1，使得每次卷积核在特征图上移动 1 个单位。同时，特征图的填充被设定为 0。在第一次卷积操作时，卷积核与输入特征图对应位置的数据相乘，然后将结果相加，从而得到卷积后的结果。以第一次卷积为例，计算过程如下：1×2＋2×0＋

3×1 + 0×0 + 1×1 + 2×2 + 3×1 + 0×0 + 1×2 = 15。接着，根据步长为 1，分别向下和向右移动，共进行 4 次操作，最终得到 2×2 的输出特征图，如图 2-3(c)所示。

单通道输入的卷积过程如图 2-1 所示。多通道输入的卷积与单通道输入的卷积过程有所不同。例如，对于 3 通道输入的 RGB 图像，需要 3 个卷积核同时使用，分别对三个通道进行卷积，然后将结果求和，才能获得一层输出。若欲得到 m 层输出，则需使用 m 个 N×N×3 的卷积核。

2.1.3 激活函数

在卷积神经网络结构中，每一层的输出是通过对上一层的输入进行线性函数计算得到的。如果在神经网络中不使用激活函数，则其无法能够学习和表示更加复杂的函数关系。那么，每一层都会执行线性变换，将前一层的输出作为输入，并对其进行线性组合。因此，即使是多层网络，最终的输出仍然可以表示为对输入的线性函数。这样的网络仅能解决一些线性问题，但由于线性函数的限制，对于非线性对象则无法发挥良好作用，从而影响神经网络的性能[40,41]。激活函数是非线性的，引入激活函数理论上可以解决线性模型无法解决的非线性问题，从而提高模型的表达能力[42,43]。非线性问题也存在一些复杂的模型中，如分类和目标检测，因此选择适当的激活函数对模型至关重要。常见的激活函数有 Sigmoid、tanh、ReLu、Leaky ReLu 等。

Sigmoid 函数作为最常用的激活函数，交叉熵函数与其配套使用。Sigmoid 函数表达式如式 2-1 所示，将其输入设为x，那么无论输入值是多少都可以将输出变换到 0 至 1 之间。神经网络中用 sigmoid 函数作为激活函数，进行信号的转换，转换后的信号被传送给下一个神经元。

$$h(x)=\frac{1}{1+\exp(-x)} \tag{2-1}$$

Sigmoid 激活函数的示意图如图 2-2 所示。

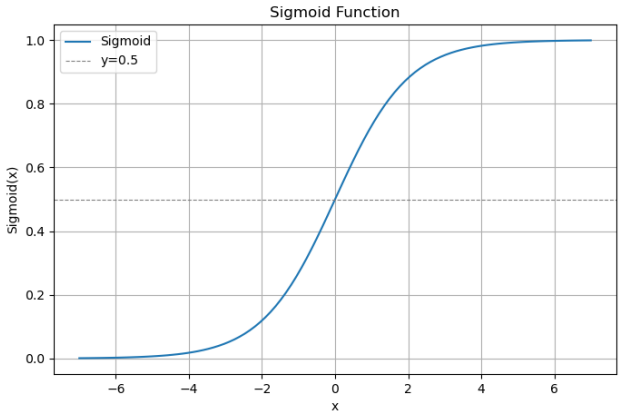


图 2-2 Sigmoid 激活函数图像

由图 2-2 可以看到，图中 Sigmoid 函数的值域为[0, 1]，当输入值无限趋近正无穷或负无穷时，输出值就会无限趋近于 0 或 1，单调递增且平滑，而且易于求导。然而，在正向和反向传播过程中都涉及到幂运算，这会导致计算量大增，从而造成逐层传递的梯度变得极小。这种情况会导致神经网络在训练过程中无法达到收敛。因此，在神经网络的反向传播过程中，梯度消失现象很容易出现[44]。

与 Sigmoid 函数类似，tanh 函数也能将其输入压缩转换到区间 (-1,1)上，tan 函数表达式如式 2-2 所示：

$$\tanh(x)=\frac{1-\exp(-2x)}{1+\exp(-2x)} \tag{2-2}$$

tanh 激活函数的示意图如图 2-3 所示。

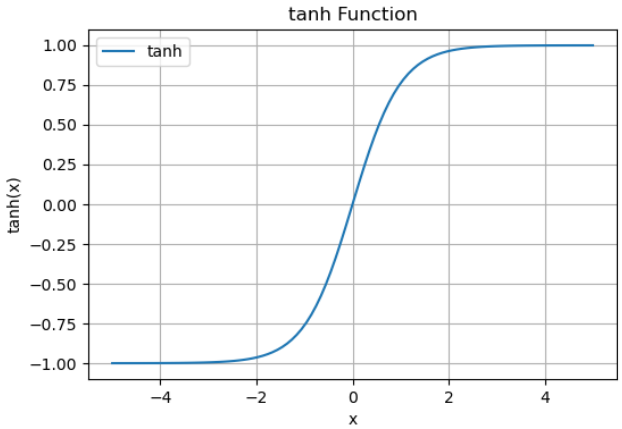


图 2-3 tanh 激活函数图像

由图 2-3 可以看到，该函数曲线形状类似于 Sigmoid 函数，但 tanh 函数的输出范围更广，该函数将任意实数映射到范围在 -1 到 1 之间的值。与 Sigmoid 函数类似，tanh 函数同样是一种 S 形曲线，但它的输出范围更广泛，涵盖了负数。具体来说，当输入为负无穷时，tanh 趋近于-1，当输入为零时，输出为零，当输入为正无穷时，tanh 趋近于 1。

tanh 函数在神经网络中广泛用于激活函数，因为它在非线性转换方面表现出良好的特性，能够帮助神经网络模型学习非线性关系。与 Sigmoid 函数相比，tanh 函数的输出值均值接近于 0，这有助于缓解梯度消失的问题，并且 tanh 函数的导数在其定义域内范围都在 0 到 1 之间，这有利于梯度下降的稳定性。

ReLu 激活函数即分段线性函数，具有稀疏性，可以使稀疏后的模型更好的挖掘相关特征，拟合训练数据，函数表达式如式 2-3 所示:

$$\mathrm{Re}\,LU(x) = Max(0, x) \tag{2-3}$$

ReLu 激活函数的示意图如图 2-4 所示。

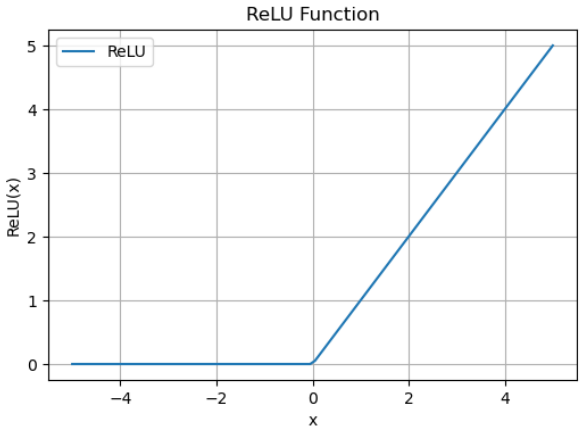


图 2-4 ReLu 激活函数图像

从图 2-4 中可以看出，这个函数可以看作是一个分段函数。当输入值大于零时，输出值等于输入值，梯度恒定为 1，计算复杂度低，解决了 Sigmoid 函数中输入值过大导致梯度消失的问题。当输入值小于或等于零时，输出值为零，这样可以使得神经元不会被激活，进而使网络数据变得稀疏，并且减小了参数之间的相互依存关系，有效缓解了网络出现过拟合的问题。然而，当输入值为零时，激活函数

的值也为零，梯度为零，这意味着 ReLU 函数会“死掉”，因此存在一部分神经元永远不会得到更新。

2.1.4 池化层

池化层 (Pooling Layer) 是深度学习中卷积神经网络的一种常见层次。它通过下采样、减少参数量、增强特征不变性和提取主要特征等方式，有效地帮助模型提高计算效率、降低过拟合风险。在池化过程中，通常会使用池化操作来获取特征图中的最显著特征或对特征进行平滑处理，具有一定程度的平移不变性和空间不变性，即对输入特征图进行小范围的平移或空间变换，池化操作的结果仍然保持相对稳定。常用的池化操作有最大池化 (Max Pooling) 和平均池化 (Average Pooling)。最大池化即每个池化窗口中选择最大值作为该区域的代表特征，通过这种方式，最大池化能够有效地减少特征图的维度，保留图像中最显著的特征，例如边缘和纹理。平均池化是在每个池化窗口中计算特征的平均值作为该区域的代表特征。与最大池化相比，平均池化更加平滑，能够保留更多的图像细节信息，但相应地会降低特征的显著性。

图 2-5 最大池化操作

图 2-5 为最大池化操作，可以看到尺寸为 4×4 的特征图经过尺寸为 2×2 的最大值池化后，特征图尺寸降为 2×2，也就是原特征图的一半，并且保留输入区域中最显著的特征，减少计算负担并提高模型的鲁棒性。最大池化通常与卷积层交替使用，卷积层的作用是负责提取特征，而最大池化层则负责降低空间维度，形成卷积池化层的堆叠，构建卷积神经网络的主体结构，这种结构有助于构建具有层次结构的特征表示，使模型更具有表达能力。在平均池化中，网络会将每个池化窗口中的所有像素值取平均值作为该窗口的输出值。这样做的目的是减少特征图的

空间尺寸，降低计算复杂度，并且可以提取图像中的整体特征。通过平均池化，网络能够保留更多的图像细节，有助于减轻过拟合现象，同时也有助于使模型对于输入图像中的平移变化更具有鲁棒性。

2.1.5 全连接层

全连接层 (Fully Connected Layer, FCL)，也称为密集连接层或仿射层，是神经网络中常见的一种层级结构。全连接层的每个节点都与前一层的所有节点相连。对于前一层提取的特征进行了全面的分析，也就是说，从前一层计算得到的特征空间被映射到样本标签空间，从而减少了特征位置对分类结果的影响，并提高了整个网络的鲁棒性。全连接层通常用于神经网络的最后几层，用于对输入数据进行最终的分类或回归预测。通过全连接层，神经网络能够学习到输入数据中的复杂特征，并将其映射到输出结果。全连接层如图 2-6 所示。

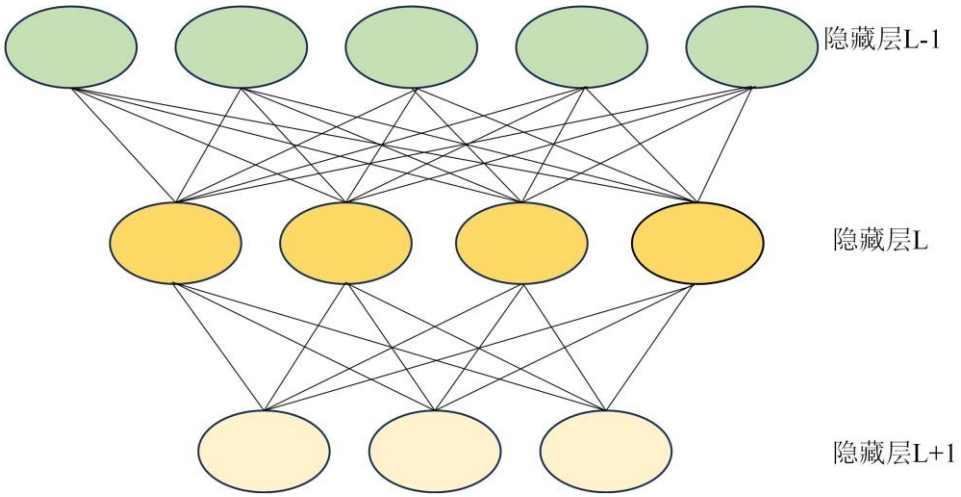


图 2-6 全连接层示意图

2.2 基于深度学习的典型目标检测算法

2.2.1 R-CNN 系列算法

R-CNN[45]作为开创性的目标检测算法之一，采用了两阶段的方法。首先，使用选择性搜索等算法生成一系列候选区域。这些候选区域经过筛选，排除了容易被当作背景的区域，并被调整为固定大小，作为卷积神经网络的输入。在第二阶段，对于每个候选区域，R-CNN 采用卷积神经网络对其进行特征提取和分类。具

体地，R-CNN 使用一个预训练好的 CNN 模型，将每个候选区域裁剪并调整为固定大小的图像，然后将其输入到 CNN 中，最后使用支持向量机 (SVM) 对每个候选区域进行分类，从而实现了目标的二分类。然而，R-CNN 也存在着计算复杂性高和训练时间长的缺点，后续的改进算法如 Fast R-CNN 和 Faster R-CNN 在此基础上进行了优化，进而加速了目标检测的速度，为该领域的发展奠定了基础。R-CNN 网络结构如图 2-7 所示。

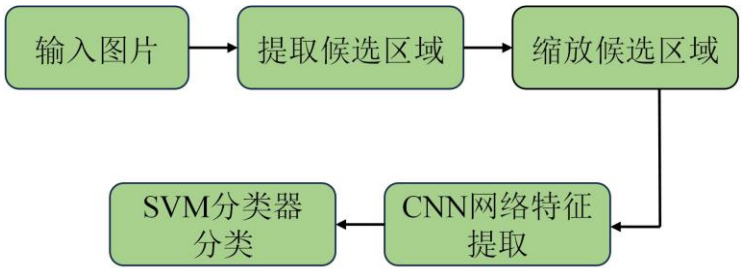


图 2-7 R-CNN 算法的网络结构图

尽管 R-CNN 在目标检测中取得了一定的成效，但在实际应用中，该算法整体上需要在每个候选框上独立运行卷积神经网络，导致计算成本高，并且训练和测试阶段需要多个步骤，增加了整体复杂度和时间开销。为应对这些挑战，Fast R-CNN 应运而生。

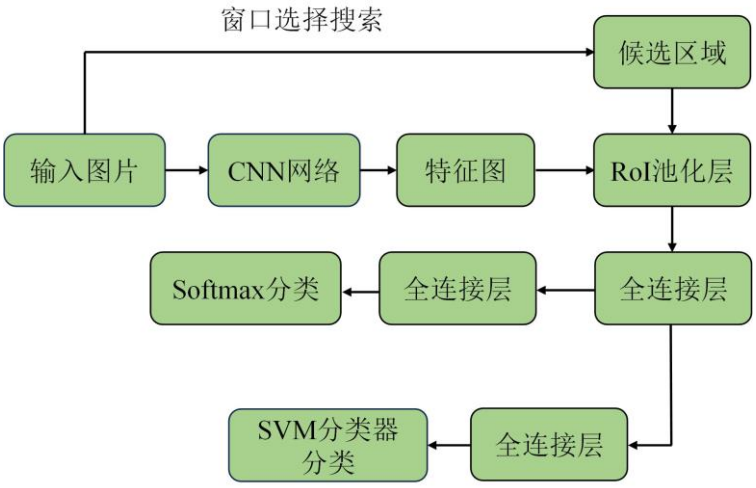


图 2-8 Fast R-CNN 算法的网络结构图

Fast R-CNN[46]特征提取网络是 VGG-16 (Visual Geometry Group 16, VGG-16) 算法，其兼顾了 SPP (Spatial Pyramid Pooling, SPP) 网络结构的优点。Fast R-CNN 算法的网

络结构如图 2-8 所示，它相较于其前身 R-CNN 和其改进版本 SPP-Net，引入了候选区域池化层，候选区域池化层将提议区域映射到固定大小的特征图上。这样，不同大小和长宽比的提议区域可以在相同大小的特征图上进行处理，从而减少了特征提取和目标检测的过程。因此，Fast R-CNN 在速度和性能方面取得了显著的改进，成为目标检测领域的重要算法之一，但针对实时性要求较高的目标检测任务，Fast R-CNN 的速度仍无法满足要求，因此学者们在此基础上提出了 Faster R-CNN。

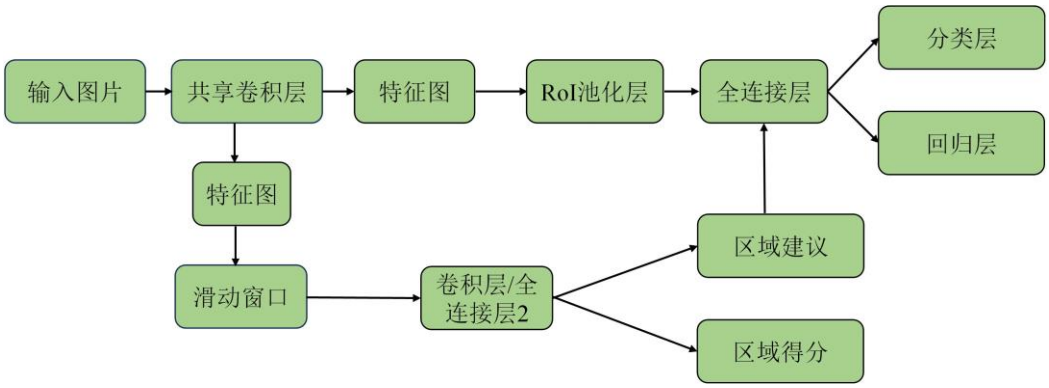


图 2-9 Faster R-CNN 算法的网络结构图

Faster R-CNN[47]是一种经典的基于深度学习的目标检测算法，算法的网络结构如图 2-9 所示，引入的区域提议网络 (Region Proposal Network，RPN)，RPN 通过滑动窗口在输入图像上生成各种大小和宽高比的锚框，然后运用分类网络判断每个锚框是否包含目标及位置，再利用回归网络来校正锚框的位置，从而提高了检测速度和准确性。Faster R-CNN 共享一个卷积神经网络(通常是用于图像分类的预训练网络，如 VGG、ResNet 等)来提取图像特征。这样，生成的锚框和实际目标框之间的特征表示可以共享，从而加速训练和提高检测性能。在 RPN 生成候选框后，这些框会被送入区域兴趣池化层，将不同大小的感兴趣区域 (Region of Interest, RoI) 映射到固定大小的特征图上。这使得具有不同尺寸的目标能够被相同大小的特征图处理，实现了位置不变性。经过 RoI 池化操作，生成的固定大小的 RoI 通过两个全连接层，分别用于目标分类和边界框位置的回归。分类层采用 Softmax 函数来预测 RoI 中是否包含目标，而回归层是用于调整预测框的位置。最后，Faster R-CNN

的损失函数由三部分组成：RPN 的分类和位置损失、目标检测网络的分类损失和位置损失。这些损失函数共同构成了模型的训练目标。

2.2.2 YOLO (You Only Look Once) 系列

YOLO 是一系列单阶段目标检测算法，包括 YOLO[48]、YOLO9000[49]、YOLOv3[50]、YOLOv4[51]和 YOLOv5[52]。这些算法通过将图像分为网格并在每个网格上直接进行目标检测，实现了实时目标检测的能力，并且 YOLOv5 在速度和精度上均有显著提升。表 2-1 展示了 YOLO 目标检测算法演变的摘要。通过持续不断地演进并整合最新技术，YOLO 始终在实时目标检测性能方面不断突破界限，使其成为各种计算机视觉应用中的强大工具。

表 2-1 YOLO 目标检测算法演化概述

算法	年份	主干网络	层数
YOLO	2016	Darknet-19	34
YOLO9000	2017	Darknet-19	53
YOLOv3	2018	Darknet-53	106
YOLOv4	2020	CSPDarknet53	215
YOLOv5	2020	CSPDarknet53	168

(1) YOLO

YOLO (YOLOv1) 是原始的 YOLO 架构，具有 34 个卷积层和 2 个全连接层。YOLO 目标检测算法将图像分割为较小的网格，每个网格分配了一些预先计算的锚定框，网格中的每个单元格负责检测出现或其部分出现在该特定单元格中的对象，前提是该单元格包含对象的中心。每个单元格为任何出现在该单元格中的对象赋予一个置信度分数，并预测该对象所属的类别。YOLO 将目标框的定位问题转化为回归问题来处理，实现了真正的端到端检测，速度也得到很大提升，可以用于视频监控等实时检测任务。

YOLOv1 对物体的检测过程如图 2-10 所示：

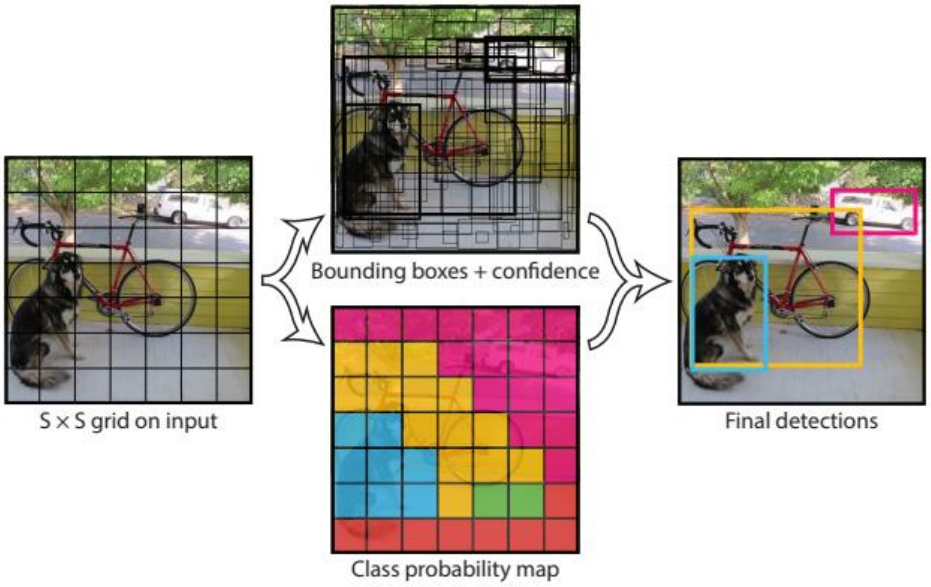


图 2-10 YOLO 检测流程

YOLOv1 算法的具体检测步骤如下：

(1) 首先将输入图像调整为固定大小，并进行归一化处理，再将图像分割成 S×S 个网格，每个网格负责检测图像中的目标，YOLOv1 通常采用 7×7 的网格。

(2) 每个网格预测 B 个候选框。每个候选框由五个预测值组成：x、y(候选框中心相对于当前网格的偏移量)、宽度、高度以及目标的置信度，同时每个候选框还对预测目标属于不同类别的概率进行预测。

(3) 将实际目标的位置与预测的位置进行比较，计算位置损失(如坐标误差、尺寸误差等)和类别损失(如交叉熵损失)。最后，将位置损失和类别损失结合起来作为最终的损失函数。

YOLO 使用均方误差 (MSE) 来计算预测值和真实框之间的差异，损失函数考虑了位置、置信度和分类误差，损失函数表达式如式 (2-4) 所示：

$$Loss = L_{box} + L_{obj} + L_{cls} \tag{2-4}$$

(2) YOLOv2

YOLO9000 也称为 YOLOv2，相比 YOLO 有许多改进。YOLO9000 在所有的卷积层中使用了批量归一化。通过使用批量归一化进行正则化，作者移去了 Dropout。Dropout 是一种正则化技术，在训练过程中会随机丢弃一些神经元。YOLO9000 还

使用了更高的 448×448 输入分辨率，而不是 YOLO 的 256×256。此外，YOLOv2 中使用 K-means 聚类来找到正确的锚定框集合。

与 YOLOv1 相比，YOLOv2 的主要不同之处在于使用了更深的 Darknet-19 网络作为主干，并且在网络结构上进行了优化，提升了特征提取的效果。引入了全局平均池化层，对整个特征图进行池化，以获取整体上下文信息，用于物体类别的预测[53]。为了降低计算负担，YOLOv2 增加了 1×1 卷积操作，用于压缩特征图之间的通道数，从而助于减少网络的参数量和运算量。最后使用 Anchor Boxes 以及多尺度训练策略等，通过在不同尺度上进行检测，使得网络更好地适应不同大小的目标，提高了目标检测的鲁棒性。这些改进使得 YOLOv2 在目标检测任务中表现出更高的准确性和更快的速度。

其中 Darknet-19 网络结构如图 2-11 所示， Darknet-19 使用最大池化来进行下采样，降低特征图的维度，为了进行上采样，它使用反卷积操作，以增加特征图的尺寸。分类部分使用全连接层，全连接层将前面卷积部分提取的特征转化为最终的类别概率分布，以便网络可以对输入进行分类。在 YOLOv2 中，Darknet-19 引入了跳跃连接 (skip connections)，将底层和中间层的特征图直接连接，有助于提高网络对不同尺度的目标的检测能力。为了更精准地适应各种物体尺寸，YOLOv2 目标检测算法汲取了 RPN (Region Proposal Network, RPN) 的思想，通过 K-means 聚类方法对先验框进行分类先验，从而计算锚框的尺寸。在目标定位的准确性上，由于每个单元格可以预测 5 个边界框，而不是 YOLOv1 的 2 个边界框，因此 YOLOv2 实现了更为准确的目标定位。

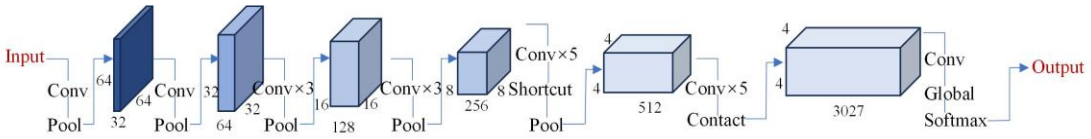


图 2-11 Darknet-19 网络结构

(3) YOLOv3

YOLOv3 在 YOLOv2 的基础上进行了扩展，使用了自定义形式的 Darknet，自定义的 Darknet 骨干网络被称为 Darknet--53。YOLOv3 总共有 106 层，使得它的速度稍慢。YOLOv3 在 3 个不同尺度上预测边界框，类似于特征金字塔网络。使用特征金字塔网络，YOLOv3 能够比 YOLO9000 更好地检测多尺度对象。

YOLOv3 的网络结构主要包括 Darknet 网络、多尺度预测和多标签预测等三个模块，这些模块共同构成了该目标检测算法的核心，YOLOv3 的网络结构如图 2-12 所示。

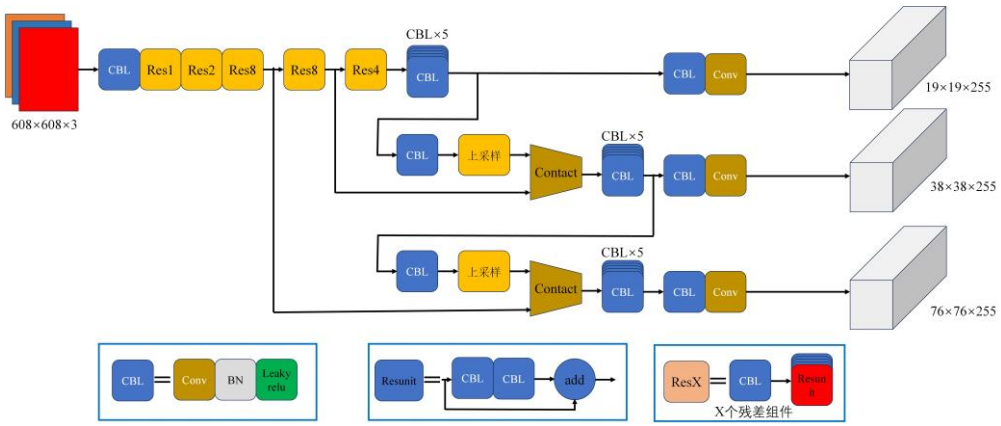


图 2-12 YOLOv3 网络结构

Darknet-53 利用残差连接和跳跃连接的结构，有助于信息的传递和梯度的流动，解决了梯度消失和梯度爆炸的问题。此外，Darknet-53 通过分阶段的特征提取，使得网络能够学习到不同层次的语义特征，从而提高了对不同尺度物体的检测能力。同时，Darknet-53 还使用了批量归一化技术，加速了网络的训练过程。借鉴特征金字塔网络 (Feature Pyramid Network, FPN) 的思想，YOLOv3 通过三次下采样操作获得不同尺度的特征图，然后进行上采样和特征融合，得到最终的预测结果，实现了 19×19、38×38 和 76×76 的预测尺度，每个尺度的特征图通过相应的检测层进行处理，负责预测目标的置信度、类别和位置，分别用于检测大目标、中目标和小目标。

YOLOv3 作为 YOLO 系列的新一代算法，采用了多标签预测的策略，即每个边界框可以同时预测多个目标类别，实现了多尺度目标预测。每个边界框的类别

预测通过逐通道的 Sigmoid 激活函数实现，使得每个类别的预测是独立的，从而允许每个边界框同时被分配多个类别。然而，为了达到这些优点，算法在速度方面做出了一定的牺牲。

(4) YOLOv4

YOLOv4对网络的主干和颈部结构都作了改进，融合了多处改进来提升检测效果，相比 YOLOv3，其在速度上有很大提升，YOLOv4 的结构如图 2-13 所示。

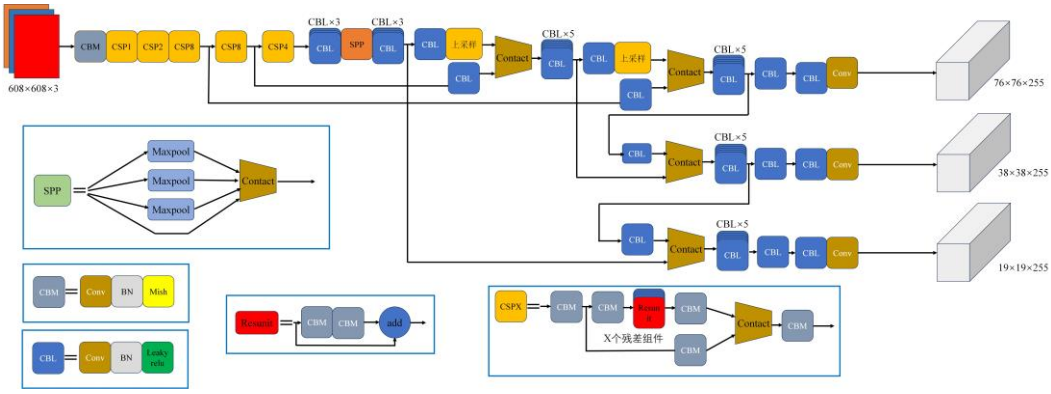


图 2-13 YOLOv4 的结构图

YOLOv4 引入了更强大的主干网络，采用了 CSPDarknet53 作为特征提取网络，相比 YOLOv3 的 Darknet-53，CSPDarknet53 结合了金字塔池化结构 (SPP) 和路径聚合网络 (PANet)，有效地减少了推理信息，提高了信息传递效率和学习能力。在网络结构的改进方面，首先，它引入了更多的卷积核，这意味着在网络中使用了更多的滤波器来捕获图像中的不同特征。其次，YOLOV4 通过改变卷积核的大小，使得网络能够适应不同尺度的特征，从而更好地检测到不同大小的目标。此外，YOLOv4还增加了网络的深度，通过增加更多的网络层来进一步提取图像中的抽象特征。这样的深度增加使得模型能够更好地理解图像中的复杂信息，从而提高了对目标的检测和识别准确度。在损失函数方面，引入了焦点损失(GIoU 损失、DIoU 损失和 CIoU 损失)，用于更精确地衡量边界框之间的相似性，改进了目标定位的准确性。在激活函数方面，使用 Mish 激活函数代替了 YOLOv3 中的 Leaky ReLU 激活函数，大量实验证明 Mish 激活函数的梯度比 Leaky ReLU 激活函数更平

滑，在一些场景下表现更好[54-56]。为了提高模型的泛化能力，引入了更丰富的数据增强策略Mosaic，随机选择4张图片，分别对这四张图片进行放大、缩小、裁剪等操作，选定一个中心点后以随机排布的方法进行组合，该方法大大增加了随机样本的多样性，在一定程度上提高了网络的鲁棒性，以增加训练数据的多样性。最后引入了加权卷积和 SAM (Spatial Attention Module, SAM) 模块，用于加强模型对空间信息的关注和学习。

(5) YOLOv5

YOLOv5 一共发布了 4 个不同的模型版本，分别为 YOLOv5s、YOLOv5m、YOLOv5l、YOLOv5x。其中 YOLOv5s 是 YOLOv5 系列中规模最小的模型，具有相对较少的参数和更快的推理速度，适用于资源有限的环境，它在速度和准确性之间取得了良好的平衡。YOLOv5m 是在 YOLOv5s 的基础上进行了一些扩展和改进，旨在提供更好的性能。它具有更大的模型规模和更好的检测性能，但相对于其它更大的模型，它的计算需求较小。YOLOv5l 是一个更大的模型，它适用于对模型精确度和实时性要求极高的场景，但需要更强大的计算资源。YOLOv5x 是YOLOv5系列中的最大模型，通常具有最多的参数和最高的准确性，它适用于需要最先进性能的任务，但也需要更大的计算资源。考虑到本文研究对象交通标志识别是汽车智能辅助驾驶重要组成部分，因此要求更高的检测速度和较低的计算资源，因此本文选取 YOLOv5s 作为网络训练模型。

YOLOv5s 模型的骨干网络模块包括来自 Darknet 的编码模块，其主要功能是从输入图像中有效地提取图像的语义特征。此外，特征融合模块包括两个子模块：特征金字塔网络和路径聚合网络。这些子模块将不同尺度的特征图进行融合，创建一个富含语义信息的综合特征图。然后，融合后的特征图传递到预测模块，进行目标检测和分类任务。YOLOv5s 的特征编码模块对输入图像进行三次下采样操作，特征编码模块通常采用步幅为2的卷积操作或最大池化操作来实现图像尺寸的

减半。每次下采样操作都会使特征图的宽度和高度减少一半，同时将特征图的深度、通道数增加，以保留更多的语义信息。通过三次下采样操作，特征编码模块能够逐渐缩小输入图像的尺寸，同时提取出不同尺度的特征表示。这种多尺度的特征表示能够捕获图像中不同尺度的目标信息，使得目标检测算法能够更好地适应不同大小和尺度的目标，并提高检测的准确性和鲁棒性。以 1280 × 1280 的初始图像为例，通过连续进行 8 倍、16 倍和 32 倍的三次下采样操作，可以生成三个具有不同分辨率的特征图。这些特征图随后被送入特征融合模块，该模块的作用是将抽象的语义信息与浅层的特征细节相结合，从而合并和重建这三个不同分辨率的特征图。图像预测模块由三个预测层组成，负责预测输入图像中每个对象的边界框和类别概率。

2.3 交通标志数据集

2.3.1 交通标志简介

交通标志是道路交通管理的关键组成部分，通过视觉元素和文字信息向驾驶员和行人传达重要交通信息，以确保道路通行的有序性和提升交通安全。这些标志通常使用图形、符号和颜色来传达信息，而不仅仅依赖于文字。广泛应用的国内外交通标志数据集见表 2-2。随着交通标志统一化的趋势，人们能够清晰地获取道路行驶信息，从而规范行车行为，确保交通畅通和安全。

表 2-2 公开交通标志数据集信息

数据集名称	类别数	图片数量	用途	图片大小
中国 TT100k[57]	221	10000	检测和识别	2048×2048
德国 GTSDB[58]	3	900	检测	1360×800
中国 CCTSDB[59]	43	10000+	识别	15×15-250×250
KITTI [60]	8	7481	检测和识别	1200×300

交通标志的主要功能包括：

信息传递： 交通标志通过直观的图形和文字传递各种交通信息，包括道路的方向、速限、危险警示、禁止通行、道路类型等，帮助驾驶员正确理解和遵守交通规则。

安全警示： 交通标志用于提醒驾驶员和行人即将面临的道路状况，如急转弯、施工区域、行人横穿等，以减少交通事故的发生，提高道路使用者的安全性。

规范行为： 交通标志规范了道路使用者的行为，例如停车标志、禁止掉头标志等，确保驾驶员在道路上遵循一致的交通规则，有助于维护交通秩序。

引导导航： 一些交通标志用于指引驾驶员前往特定地点，如道路编号、指示牌等，提供导航支持，方便行车和旅行。

管理交通流： 通过交通标志的设置，可以合理引导交通流，提高道路的通行效率，减缓交通拥堵，优化道路网络的使用。

禁令标志： 明确规定在某些区域或情境下不允许执行特定行为，例如禁止停车、禁止通行等。

指示标志： 提供有关服务、设施或地点的信息，例如加油站、医院、酒店等。

信息标志： 提供额外信息，如距离、方向、道路条件等。

这些标志在国家和地区之间可能有所不同，因此在驾驶或行走时，理解并遵守当地交通标志是至关重要的。

2.3.2 经典交通标志数据集

(1) 德国 GTSDB 数据集

GTSDB (German Traffic Sign Detection Benchmark，GTSDB) 数据集由德国莱布尼茨大学波恩分校 (University of Bonn) 的计算机视觉实验室 (Computer Vision Lab) 创建和维护。GTSDB 数据集包含数千张车载摄像头拍摄的道路图像，这些图像的分辨率和环境条件也各不相同，使得交通标志检测算法需要具备一定的鲁棒性，该数

据集能够全面地测试算法在不同场景和标志种类上的性能，但 GTSDB 数据集的规模相对较小，部分数据集如图 2-14 所示。

图 2-14 GTSDB 数据集部分数据

(2) TT100k 数据集

TT100K (Tsinghua-Tencent 100K) 由清华大学和腾讯公司共同对行车记录仪拍摄的视频进行裁剪制作完成的，包含 10 万多张真实驾驶场景的图像，涵盖了 20 万个交通标志实例，分辨率高达 2048×2048。大规模的数据集不仅提供了充足的训练样本，而且通过详细的标注，包括标志的位置、类别和形状等信息，为深度学习模型的训练和评估提供了有力支持。TT100K 覆盖了多种交通标志类别，如限速标志、停车标志、禁令标志、警告标志等，并且这些标志类图像涵盖了各种天气和光照条件，以及不同的驾驶场景，与其他数据集的主要区别在于其包含大量小尺度的交通标志目标以及丰富多样的交通标志类别。这使得 TT100k 数据集成为一个理想的基准，能够全面评估交通标志检测算法的性能。部分数据集图片和类别对应的标签如图 2-15 和图 2-16 所示。

图 2-15TT100k 数据集部分数据

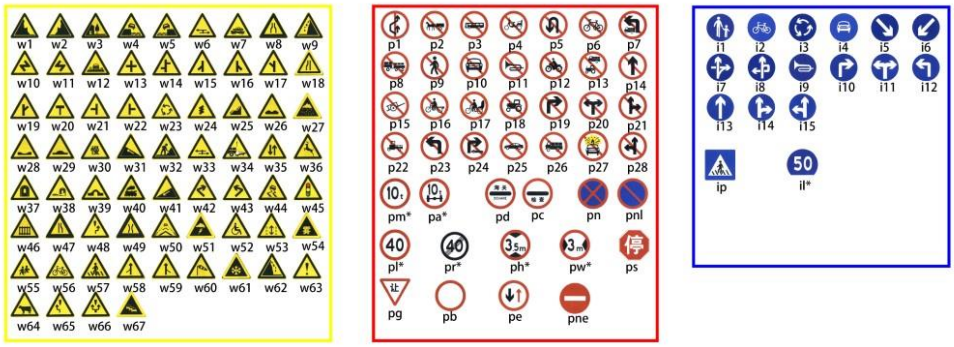


图 2-16 TT100k 数据集标注与名称对应图

2.4 目标检测评价指标

本文采用精确率 P (Precision, P)、召回率 R (Recall，R)、平均精度均值 *m*AP (mean Average Precision，*m*AP) 以及检测速率每秒帧数 FPS (Frame Per Second，FPS) 作为交通标志目标识别算法性能的评价指标。前三个指标是基于公式 (2-5)至公式 (2-7)计算的，其中 TP、FP 和 FN 分别代表真正例样本、假正例样本和假负例样本。在公式 (2-8)中，AP 表示平均准确率，P(*r*) 描述了召回率为 R 时的精度，它们被用来评估 YOLOv5s 的检测准确性，后一个指标 FPS 用于评估检测速度。

$$\text{Precision}=\frac{\text{True Positive}}{\text{True Positive}+\text{False Positive}} \tag{2-5}$$

$$\text{Recall}=\frac{\text{True Positive}}{\text{True Positive}+\text{False Negative}} \tag{2-6}$$

$$\mathrm{AP}=\int_0^1 \mathrm{P}(r)dr \tag{2-7}$$

$$m\mathrm{AP}=\frac{\sum_{q-1}^{Q}\mathrm{AP}(q)}{\mathrm{Q}} \tag{2-8}$$

式中： Q 为类别个数，AP(q) 为不同类别的平均精度。

在深度学习中，衡量模型复杂度的两个关键指标是参数数量 (Parameters) 和理论计算量 (FLOPs)。这两个指标在模型设计和选择中被广泛用于平衡性能和计算成本，但完整的模型评估还需要考虑其他因素，如泛化能力、训练时间和实际推理速度。模型中的参数数量包括卷积参数、全连接参数、BatchNorm 参数、嵌入参数等，模型的最终大小直接受参数数量的影响。

理论计算量则是指模型在前向传播时执行的浮点运算总数。通常大模型的单位是 G (GFLOPs，表示十亿次浮点计算)，它用作算法复杂性的度量标准。更多的参数通常表示更大的模型容量，可以更好地适应训练数据，而较低的理论计算量则意味着模型更轻量，适用于资源受限的环境。计算公式如式 (2-9) 所示。

$$\mathrm{GFLOPs}=(2C_i\mathrm{K}^2\text{-}1)\mathrm{HW}C_0 \tag{2-9}$$

其中，C_i 和 C_0 表示输入和输出的通道数量，H 和 W 表示特征图的尺寸，K 表示卷积核的尺寸。

FPS 是衡量系统处理或发现图像数量的速度的指标。计算公式如式 (2-10) 所示：

$$\mathrm{FPS}=\frac{\mathrm{N}}{\mathrm{T}} \tag{2-10}$$

其中，N 代表检测到的图片总数，T 代表处理所有图片所需的总时间。

2.5 本章小结

本章介绍并总结了交通标志检测与识别相关的理论概述，包括卷积神经网络、RNN 系列算法、YOLO 系列算法，以及现有的交通标志检测数据集和算法性能评估指标。此外，还详细阐述了卷积神经网络及系列目标检测算法的内容和特点，重点介绍了各种检测算法的优势与不足。

第 3 章 基于 CBAM 和改进的 YOLOv5 交通标志检测与识别

随着人工智能 (AI) 技术的发展，智能辅助驾驶系统成为科研人员研究的热点。传统基于特征提取的交通标志检测方法难以处理复杂多变的交通场景，智能驾驶需要能够在各种天气条件、光照条件、道路情况和交通标志布局下可靠地检测交通标志。基于深度学习的方法可以从大量数据中学习到更加丰富和抽象的特征，能够更好地适应复杂多变的交通场景。

针对上述传统检测方法存在的问题，本章介绍了一种改进的YOLOv5方法。该方法通过使用 SIoU 损失函数替换 YOLOv5 模型中的 GIoU 损失函数来优化训练模型。同时，在主干网络的 CSP1_3 模块中引入卷积注意力模块，形成新的 CSP1_3CBAM 模块，以有效增强特征表示，从而提高模型性能。此外，引入 ACONC 作为 YOLOv5 的激活函数，使得模型具备更好的收敛性，并提升模型的鲁棒性。

3.1 网络架构分析

YOLO 系列网络结构是一种经典的单阶段架构，主要由三个组件组成：骨干网络 (backbone)、颈部网络 (neck) 和检测头部 (head)。骨干网络主要由 BottleneckCSP 模块组成，用于提取特征信息。颈部网络采用了改进的特征金字塔网络 (FPN) 结构，用于融合提取的特征图。检测头部是模型的最终检测部分，用于预测检测到的对象类别和位置。这种结构在 YOLOv3 算法中逐渐形成，并且后续的 YOLOv4 在 YOLOv3 的基础上对各个部分进行改进。而 YOLOv5 和 YOLOv4 的结构相似，但模型大小仅为 YOLOv4 的九分之一，具有简洁、易于部署的优点，因此在目标检测场景中得到了广泛应用。YOLOv5 在 Input 方面增加了 Mosaic 数据增强和自适应锚框计算，在 Backbone 上增加了 Focus 模块和 CSP 模块，Neck 方面采用了 FPN+PAN 的架构，Head 则采用 CloU_Loss 作为 Bounding box 的损失函数。YOLOv5 网络结构如图 3-1 所示。

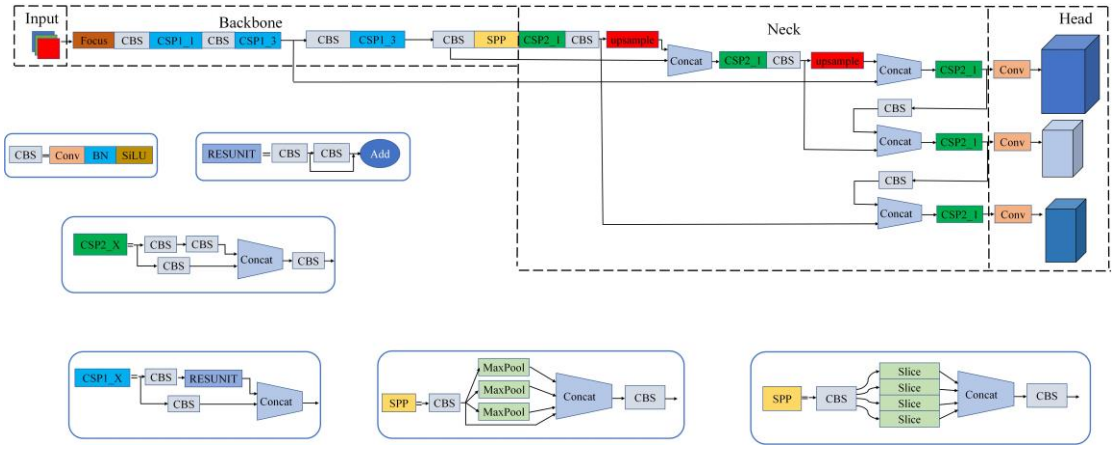


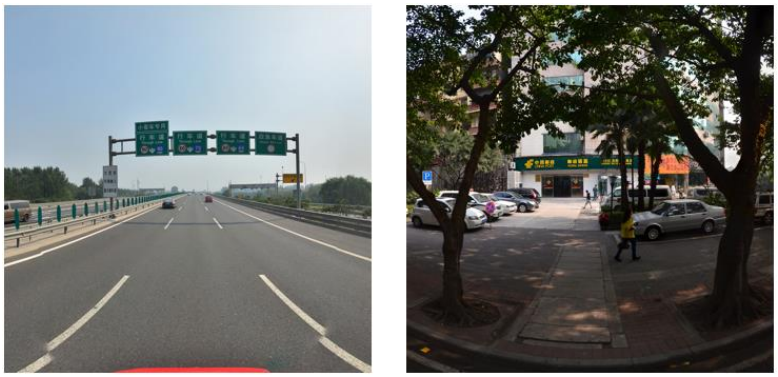
图 3-1 YOLOv5 网络结构图

3.1.1 Input 部分

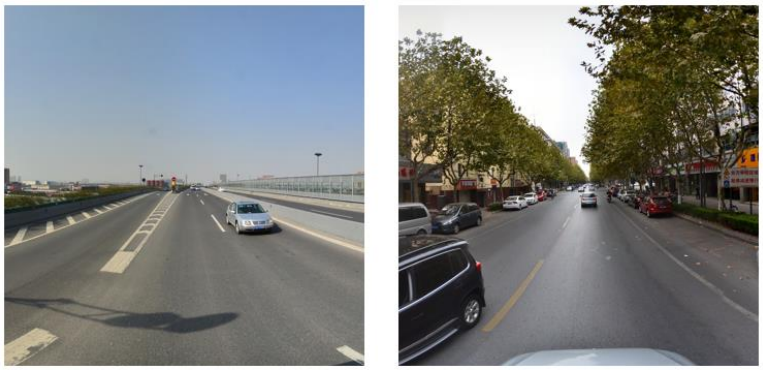
(1) Mosaic 数据增强

YOLOv5 在输入端采用了 Mosaic 数据增强算法将多张图片拼接成一张大图像，在拼接过程中随机调整目标的位置和大小，从而增强模型对目标在不同尺度和位置的适应能力。首先，从训练数据集中随机挑选出四张图片，然后将它们拼接成一张大图像，有助于模型学习到目标在不同区域的分布情况。接着，在拼接过程中，通过随机调整目标的位置和大小来模拟不同场景下的目标位置和尺度变化，包括随机选择位置和调整目标尺度。在调整图像时，需要相应地调整目标的标签信息，以适应图像和目标位置的变化。最后，将拼接后的大图像和相应的标签信息作为模型的训练样本，模型通过学习这些样本逐渐提高对不同尺度和位置目标的识别能力。

Mosaic 数据增强具有提供更多、更丰富的样本信息、增强模型鲁棒性、节省计算资源等优点，有助于提高目标检测模型的性能和效率。具体操作如图 3-2 和图 3-3。



(a) 高速与街景范例 1



(b) 高速与街景范例 2

图 3-2 未经 Mosaic 数据增强的原始图片

图 3-3 经 Mosaic 数据增强后的图片

(2) 自适应锚框计算

在训练过程中，网络需要基于预先设定的初始锚框生成预测框，并利用 K-means 聚类方法获取数据集的训练锚框。通过根据训练数据集中目标的统计特征动态生成或调整锚框，以更好地适应不同尺度和长宽比的目标，从而提高检测性能和准确性。这种方法能够显著改善处理具有较大尺度变化的目标时的检测效果。

(3) 自适应图片缩放

自适应图片缩放指的是在训练和推理阶段根据网络的输入要求和图像的原始尺寸动态调整图像大小，而图片的缩放会使其变形失真，因此采用自适应灰度填充，即将图片按照原尺寸比例缩放，其余填充灰度图像。

3.1.2 Backbone 部分

如图 3-4 所示，YOLOv5 主干网的第 1 层为 Focus 层，其主要作用是对输入的图片进行切片操作，并进行等间隔采样, Focus 层将输入的特征图分成四个子图，然后将这四个子图叠加起来，形成一个更小的特征图，以减少计算成本同时保留更多的空间信息。第 2 层为 CBL 层，由卷积层 (Conv)、归一化层 (BN) 和激活函数层 (Leaky ReLU) 组成，是 Backbone 网络的基础模块，用于对输入的特征图进行卷积、批标准化 (Batch Normalization) 和 SiLU 激活函数操作。此外，在 YOLOv4 中，仅设计了一种 CSP 结构，而在 YOLOv5 中则设计了两种 CSP 结构，其主要特点是一条支路进行卷积操作，另一条支路对上一条支路进行卷积操作后的结果进行拼接。这两种 CSP 结构分别位于主干网络和 Neck 中。

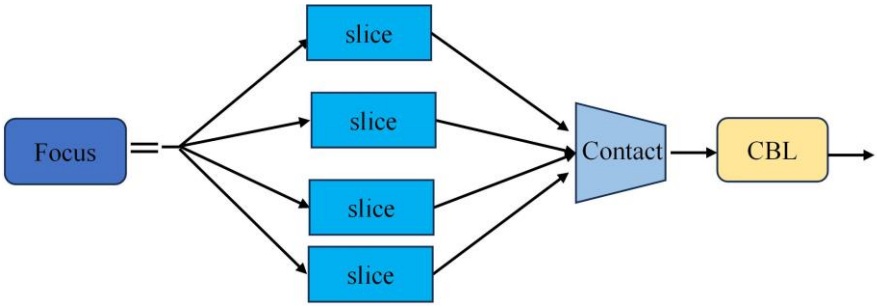


图 3-4 Focus 结构图

3.1.3 Neck 部分

在骨干网络之后，YOLOv5 引入了 FPN[61]作为颈部结构。FPN 在不同尺度上自上而下地构建高级语义特征图，用于融合不同层次的特征。与此同时，单路径聚合网络自下而上地弥补并加强了空间信息和语义信息，从而改善了经过多层网络后底层目标的信息模糊问题，使得网络模型更好地适应不同尺度的目标。颈部结构的主要功能是生成一个特征金字塔，并且可以通过该金字塔将模型底层特征进行上采样操作后与深层特征进行融合。FPN 的引入有助于提高目标检测的多尺度性能。

3.1.4 Head 部分

YOLOv5 的 Head 结构是检测任务的核心部分，采用了 YOLOv3 的思想，通过多层次的卷积操作来预测目标的边界框、置信度和类别信息。为了改进预

测分支的表达能力，YOLOv5 引入了 Panet(Path Aggregation Network，Panet) 结构[62]。同时，YOLOv5 使用了 sigmoid 函数输出目标的置信度和类别概率，以及坐标回归用于目标框的定位。

3.2 改进的 YOLOv5 算法

3.2.1 注意力机制

科学家在对人类视觉系统的研究发现，次要信息会被人们在处理大量信息时自动忽略，将更多的注意力集中在需要特别关注的要素上，以便在有限的时间内迅速筛选出高价值信息。传统的神经网络会无差别对待所有输入数据的信息，而忽略了不同输入之间的重要性和相关性。这可能导致网络在处理数据时丢失重要信息，或者受到数据中冗余信息的干扰。因此，受人类视觉系统启发，科学家们将注意力机制与神经网络融合，模仿人类视觉，使神经网络可以在特征图的卷积过程中发现目标对象后聚焦于目标对象，并抑制特征图中的无关特征信息[63]。与传统的卷积单元不同，注意力机制不仅克服了感受野仅能关注相邻区域特征的不足，而且还能全局地考虑其他区域对当前区域的影响。在神经网络中，注意力机制使模型能够更清晰地解释在输入中哪些部分引起了特定输出，提高了对模型决策的理解。

因此，随着人工智能研究的不断深入，注意力机制已被广泛应用于各学科领域[64]。通过引入注意力机制，研究人员在神经网络研究中取得了显著的性能提升，并使得模型更具灵活性和智能性，能够处理不同任务和输入数据。这一趋势使得注意力机制成为深度学习领域备受关注的关键技术之一。

注意力机制主要包括空间域、通道域和混合域三种类型。空间域注意力通过在特征图的通道维度进行降维，为不同区域附加权重，从而强化关键区域特征、抑制次要区域，突显模型对图像空间信息的关注。通道域注意力以 SENet(Squeeze-and-Excitation Networks, SENet)[65]为例，通过压缩特征图的空间维度，为每个特征通道分配权重，专注于关键通道，减弱非关键通道的影响，提高了网络在通道层面的表达能力。SENet 通过赋予包含重要信息的通道更大的权重，从而提升了模型的性能，并成功地整合到分类或检测模型中，取得了显著的效果。然而，该模块存在一个不足之处，即通过全局平均池化将空间信息嵌入通道时会丢失一部分信息。混合域注意力以 BAM(Bottleneck Attention

Module, BAM)[66]、CBAM(Convolutional Block Attention Module, CBAM)[67]为代表，兼具通道和空间特征的优势，通过综合这两方面的信息，模型在性能上表现更出色。这些不同类型的注意力机制使得神经网络更加灵活，更智能地处理关键信息，提高模型在各种任务中的性能和泛化能力。在实际应用中，可以根据任务需求选择适合的注意力机制类型，或者组合不同类型的注意力机制，以优化模型的表现。

CBAM 注意力机制将空间注意力机制 (Spatial Attention Module, SAM)[68]与通道注意机制 (Class Activation Map, CAM)[69]相结合。其中，SAM 是一种空间的注意力机制，其主要作用是突出显示卷积神经网络中特定空间区域的信息。SAM 与 CAM 相似，旨在提高对输入图像中重要区域的关注度，但其主要着眼于空间注意力的调整。其基本思想是通过学习特征图上每个空间位置的权重，来调整特征图中不同位置对任务的贡献。通过这种方式，SAM 可以帮助网络更好地理解和利用图像的空间信息，从而提高模型的性能和泛化能力。SAM 网络结构图如图 3-5 所示。空间注意力机制函数表达式如式 (3-1)所示。

$$M_S(F)=\sigma(f^{7\times7}([AvgPool(Fn); MaxPool(Fn)]) \otimes Fn \qquad (3\text{-}1)$$

式中：σ 为 Sigmoid 激活函数；$f^{7\times7}$ 表示滤波器大小为 7×7 的卷积运算；$\otimes$ 为逐元素相乘。

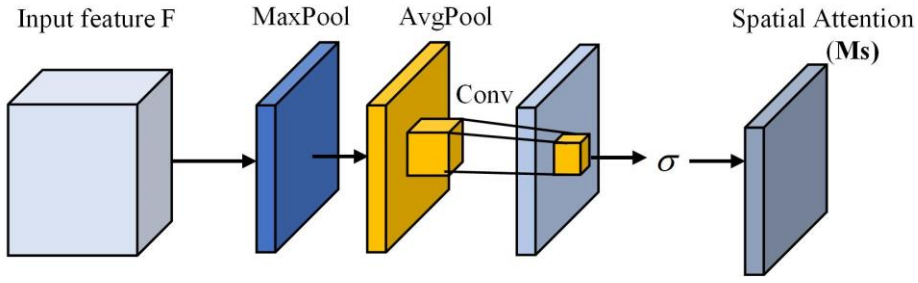


图 3-5 SAM 网络结构图

CAM 注意力机制则是用于解释卷积神经网络中特定类别激活区域的方法。其基本思想是通过捕获网络中最后一个卷积层的特征图与全局平均池化的关系来生成类激活图。CAM 的引入使得网络能够更加直观地展示其分类决策的依据，增强了模型的可解释性和可理解性。CAM 网络结构图如图 3-6 所示。通道注意力机制函数表达式如式 (3-2)。

$$M_C(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \qquad (3\text{-}2)$$

式中： σ 为 Sigmoid 激活函数；MLP 为多层感知机，AvgPool 为平均池化；MaxPool 为最大池化。

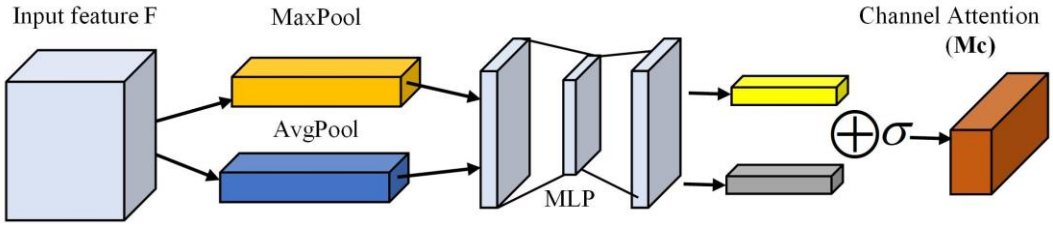


图 3-6 CAM 网络结构图

SAM 和 CAM 结合后的 CBAM 网络结构示意图如图 3-7 所示，CBAM 旨在根据原始特征图获得不同的值，并关注不同通道上的信息。然后，它将空间和通道分布信息与原始特征图信息相乘，以获得新的特征图。CBAM 在输入端将目标图像输入，并在通道注意力和空间注意力方向上对特征对象进行全局最大池化和平均池化。在通道方向生成注意力的过程中，将池化结果添加到全连接层，并将其转换为一维向量，然后将一维向量与通道注意力相乘，以获得特征图。而空间注意力首先进行池化以生成空间二维向量，然后将其串联和卷积以获得空间注意力特征图。

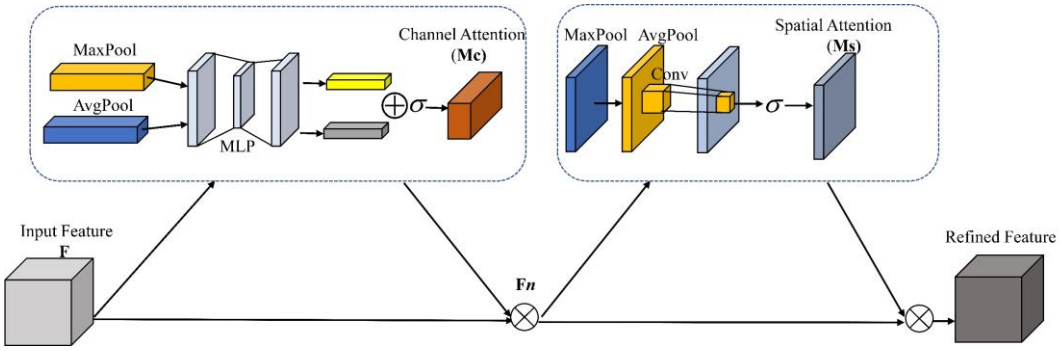


图 3-7 CBAM 网络结构图

3.2.2 基于 CBAM 的主干网络改进

YOLOv5 主干网络中的 CSP1_3 模块虽然能够让模型学习到更多的图像特征，但其特征提取能力较弱，无法有效地关注和抑制特征。因此，本文将轻量级卷积注意力模块与主干网络中的 CSP1_3 模块融合，形成了新的 CSP1_3CBAM 模块，增强网络的性能。这个模块可以让模型学习到更多的特征，并且沿着通道和空间两个独立的维度推断注意力图。然后，将这些注意力图与输入特征图相乘，实现自适应特征细化，注意力机制可以在特征图的卷积过程中发现目标对

象后聚焦于目标对象，从而改善有效特征的提取并抑制提取无关特征。改进后的网络流程图和框架如下图 3-8、图 3-9 和图 3-10 所示。

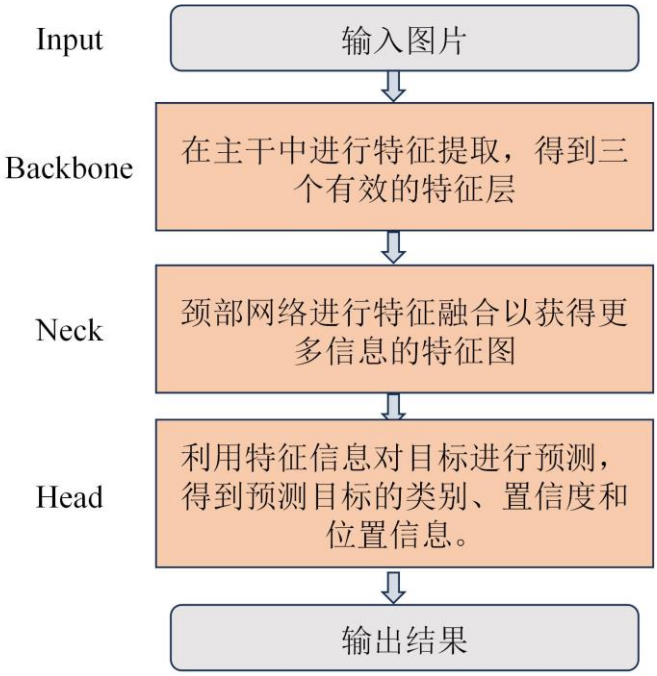


图 3-8 YOLOv5 算法流程图

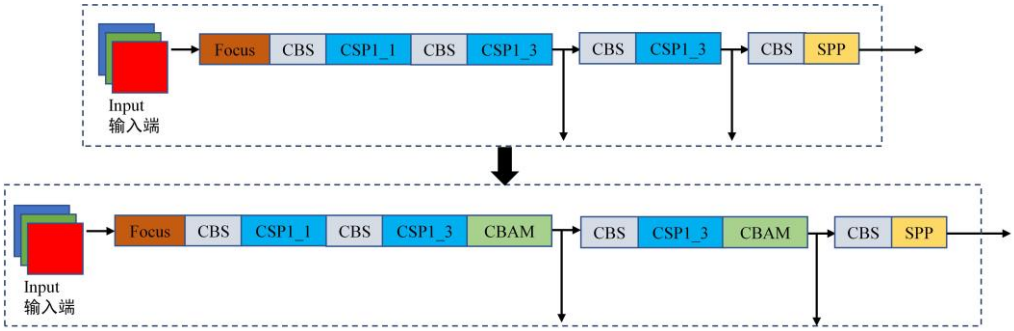


图 3-9 YOLOv5 主干网络改进前后对比图

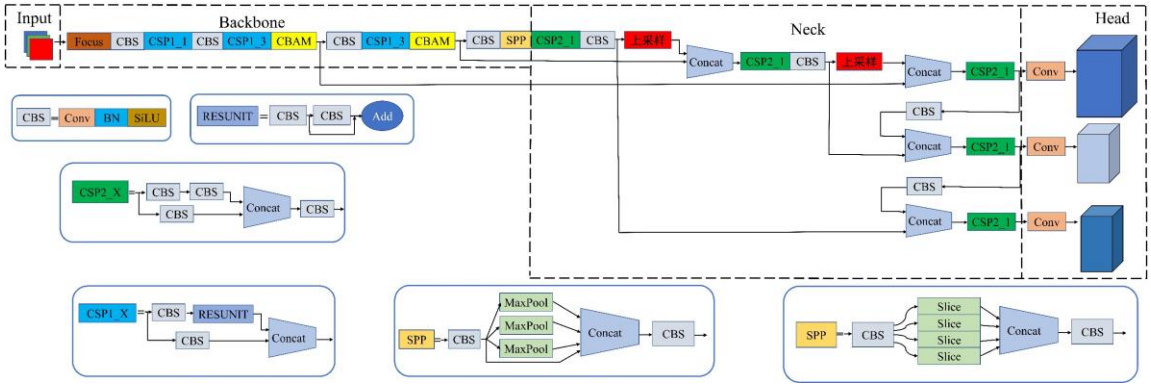


图 3-10 改进后的 YOLOv5 网络结构图

3.2.3 ACON-C 激活函数

激活函数是神经网络中的一种函数，用于对神经元的输入进行非线性变换[70,71]，从而引入非线性性质，增强模型的表达能力。在神经网络中，每个神经元的输出都会经过激活函数的作用，以确定是否激活神经元，并将激活后的输出传递给下一层。激活函数的引入使得神经网络能够学习和表达复杂的非线性关系，这在解决许多实际问题中至关重要。若没有激活函数，即使神经网络有多层，整体仍然只是一个线性变换，无法捕捉到数据中的非线性特征，从而限制了网络的表达能力和学习能力。因此，选择适当的激活函数对于神经网络的性能和训练效果至关重要。

本文选取 ACON-C 激活函数作为网络标准卷积层中的激活函数，与传统的激活函数相比，ACON-C 允许每个神经元自适应地激活或不激活，通过引入切换因子去学习非线性(激活)和线性(非激活)之间的参数切换[72]。这种激活行为可有效提高模型的泛化能力和迁移性能。其函数表达式如式 (3-3) 所示：

$$f_{\text{ACON-C}}(x)=(l_1-l_2)x\cdot\sigma\left[\beta(l_1-l_2)\right]+l_2x \tag{3-3}$$

式中：l_1 和 l_2 表示可以学习的参数，初始值 $l_1=1$，$l_2=0$；σ 为 Sigmoid 激活函数；β 为切换因子。

切换因子 β 直接决定激活函数的非线性程度，其值的大小主要取决于模型结构的通道数(C)、特征图的大小($H\times W$)，其函数表达式如式(3-4)所示：

$$\beta=\sigma\sum_{c=1}^{C}\sum_{h=1}^{H}\sum_{w=1}^{W}x_{c,h,w} \tag{3-4}$$

3.2.4 损失函数优化

YOLOv5 采用的损失函数是一种组合损失，主要包括目标位置损失、目标类别损失和目标置信度损失。这些损失函数的设计旨在确保模型对目标进行准确检测和定位。目标分类损失关注模型是否能正确预测目标的存在，位置损失考虑边界框的中心坐标和宽高坐标的准确性，目标置信度损失用于区分前景目标和背景，而分类损失则确保模型能够准确的识别目标类别。这些损失通过权重超参数进行线性组合，形成 YOLOv5 的损失函数，以平衡各个损失部分的影响。总体而言，这样的损失函数设计有助于模型在目标检测任务中取得更为精准和鲁棒的性能。

因此，损失函数的选择对于图像分类和语义分割等任务中使用的模型至关重要[73-75]。因为它可以量化模型对结果的预测程度，并估计检测模型与真实数据之间的差距。传统的损失函数通常依赖于边界框回归指标的聚合，如预测框与真实框之间的距离、重叠面积和宽高比。然而，目前提出和使用的许多方法中真实框与预测框之间的方向匹配问题未被考虑到。这种缺失导致网络收敛速度较慢且效率较低。为了解决这一问题，可以将边界框的四个边界坐标点作为整体进行回归，其回归值取决于网络训练过程中产生的预测框与真实框之间的交集与并集的比值[76]。其函数表达式如式 (3-5)和式 (3-6)所示：

$$\mathrm{IoU}=\frac{|A\cap B|}{|A\cup B|} \tag{3-5}$$

$$\text{IoU-loss}=1-\mathrm{IoU}=1-\frac{|A\cap B|}{|A\cup B|} \tag{3-6}$$

式中：A 为预测框面积；B 为真实框面积。

在这种情况下，当 IoU-loss 的值越小时，意味着预测框和真实框之间的重叠程度越高，相反， IoU-loss 的值越大，则表示两者之间的重叠程度较低。然而，在实际情况中，可能会遇到预测框与真实框完全不重叠的情况，此时 IoU 的值为 0， IoU-loss 的值为 1，导致损失函数失去了可导性质，无法计算预测框与真实框之间的距离，从而无法进行进一步的学习。尽管 IoU 的值可以被计算出来，但是由于预测框的位置不同，也会影响到识别精度[77]。IoU-loss 无法正常作用情形如图 3-11 所示。

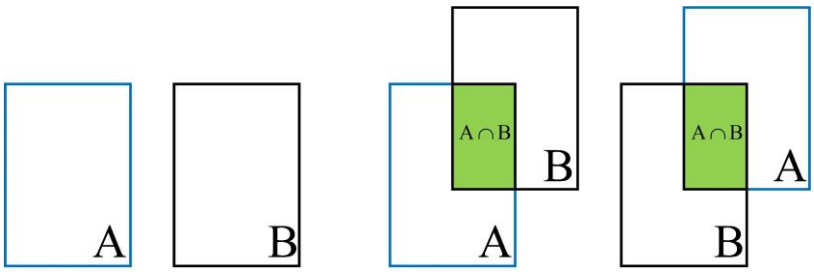


图 3-11 IoU-loss 无法正常作用情形图

针对有关损失函数，如 GIoU (General IoU， GIoU)[78]、 DIoU (Distance IoU, DIoU)[79] 和 CIoU 等，均未考虑到真实框与预测框之间的方向不匹配问题，导致

模型收敛速度较慢且效率低，与 DIoU-loss 相比，CIoU-loss 考虑了回归要素中的长宽比，但仍未考虑到期望之间的向量角度。

因此，本文借助于 SIoU (Smooth Intersection over Union, SIoU) 损失函数来对 YOLOv5 的损失函数进行了改进，解决传统交并比 (IoU) 计算中的不连续性问题。通过引入平滑处理，使 IoU 的计算更加连续，该损失函数计算平滑交并比 (Smooth IoU) 并将其与 1 之间的差异值作为损失值，从而有助于提高训练过程的稳定性，并能更好地引导模型学习准确的目标位置。该函数考虑了期望回归之间的向量角度问题，并定义了新的惩罚指标，以提高训练速度和推理准确性， SIoU[80]损失函数由四个组件组成：角度损失 (angle cost)、距离损失 (distance cost)、形状损失(shape cost) IoU 损失 (IoU cost)。

(1) 角度损失如图 3-12 和公式 (3-7)-公式 (3-10)所示：

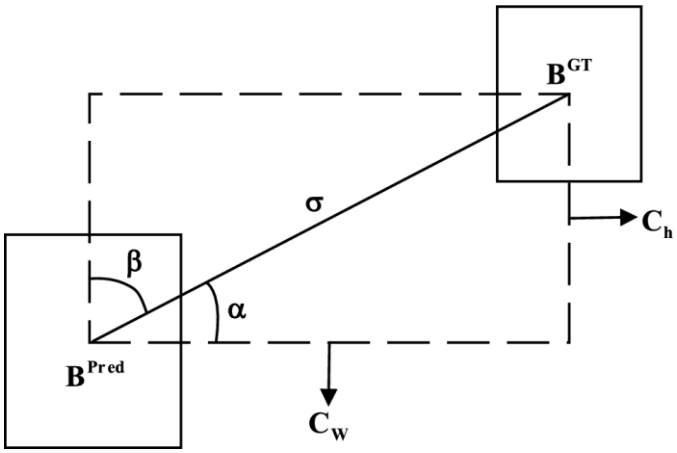


图 3-12 计算角度损失在损失函数中的贡献方案

$$\begin{aligned}\Lambda &= 1-2*\sin^2(\arcsin(\frac{c_h}{\sigma})-\frac{\pi}{4}) \\ &= \cos(2*(\arcsin(\frac{c_h}{\sigma})-\frac{\pi}{4}))\end{aligned} \tag{3-7}$$

其中 c_h 为真实框和预测框中心点的高度差，σ 为真实框和预测框中心点的距离，事实上 $arcsin(\frac{c_h}{\sigma})$ 等于角度 α。

$$\frac{c_h}{\sigma}=\sin(\sigma) \tag{3-8}$$

$$\sigma=\sqrt{(b_{c_x}^{gt}-b_{c_x})^2+(b_{c_y}^{gt}-b_{c_y})^2} \tag{3-9}$$

$$c_h=\max(b_{c_y}^{gt},b_{cy})-\min(b_{c_y}^{gt},b_{cy}) \tag{3-10}$$

$(b_{c_x}^{gt}, b_{c_y}^{gt})$为真实框中心坐标，$(b_{c_x}, b_{c_y})$为预测框中心坐标，可以注意到当 α 为$\frac{\pi}{2}$,或 0 时，角度损失为 0，在训练过程中若 α<$\frac{\pi}{2}$，则最小化 α，否则最小化 β。

(2) 距离损失如下图 3-13 和公式(3-11)所示：

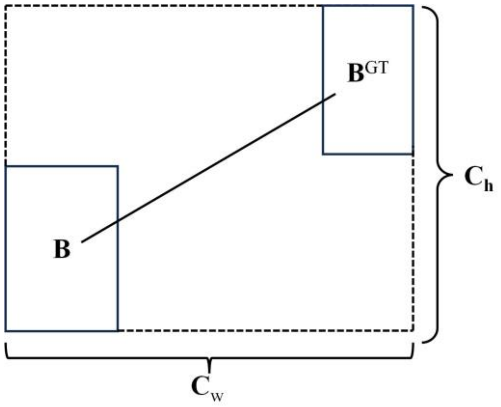


图 3-13 计算真实边界框与其预测之间距离的方案

$$\begin{aligned}\Delta &= \sum_{t=x,y}(1-e^{-\gamma\rho_t}) \\ &= 2-e^{-\gamma\rho_x}-e^{-\gamma\rho_y}\end{aligned} \tag{3-11}$$

其中：$\rho_x = \frac{b_{c_x}^{gt}-b_{c_x}}{c_w}$，$\rho_y = \frac{b_{c_y}^{gt}-b_{c_y}}{c_h}$，$\gamma = 2-\Lambda$，$c_w, c_h$为真实框和预测框最小外接矩形的宽和高。

上述公式计算了角度 Δ，其中包括了 SIoU 中涉及的角度和距离计算。它表示角度计算的最终结果。x 和 y 表示实际框和预测框中心点之间角度的正弦值。同时，ρ 是实际框和预测框中心点之间距离相对于它们的最小包围矩形的宽度和高度的平方的比率。

(3) 形状损失如公式 (3-12)和公式(3-13)所示：

$$\Omega=\sum_{t=w,h}(1-e^{-W_t})^{\theta}=(1-e^{-W_w})^{\theta}+(1-e^{-W_h})^{\theta} \tag{3-12}$$

其中：

$$W_w=\frac{\left|w-w^{gt}\right|}{\max(w,w^{gt})}, W_h=\frac{\left|h-h^{gt}\right|}{\max(h,h^{gt})} \tag{3-13}$$

这里 e 表示自然对数的底，(w, h)和(w^{gt}, h^{gt})分别为预测框和真实框的宽和高，这里的 θ 值确定了形状损失对边界框损失的贡献，并且定义了模型对形状损失的关注程度，在本文中设置为 4。

(4) IoU 损失如下图 3-14 和公式 (3-14)所示：

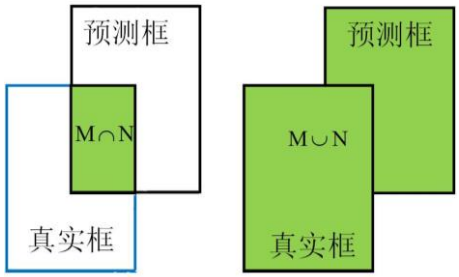


图 3-14 IoU 损失的交并集示例

$$\mathrm{IoU}=\frac{|A\cap B|}{|A\cup B|} \tag{3-14}$$

通过在损失函数中引入方向，与现有方法(如 CIoU 损失)相比，在训练阶段更快地收敛，推理性能更好。所提出的改进方法提高了模型检测精度，加快了收敛速度，最终 SIoU 函数表达式如式 (3-15)所示：

$$\text{SIoU-loss}=1-\mathrm{IoU}+\frac{\Delta+\Omega}{2} \tag{3-15}$$

3.3 实验验证及结果分析

3.3.1 数据集

TT100K 中国交通标志数据集由清华大学和腾讯根据行车记录仪拍摄的视频制作完成。在本章的实验中，选择了超过 9000 张图像作为训练集和验证集。数据集涵盖了 151 个类别，但只有 45 个类别的实例数量超过 50，这导致了严重的数据分布不均衡问题，直接训练可能会导致过拟合现象。因此，对数据集进行了筛选，只选择了实例数量超过 50 的 45 个类别进行训练。

3.3.2 实验环境

本文实验是在 Windows10 系统下进行训练，采用 Pytorch 深度学习框架，编程语言为 Python3.8，CUDA 和 Cudnn 版本分别为 11.3 和 7.5，显卡为 NVIDIA GeForce GTX3060，显存 6GB，CPU 为 i7-11800H 的八核处理器。其中输入图像大小、初始学习率、batch size、优化器、Epoch(训练轮数)具体设置参数如下表 3-1 所示，在训练过程中使用余弦退火(Cosine Annealing)方法来衰减学习率。

表 3-1 有关训练参数

参数	数值
图像大小	640 ×640
batch size	16
epoch	150
初始学习率	0.0001
weight decay	0.0005
优化器	SGD

图 3-15 为损失函数改进前后的对比，在 0-30 轮之间，改进前后损失函数曲线基本重合，在 30 轮后，改进后的损失函数曲线明显低于原模型曲线，且曲线更加平滑，其结果说明改进后的损失函数优于改进前的损失函数， 改进后的损失函数使得模型的预测与实际目标之间的差异也变小，因此本章选取 SIoU 损失函数作为 YOLOv5 算法的损失函数。

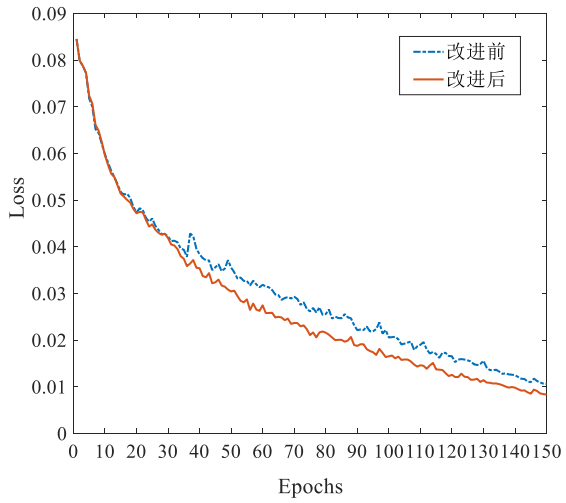


图 3-15 改进后与改进前损失函数训练效果图

3.3.3 定量分析

本章在 TT100K 数据集上进行原算法、改进后的算法实验和消融实验来验证加入 CBAM 注意力机制、改进激活函数以及改进损失函数对网络性能的影响。将不同模型放在同一硬件环境下测试，改进前后 mAP 的曲线放在同一张图中进行对比，改进前后模型均训练轮数为 150 轮，如图 3-16 所示，在 0-10 轮之间改进前后的模型曲线基本重合，在 10 轮后改进后的模型曲线逐渐高于改进前的模型曲线，且曲线整体波动范围小，因此改进后 YOLOv5 算法的 mAP 有了显著提升。

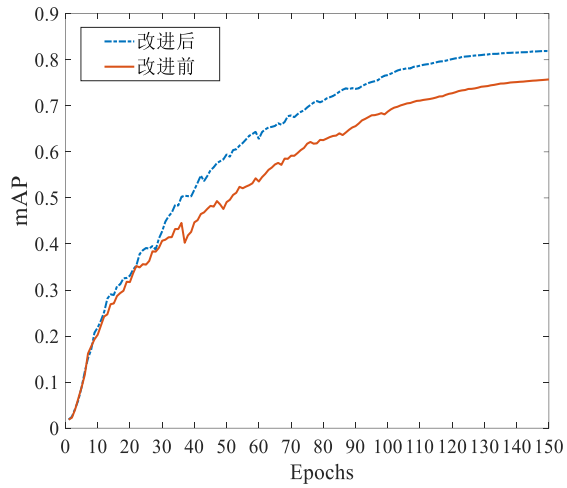


图 3-16 改进前后模型 mAP 对比图

为了进一步探讨 SIoU 损失函数，ACONC 激活函数以及 CBAM 注意力机制对交通标志检测算法的参数量、GFLOPs、召回率和准确率的具体影响，设计了一组消融实验，具体做法是分别加入三个变量和同时加入三个变量，在相同实验条件下和同一数据集上进行消融实验。

通过对比分析：(1)YOLOv5 原模型；(2)YOLOv5 原模型+SIoU 损失函数；(3)YOLOv5 原模型+ACONC 激活函数；(4)YOLOv5 原模型+CBAM；(5)YOLOv5 原模型+SIoU+ACONC+CBAM；具体消融结果如表 3-2 所示。

表 3-2 消融实验结果

模型	SIoU	ACONC	CBAM	Parameters	GFLOPs	P/%	R/%	mAP@0.5/%
YOLOv5				7066,239	16.4	73.2	74.2	75.7
A	√			7066,239	16.4	79.3	75.8	79.7
B		√		7469,903	16.7	77.2	74.4	78.7
C			√	7110,151	16.5	78.0	73.5	78.6
Ours	√	√	√	7513,815	16.8	81.9	77.2	81.9

由表 3-2 所示，YOLOv5 原算法的参数量为 7066239，GFLOPs 为 16.4G，精确率为 73.2%，召回率为 74.2%，平均精确率为 75.7%，当在 YOLOv5 算法中加入 SIoU 损失函数时，由消融实验结果可知，参数量和 GFLOPs 保持不变，表明在算法改变损失函数时的网络的层数未发生变化，精确率和召回率也得到提升，mAP 从 75.7%增至 79.7%，提高了 4.0 个百分点。当仅改进激活函数时，随着参数的增加，GFLOPs 也增加，但网络的 mAP 从 75.7%增至 78.7%，提高了 3.0 个百分点，精确率和召回率也得有提升。当仅融合注意力机制模型时，参数

和 GFLOPs 略微增加，网络的精度值也略微提高，mAP 从 75.7%增至 78.6%，提高了 2.9%。

当将这三者都添加到 YOLOv5 算法中时，网络的参数和 GFLOPs 增加，网络的精度值也提高，改进后的 YOLOv5 精确率从 73.2%提高到 81.9%，增加了 8.7%，召回率从 74.2%提高到 77.2%，增加了 3.0%，mAP 从 75.7%提高到 81.9%，增加了 6.2%。经过计算，FPS 也从每秒 26.88 帧增加到 30.42 帧。总体而言，改进后的模型的检测速度和准确性均得到有效提升。

本章将改进后的算法与其他流行算法进行了比较，如 SSD300、Faster R-CNN、Zhu 的 TT100K、YOLOv3 和 YOLOv4[81,82]。所有模型的学习率设为 0.001；输入大小、Epoch 和 Batch Size 大小如表 3-3 所示。锚框设置为原始数据，YOLOv5 模型采用与本文相同的参数设置策略进行参数的训练和调整。输入图片大小为 640 × 640，Batch Size 设为 16，epoch 为 150。对比结果如表 3-3 所示。

表 3-3 模型的部分训练参数

模型	输入图片尺寸	学习率	Epoch	Batch Size
YOLOv5	640 × 640	0.001	150	16
SSD300	300 × 300	0.001	300	32
Faster R-CNN	416 × 416	0.001	2500	20
Zhu	-	-	-	-
YOLOv3	416 × 416	0.001	200	8
YOLOv4	416 × 416	0.001	200	8
文献[82]	416 × 416	0.001	200	8
本章算法	640 × 640	0.001	150	16

表 3-4 改进算法与其他算法比较

检测模型	P/%	R/%	mAP/%	FPS/(帧/秒)
YOLOv5	73.2	74.2	75.7	26.88
SSD300	76.5	62.3	76.3	14.1
Faster R-CNN	69.8	53.6	80.9	0.6
Zhu	-	-	81.6	5.8
YOLOv3	26.3	37.9	58.1	29.6
YOLOv4	54.0	64.6	74.2	41.7
文献[82]	-	-	75.2	31.3
本文算法	81.9	77.2	81.9	30.42

从表 3-4 可以看出，相较于所示 SSD300 和 Faster R-CNN 算法，本文改进算法的 mAP@0.5 明显提升，Faster R-CNN 算法在 TT100K 数据集下检测交通标志

时 mAP 为 80.9%，精准度 P 为 69.8%，召回率 R 为 53.6%，mAP 虽与本文算法接近，但检测速率本文算法高于 Faster R-CNN 算法。

3.3.4 定性分析

最后，为了能够直观地看出本章改进算法的性能，选择在校园路面、街道和高速公路、高架桥采集的 4 张图片进行测试，测试结果如图 3-18 所示。

(a) 高速公路上的多个交通标志和小目标

(b) 街道上的单一交通标志

(c) 校园内的单一交通标志

(d) 中小目标

图 3-17 改进的 YOLOv5 算法对比检测图

在图 3-17 中，由第 2 章第 2.3.2 小节可知，p23 代表禁止左转，pl20 代表限速 20 km/h，pl30 代表限速 30 km/h，pl40 代表限速 40 km/h。图中最左列的是交通标志的样本图像，右边的两列分别是未改进模型和改进模型的检测结果。每个预测结果包括三个信息：交通标志的位置、类别和置信度。当遇到有多个待检测目标的场景时，YOLOv5 对部分目标的检测不准确。如图 3-17 所示，当输入图像中有许多目标时，就会出现目标丢失检测问题或不正确检测问题。在图 3-18(a)中，由于数据集中实例数量不足，左下角的禁止通行标志它既没有被训练也没有被识别，因此遗漏了禁止通行标志。在图 3-17(b)中，初始模型检测的置信水平为 0.85，改进模型检测的置信水平为 0.95。在图 3-17(c)中，出现限速标志识别错误，将自行车前轮误检测为限速标志。在图 3-18(d)中，本文改进的算法可以正确检测距离较远、目标较小的交通标志，并且置信度高于初始模型。因此本研究中使用的新的 YOLOv5 模型准确地识别了数据集收集的图像中存在的交通标志类别，未出现遗漏或错误检测的实例。

图 3-18 交通标志部分遮挡检测示例

对于部分遮挡的交通标志检测，由于本文在卷积层中加入了注意机制模块，可以有目的地聚焦特征图中的重要信息，从而更好地检测部分遮挡的交通标志。如图 3-18 所示，被遮挡的限速 40km/h 标志和禁止停车标志均被正确识别。

3.4 本章小结

本章详细的介绍了针对当前交通标志识别技术不能满足智能辅助驾驶系统实时性和准确性要求的问题，提出了一种改进的 YOLOv5 算法，该卷积注意力模型和 CSP1_3CBAM 模型融合在一起，建立了一个新的 CSP1_3CBAM 模型；同时引入 ACONC 作为 YOLOv5 的激活函数，从而提高了模型对有用信息的特征提取能力，提高了召回率、准确率、检测准确率和检测速度，显著减少了实时检测过程中的误检和漏检。最后，在 TT100K 和进行训练和测试，其结果表明，改进后的 YOLOv5 算法的损失函数值比改进前明显降低。本章所提出的改进方法虽然在精度上有所提升，但参数量大，导致存在不易于部署到移动设备上的不足。

第 4 章 基于改进 Ghost 卷积的交通标志轻量化模型

在研究交通标志识别技术时，不仅需要关注算法在检测精度方面的表现，还必须考虑算法的检测速度、实时性等因素，以确保算法能够在实际环境中迅速而准确地识别交通标志。对于以上要求，第 3 章提出了一种改进的 YOLOv5 算法，解决了检测精度与速度无法很好兼顾的问题，但该算法参数量较大，轻量化不足，不利于算法的实际应用。

基于此问题，本章在改进 YOLOv5 算法基础之上，融合轻量级神经网络 GhostNet 和 CBAM 注意力机制对 YOLOv5s 算法的网络结构进行改进，设计了一种名为 YOLO-GCC 的交通标志识别算法。该算法成功减少了网络模型的参数量，并且提高了模型的检测速度，因此更满足驾驶辅助系统的实时性需求。

4.1 轻量化网络的构建

4.1.1 方案一：基于 MobileNetV3 的算法轻量化设计

(1) MobileNetV1 介绍

自从 2012 年 AlexNet 首次将卷积神经网络用于图像分类并在 ImageNet 竞赛中夺冠以来，研究人员陆续设计了越来越多的深度神经网络模型，其中包括经典的 VGGNet16/19[83]、GoogleNet[84]、ResNet50[85]等。相对于传统的分类算法，这些模型表现出了非凡的性能。然而，随着对网络的深入研究，由于模型计算带来的巨大存储压力和计算负担开始限制了深度学习模型在实际应用中的广泛使用，这使得它们无法在车载系统中高效运行。

为解决这一问题，Google 提出了一种轻量级的深度神经网络，即 MobileNetV1[86]，MobileNetV1 的设计目的在于在保持足够性能的前提下，显著减小神经网络的参数量和计算量，从而使模型能够在移动端实时应用中发挥作用。其创新之处主要体现在深度可分离卷积、轻量级设计和宽度乘数等方面。深度可分离卷积是 MobileNetV1 的核心组件之一。相对于传统卷积操作，深度可分离卷积分为深度卷积和逐点卷积两个步骤，有效地减小了每个卷积核的通道数，从而降低了计算成本。这种设计策略使得 MobileNetV1 能够在轻量级硬件上运行，同时仍能保持令人满意的性能。

对于输入特征图尺寸为$H_1 \times W_1$，输入通道为C_1，输出通道为C_2，卷积核大小为$K \times K$的标准卷积，可得出其参数量为式(4-1)所示。

$$K \times K \times C_1 \times C_2 \tag{4-1}$$

深度可分离卷积的参数量分两部分计算，首先是深度卷积的参数量，深度卷积的输入特征图尺寸为$H_1 \times W_1$，C_1个尺寸为$K \times K$的深度卷积核对输入进行卷积，得到尺寸为$H_2 \times W_2 \times C_1$的中间特征图，因此深度卷积参数量为式(4-2)所示。

$$K \times K \times C_1 \times 1 \tag{4-2}$$

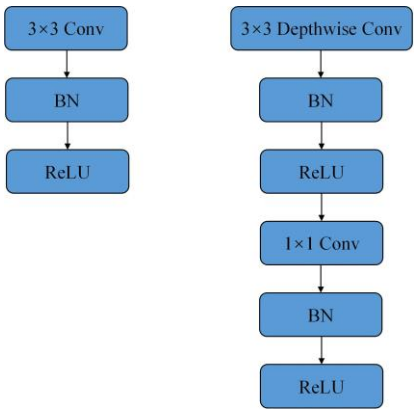
计算深度可分离卷积中逐点卷积的参数量，C_2个尺寸为1×1的逐点卷积的参数量为式(4-3)所示。

$$K \times K \times C_1 + C_1 \times C_2 \tag{4-3}$$

深度可分离卷积计算量的下降倍数，即标准卷积与深度可分离卷积的参数量比值如公式 (4-4)所示。

$$P = \frac{K \times K \times C_1 + C_2 \times C_2}{K \times K \cdot C_1 \times C_2} = \frac{1}{C_2} + \frac{1}{K^2} \tag{4-4}$$

轻量级设计体现在多个方面，例如 1×1 卷积用于降维，全局平均池化有助于减小特征图的尺寸，批标准化有助于提高训练的稳定性。标准卷积和深度可分离卷积如图 4-1 所示。从图中可以看出深度可分离卷积将标准卷积的 3×3 卷积换成了同样尺寸的深度卷积，再在后面加上一个 1×1 的逐点卷积并连接 BN 层和 ReLU 激活函数构成。



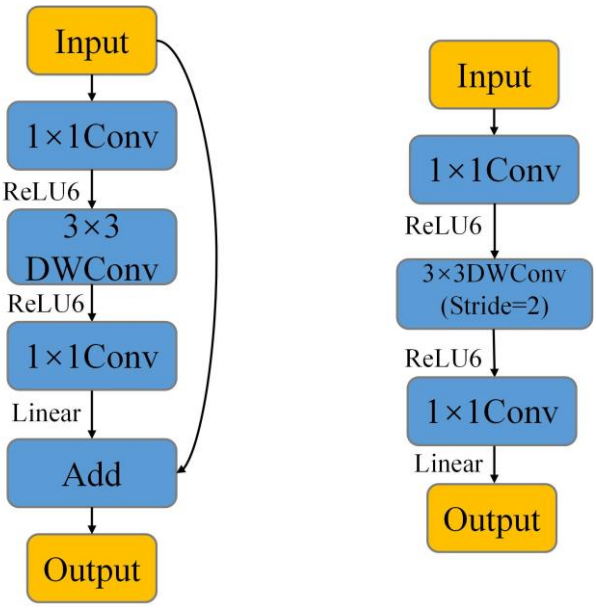
(a) 标准卷积　　(b) 深度可分离卷积

图 4-1 标准卷积和深度可分离卷积

MobileNetV1 中的宽度乘数 (width multiplier) 是一个用于调整模型宽度的超参数。通过调整宽度乘数，可以在不同的网络层中缩放每个卷积层的通道数，从而灵活地适应不同的计算资源和性能需求。典型的宽度乘数取值范围是 (0，1)，其中 1 表示保持原始通道数，0.5 表示减少一半的通道数，0.25 表示减少至四分之一的通道数。这个调整可以在模型的速度和准确性之间找到平衡，适应不同设备和应用场景的需求。

(2) MobileNetV2 介绍

MobileNetV2[87]由 Google 于 2018 年提出，旨在对 MobileNetV1 进行改进和扩展，以提供更高的性能和更好的模型表示能力。在多个方面，MobileNetV2 对 MobileNetV1 进行了优化和改进。首先，MobileNetV2 引入了一种新的网络结构，称为倒残差结构 (Inverted Residuals)。这一结构通过使用轻量级的深度可分离卷积和跳跃连接，加强了信息流动，有助于提高模型的表达能力。相比于 MobileNetV1 的普通残差结构，倒残差结构更适用于轻量级网络。其次，MobileNetV2 采用了一种线性瓶颈 (Linear Bottleneck) 设计，即在卷积之前和之后都应用了线性激活函数。这有助于减小特征图的维度，提高网络的性能。此外，MobileNetV2 引入了深度可分离卷积的残差块，使得模型在保持轻量级的同时，能够更好地捕捉复杂的特征。



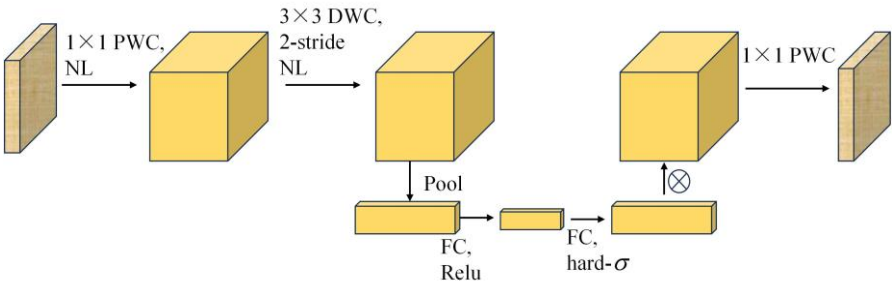
(a) stride=1　　(b) stride=2

图 4-2 MobileNet V2 瓶颈层网络结构

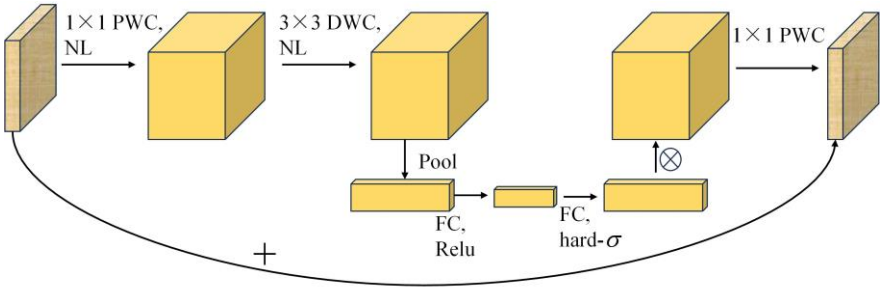
(3) MobileNetV3 介绍

MobileNetV3 是一种专为移动设备和资源受限设备设计的高效卷积神经网络模型[88]。它在原始 MobileNet 架构的基础上进行了几项改进，经过优化，提高了识别准确性并缩短了训练时间[89,90]。MobileNetV3 利用深度可分离卷积，将标准卷积操作拆分为独立的深度卷积和逐点卷积。这种拆分将空间滤波和特征生成分离，显著减少了卷积的参数和计算量。因此，深度可分离卷积在降低计算复杂性的同时保持了高准确性。此外，MobileNetV3 引入了残差倒置结构和线性瓶颈层，以促进特征融合。这种设计能够从高维空间中提取特征并减少信息损失，使得 MobileNetV3 在准确性和效率之间取得了良好的平衡，因此适于在有限计算资源的设备上的实时应用。先前的研究表明，相较于 XceptionV3，MobileNetV3 仅使用了其 1/10 的参数，并且在模型训练时间上仅需 1/7 的时间，就能达到更高的分类准确性。

MobileNetV3 作为前几个版本的继任者，进一步简化了网络结构，并将挤压和激励模块应用于残差倒置块。MobileNetV3 的残差倒置结构如图 4-3 所示。



(a) stride=1



(b) stride=2

图 4-3 MobileNet V3 网络残差倒置结构

SE 模块将深度可分离卷积的输出特征映射压缩成一个大小为 1×1×C 的向量，通过全局平均池化生成通道描述符。接着，通过两个连续的全连接层生成由特征映射通道贡献权重表示的向量。SE 模块的输出向量用于与深度可分离卷积的

原始输出特征映射进行逐通道乘法，从而提高了瓶颈的输出特征映射的质量。值得注意的是，SE 模块是一些瓶颈中的可选组件，并非在所有层中都存在。MobileNetV3 规格表如下表 4-1 所示。

表 4-1 MobileNetV3 规格表

输入	操作	扩展层输出通道数	输出通道数	SE	步长
224×224×3	Conv2d	-	16	-	2
112×112×16	bneck,3×3	16	16	-	1
112×112×16	bneck,3×3	64	24	-	2
56×56×24	bneck,3×3	72	24	-	1
56×56×24	bneck,5×5	72	40	√	2
28×28×40	bneck,5×5	120	40	√	1
28×28×40	bneck,5×5	120	40	√	1
28×28×40	bneck,3×3	240	80	-	2
14×14×80	bneck,3×3	200	80	-	1
14×14×80	bneck,3×3	184	80	-	1
14×14×80	bneck,3×3	184	80	-	1
14×14×80	bneck,3×3	480	112	√	1
14×14×112	bneck,3×3	672	112	√	1
14×14×112	bneck,5×5	672	160	√	2
7×7×160	bneck,5×5	960	160	√	1
7×7×160	bneck,5×5	960	160	√	1
7×7×160	Conv2d,1×1	-	960	-	1
7×7×960	pool, 7×7	-	-	-	1
1×1×960	Conv2d,1×1,NBN	-	1280	-	1
1×1×1280	Conv2d,1×1,NBN	-	k	-	1

本节引入 MobileNetV3 替换 YOLOv5 的主干网络，同时取消 Focus 模块和 SPPF 模块，黄色标注为改进后的主干网络，改进后的 YOLOv5 网络结构如下图 4-4 所示。

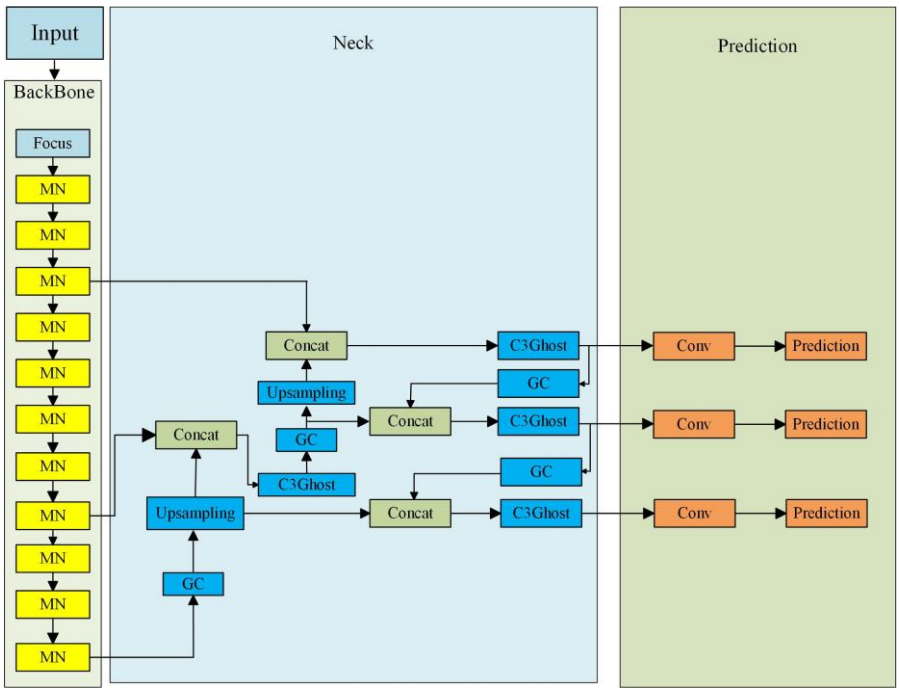


图 4-4 基于 MobileNetV3 的 YOLOv5 算法结构图

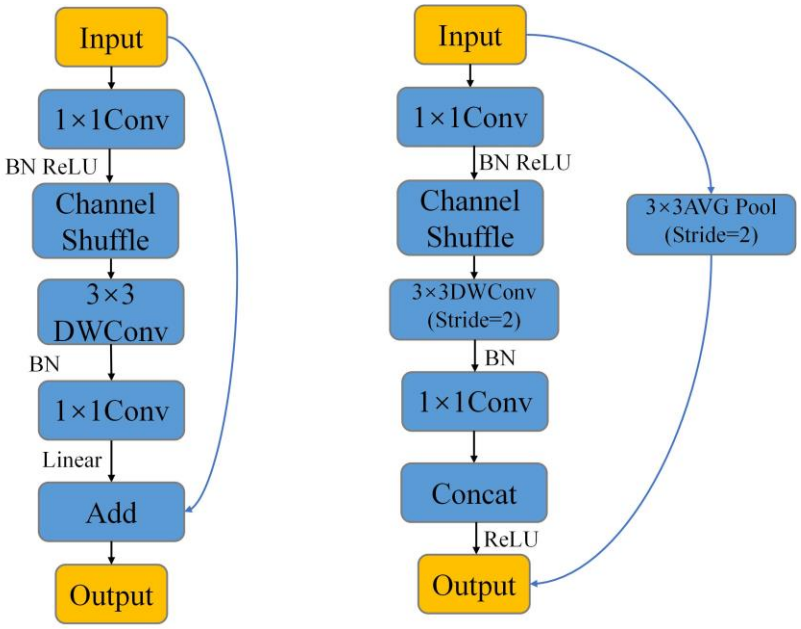
4.1.2 方案二：基于 ShuffleNetV2 的算法轻量化设计

(1) ShuffleNetV1 介绍

目前，为了实现轻量化效果，许多学者倾向于采用分组卷积操作，将主要计算量集中在 1×1 卷积运算上，典型的例子包括前文提到的 MobileNetV1。这种方法在模型压缩和速度方面有一定优势，但却牺牲了模型的检测精度并且在某种程度上导致了跨通道操作能力的丧失。基于这一考虑，ShuffleNetV[91]通过精心设计的通道重排技术来弥补跨通道操作能力的损失，既保持了轻量级的特性，又提高了模型的检测精度，因此成为解决轻量神经网络在精度和效率平衡方面的重要选择。

ShuffleNetV1 的主要创新点在于其基于分组逐点卷积 (group pointwise convolution) 和通道重排 (channel shuffle) 操作的轻量化设计。深度可分离卷积中，计算瓶颈主要出现在 PW 卷积阶段。为了减少 PW 卷积的计算负担，ShuffleNetV1 引入了分组逐点卷积，即将特征分成 g 组，只在每个组内进行 PW 卷积。然而，分组逐点卷积可能导致模型信息在各个组内被分割，组与组之间的信息交流丢失，进而影响模型的表达能力和识别精度。这一问题成为 ShuffleNetV1 在设计上需要解决的挑战之一。为了解决这一问题，ShuffleNetV1 在分组逐点卷积后引入了通道重排操作，以完成组间信息的有效交流。

ShuffleNetV1 提出了两种关键的网络单元，基本单元和空间下采样单元，整体网络结构如图 4-5 所示。基本单元是在残差单元的基础上进行改进的，将密集的 1×1 卷积替换为 1×1 的分组逐点卷积，在第一个分组逐点卷积后引入通道重排操作，删除 DW 卷积后的 ReLU 激活操作。空间下采样单元在基本单元的基础上进一步演变，在 shortcut 分支中插入一个步长为 2 的平均池化层，同时通过将 DW 卷积的步长由 1 调整为 2 来实现特征图的下采样，最后通过拼接特征图的方式使特征通道数翻倍。这些设计使得 ShuffleNetV1 在保持高效性的同时实现了更灵活的特征学习和通道交流。



(a) 基本单元　　(b) 下采样单元

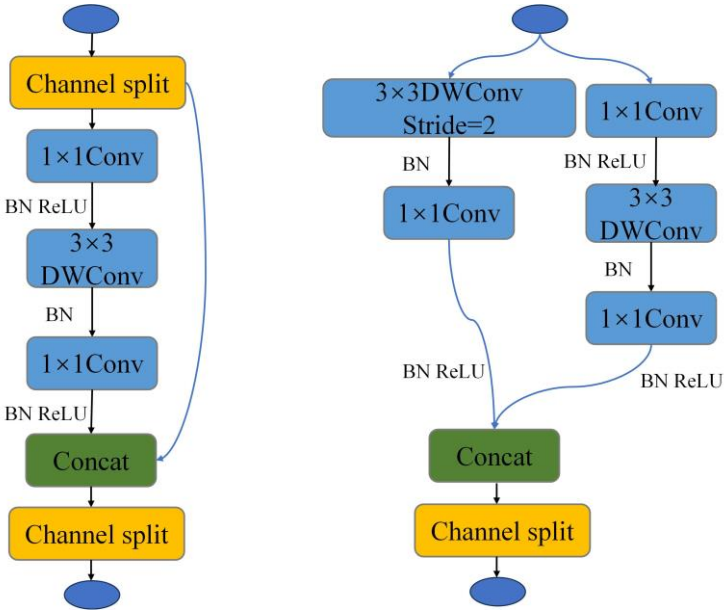
图 4-5 ShuffleNetV1 单元网络结构

(2) ShuffleNetV2 介绍

ShuffleNetV2[92]是对 ShuffleNetV1 的进一步改进和扩展，旨在提供更高的性能和更好的模型表示能力，该架构保留了 ShuffleNetV1 的核心设计理念，如逐通道的分组卷积和通道随机重排，同时在细节上进行了优化，以进一步提高网络的效率和表达能力。

ShuffleNetV2 轻量化网络将自身结构精妙地划分为基本模块和下采样模块，构建了一套高效的特征提取体系。基本模块的设计如图 4-6a 所示，专注于障碍物目标的特征捕捉，巧妙地将输入特征图在通道维度上分为两路。一路经过恒

等映射，保留原始特征。另一路则经历多重卷积操作，通过强化特征提取来更好地捕捉目标特征。两路特征图在最终进行拼接，使输出通道数与基本模块的输入通道数保持一致，通过通道混洗完成信息融合。这一设计既有力地强化了特征提取效果，又保持了特征图的尺寸和通道数量。下采样模块如图 4-6b 所示，它的作用在于扩大感受野、增加网络深度并获取全局信息。相较于基本模块，下采样模块去除了 Channel Split 通道分离组件，在两路分支上通过调整深度可分离卷积的步长，实现了特征图的下采样。当步长设为 2 时，输出特征图的尺寸变为原来的 1/2，两路压缩后的特征图在通道维度上进行拼接，通道数变为原先的两倍，最后进行通道混洗。这一设计的目的在于有效实现特征的下采样和通道数的增加，为网络的深度学习和全局信息的获取提供了强有力的支持。整体而言，ShuffleNetV2 的结构设计既注重网络的轻量化，适应移动端等资源受限场景，又在特征学习和信息传递方面具备了高度的灵活性[93]。这种高效而灵活的特性使得 ShuffleNetV2 在计算性能和模型精度之间找到了平衡，为深度学习在实际应用中提供了可行的解决方案。



(a) 基本单元　　(b) 下采样单元

图 4-6 ShuffleNetV2 单元网络结构

本节引入 ShuffleNetV2 模块替换 YOLOv5 的主干网络，使用 SN-a 代替图 4-6(a)基本单元，再使用 SN-b 代替图 4-6(b)下采样单元，替换后的 YOLOv5 网络算法结构如下图 4-7 所示。

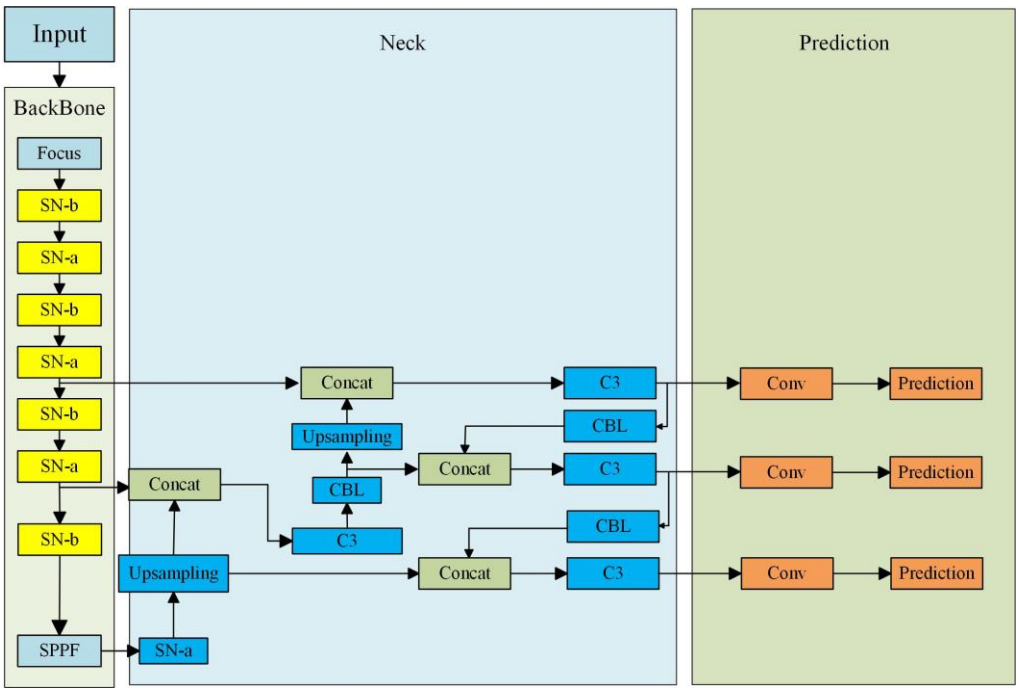


图 4-7 基于 ShuffleNetV2 的 YOLOv5 算法结构图

4.1.3 方案三：基于 GhostNet 的算法轻量化设计

GhostNet 是一种新型的特征增强模块，它可以增强特征的表达能力，提高模型的性能[94-96]。给定输入数据 $X \in R^{c\times h\times w}$，$h$ 和 w 表示输入数据的高度和宽度，c 表示输入数据的通道数，生成 n 个特征图的任何卷积运算可以表示为：

$$Y = X \cdot \omega + b \tag{4-5}$$

其中 $Y \in R^{n\times h'\times w'}$ 表示输出是高度为 h'、宽度为 w' 的 n 个特征图，$\omega \in R^{c\times k\times k\times n}$ 表示卷积核大小为 $k\times k$ 的 $c\times n$ 卷积运算，b 表示偏置。不难发现，该卷积所需的 FLOPs 可以计算为 $n\cdot h'\cdot w'\cdot c\cdot k\cdot k$，这个值通常达到数十万，因为卷积核数量 n 和通道数 c 通常非常大。

因此，输入和输出特征图的大小决定了运算待优化的参数 (w 和 b)，主流的卷积操作后会生成大量的冗余，并且其中部分冗余具有相似性，因此无需使用大量参数 FLOPs 来生成冗余特征图，而输出特征图是少数原始特征图通过一些廉价线性运算转换的而成的“Ghost”。这些原始特征图通常由普通卷积核生成，是具有通道信息较少的特征图，Ghost 模块生成 m 个原始特征图 $Y \in R^{n\times h'\times w'}$ 提取过程如公式 (4-6)所示：

$$Y' = X \cdot \omega' \tag{4-6}$$

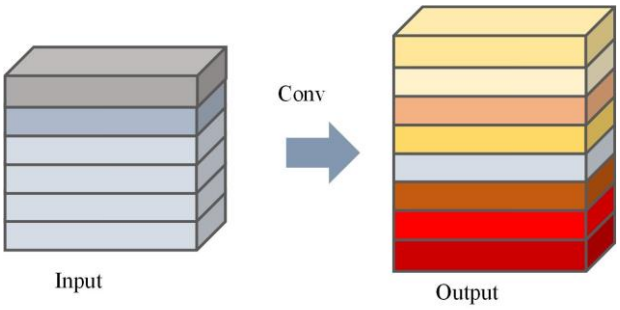
其中 $\omega' \in R^{c \times k \times k \times m}$ 为卷积核，$m \le n$。为确保与输出特征图的大小相同，卷积核大小、步长和填充等其他超参数与普通卷积一致，为进一步获得所需的 n 个特征图，对 Y' 中的每个原始特征使用一系列廉价线性运算，生成 s 个 Ghost 特征图如式 (4-7)示：

$$Y_{ij} = \phi_{i,j}\left(Y_i'\right), \forall i = 1,2,3,...s \tag{4-7}$$

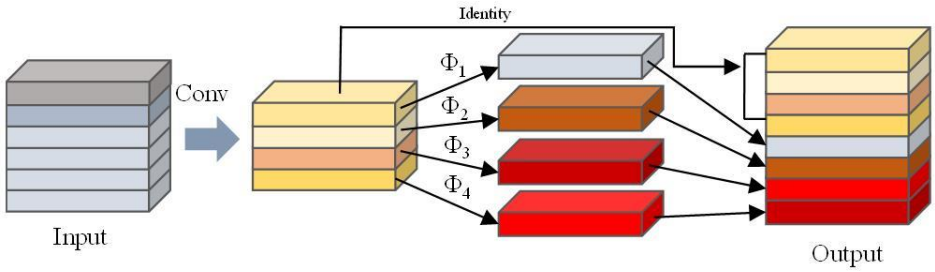
其中 Y_i' 表示 Y' 中第 i 个特征图，$\phi_{i,j}$ 表示第 i 个线性运算，生成第 j 个 Ghost 特征图，将生成的特征图与原始卷积生成的特征图直接拼接即可得到所需的特征图。当使用 Ghost 模块后，将线性卷积核大小设置为 $d \times d$，则其运算量如公式 (4-8)所示：

$$\begin{aligned} C_s &= \frac{c \cdot k \cdot k \cdot n \cdot h' \cdot w'}{\frac{n}{s} \cdot h' \cdot w' \cdot c \cdot k \cdot k + (s-1) \cdot h' \cdot w' \cdot d \cdot d} \\ &= \frac{c \cdot k \cdot k}{\frac{1}{s} \cdot c \cdot k \cdot k + \frac{(s-1)}{s} \cdot d \cdot d} \approx \frac{s \cdot c}{s + c - 1} \approx s \end{aligned} \tag{4-8}$$

如下图 4-8 所示，新的模块成功地用更少的参数和计算获得了更多的特征图，其中，Φ表示廉价的线性运算，Identity 表示恒等映射，GhostNet 将原始卷积层分为两部分：首先使用较少的卷积核来生成原始特征图，然后通过廉价的线性变换操作以高效产生更多幻影特征图，最后将两部分特征图拼接，得到最终的特征图。C3 模块是一种轻量级的卷积神经网络模块，它可以有效地减少模型的参数数量和计算量，同时保持较高的精度。



(a) 普通卷积



(b) Ghost 卷积

图 4-8 普通卷积与 Ghost 卷积

本节引入 GhostNet 模块改进 YOLOv5 的主干网络和颈部网络，C3 结构和 GhostNet 模块进行融合，具体改进如下图 4-9 所示。

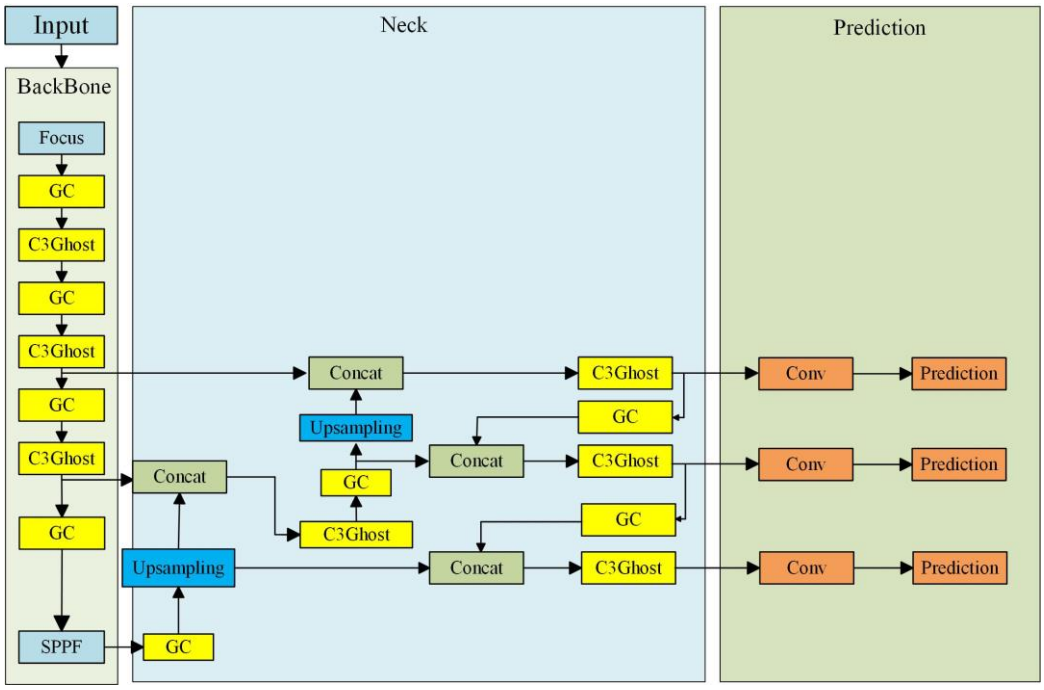


图 4-9 基于 GhostNet 的 YOLOv5 算法结构图

4.2 三种方案实验对比分析

本章对 YOLOv5 算法分别引入 MobileNetv3、ShuffleNetv2 和 GhostNet 三种方法来减少参数量，优化模型。同时，针对三种改进后的模型在同一实验环境，同一数据集下进行训练，训练超参数设置一致，对训练结果进行整理，具体结果如表 4-2 所示。

表 4-2 三种改进后的模型性能对比

模型	参数量(G)	权重大小(M)	FPS	权重压缩量(%)	mAP@0.5
YOLOv5	7.11	14.6	76	0	93.6
YOLOv5+MobileNetv3	3.6	7.9	78	45.9	80.5
YOLOv5+ShuffleNetv2	3.8	8.5	73	41.2	87.8
YOLOv5+GhostNet	3.7	8.3	82	43.2	90.6

通过上表对比实验结果可知，三种改进后的模型中，MobileNetv3 的压缩量最高，达到了 45.9%，参数量最小，仅为 3.6G。在三种轻量化后的模型中，精度和 FPS 最高的为 GhostNet。将 MobileNetv3 应用于 YOLOv5 网络后，相对于原网络，参数量下降了 49.36%；将 ShuffleNetv2 应用于 YOLOv5 网络后，参数量下降了 46.55%，但检测速度略有下降。而加入 GhostNet 后，相对于其他轻量化网络，参数量和权重分别下降了 47.96%和 43.2%，同时检测速度有显著提升，上升了 6.0fps。尽管检测精度下降了 3.0%，这表明 GhostNet 成功地减少了原模型的计算量和参数量。三种轻量化网络的 mAP 对比如下图 4-10 所示。加入 GhostNet 后，网络精度下降较少，且精度明显高于其余轻量化网络。为了降低轻量化后的网络在实时检测中的错误率，因此本章选取 GhostNet 作为 YOLOv5 轻量化主干网络。

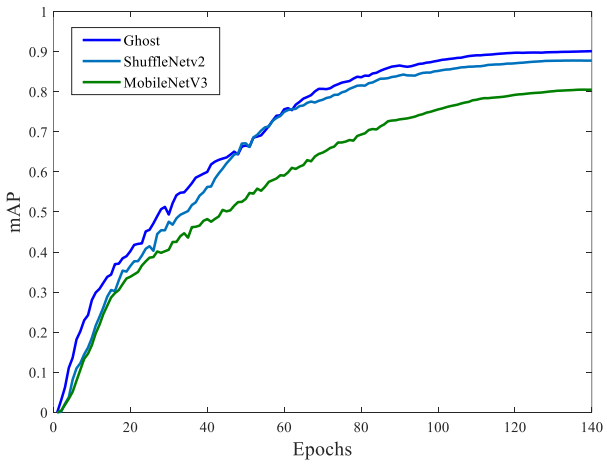


图 4-10 不同轻量化模型对比

4.3 GhostNet 与 CBAM 级联结构

由于大量的卷积层堆叠导致卷积神经网络中存在大量的冗余特征图，这将导致参数量增加及计算成本增大[97]。通过前文的分析可知，对 YOLOv5 进行了

三种轻量化网络的仿真，结果显示 GhostNet 在本文所选取的数据集上的应用效果最为显著。GhostNet 能够生成更多的特征图，并且通过廉价的线性运算达到与原始卷积相同的效果。因此，选取 GhostNet 作为 YOLOv5 的轻量化网络具有显著的优势，能够大大降低模型的参数量，并且通过少量的计算提取大量的特征，从而解决了特征提取中存在的冗余问题。此外，GhostConv 在准确率等方面也超越了 MobileNetV3 和 Shufflenetv2 等网络，进一步证明了其作为轻量化网络的优越性。

虽然 GhostNet 已有效地减少模型的参数量和计算量，但无法保证得到的特征图与标准卷积操作的特征图完全一致，易导致特征图的关键信息缺失，因此，本文设计了一种自适应 Ghost 卷积网络 (Self-Adaptation GhostNet, SAGN)，并将 SAGN 作为 YOLOv5 模型的特征提取网络，使 YOLOv5 模型更加紧凑，仿真结果表明，SAGN 可以将 YOLOv5 模型的参数量减少到原来的 53.0%，mAP 增加了 1.2%。其结构如图 4-11 所示，具体思想是对 CBAM 和 GhostNet 进行级联操作，由前文可知，CBAM 是一种轻量级的通用模型，可以对特征图进行自适应特征优化，卷积层的输出将首先通过通道关注模块得到加权结果，然后通过空间关注模型得到最终的加权结果。注意力机制可以在特征图的卷积过程中发现目标对象后聚焦于目标对象，并抑制特征图中的无关特征信息。

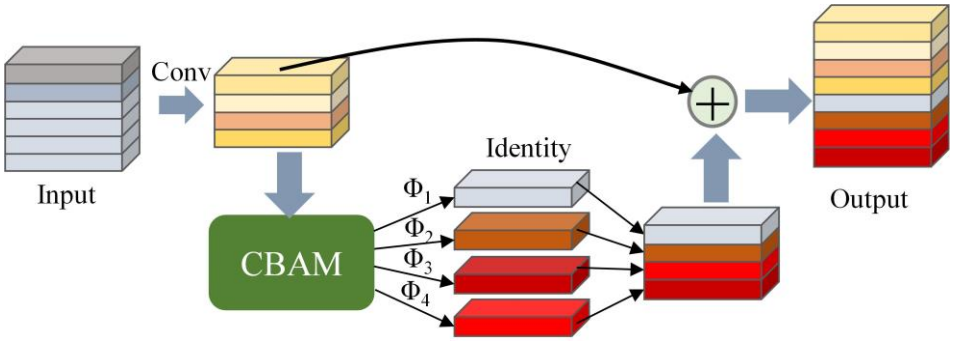


图 4-11 CBAM 与 GhostNet 级联结构

4.4 改进上采样模块

在特征金字塔生成过程中，传统的特征金字塔网络主要使用双线性插值对高阶特征图进行上采样，即最近邻上采样和线性上采样[98]。这些方法主要关注确定上采样核时的空间像素位置，但在充分利用特征图的语义信息方面存在局

限，导致感知域受限。另一种上采样技术是反卷积，但它存在一个关键缺点，它在整个图像上应用统一的核，忽略了局部变化，并涉及大量参数。因此，双线性插值只对相邻四个像素进行插值，不能完全捕捉到局部细节的变化，只在像素值较大变化的区域，双线性插值可能会导致信息丢失，在图像的边界区域，双线性插值可能会产生边界效应，导致插值结果不准确。

为解决上采样中语义关联不足的问题，对算法改进了上采样方法，引入CARAFE (Context-Aware Reassembly and Feature Enhancement，CARAFE)[99]作为最近邻上采样的替代方法。CARAFE 是一种基于像素级语义分割网络的上采样方法，通过学习像素之间的关联来重建特征图，从而获得更高分辨率的特征图。其包含两个模块：上采样核预测模块和内容感知模块两个关键模块，它们解决了传统卷积神经网络在处理图像细粒度语义信息时的信息丢失、特征混淆、计算效率和参数数量等问题。通过学习动态的卷积核保留信息，引入非线性的特征重组操作减少特征混淆，设计轻量级操作提高计算效率，并考虑参数数量优化模型性能。其主要创新之处在于引进一种全新的上下文自适应重组操作，以优化卷积网络在上采样阶段的特征表达。传统的上采样操作，如转置卷积或双线性插值，往往会导致信息的丢失和模糊，特别是在需要保留细节信息的任务中。为了解决这一问题，CARAFE 在上采样后引入了上下文自适应重组，通过对特征进行重新组装，以更好地保留和细化语义信息。

具体来说，CARAFE 的主要特点包括：

Context 自适应重组操作： 这是 CARAFE 的核心，旨在通过自适应聚合临近位置的特征来增强上采样后的特征表达。该操作通过考虑特征点周围的上下文信息，动态调整特征的权重，以更好地捕捉细粒度的语义信息。

空间自适应卷积： CARAFE 引入了空间自适应卷积，这是一种新颖的卷积操作。与传统的固定大小卷积核不同，空间自适应卷积可以根据不同区域的特征分布，动态地调整卷积核的大小和形状。这种特性使得 CARAFE 能够更好地适应图像中不同尺度和分辨率的信息，提高了模型的灵活性。

上下文感知的重组： 通过引入上下文感知的重组，CARAFE 能够更好地保持特征之间的空间关系，避免信息的错位和混淆。这有助于改善模型在处理具有复杂语义结构图像时的性能。

本章引用轻量化采样算子 CARAFE 替换 YOLOv5 中的最邻元插值，该算子具有通用、轻量级、冗余度小、特征融合能力强的高效算子，CARAFE 上采样算法结构如图 4-12 所示。

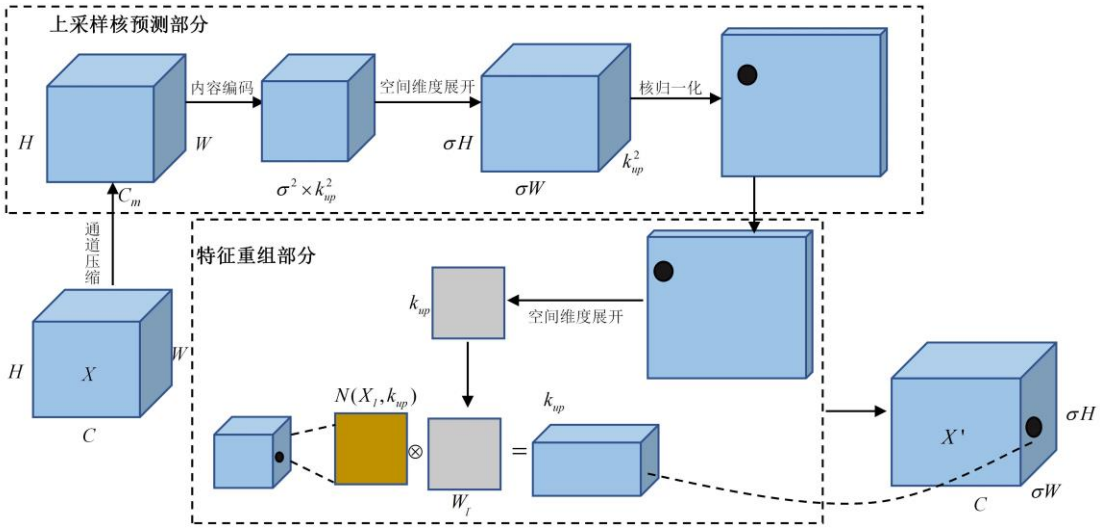


图 4-12 CARAFE 上采样算法结构

其中上采样核预测部分利用可分离卷积生成卷积核，用于对输入特征图进行下采样并与下采样的特征图进行卷积。最初，大小为 $H \times W \times C$ 的输入特征图通过 1 ×1 的卷积层进行通道压缩分解。这种压缩将通道数减少到C_m，从而减少了用于后续操作的参数数量。然后，使用具有维度的$\sigma H \times \sigma W \times k_{up}^2$卷积层对压缩的特征图进行卷积。这个卷积层负责进行线性内容编码。此外，特征图在空间维度上扩展，以获得大小为$\sigma H \times \sigma W \times k_{up}^2$的上采样核。该过程有助于学习和重建卷积核内的像素相关性。生成卷积核依赖于两个关键的超参数k_{up}和σ，其中k_{up}为确定重建卷积核的大小以及 σ代表重建因子。

特征重组部分负责从前一层重建特征图以实现上采样。首先，对下采样的特征图应用像素洗牌操作[100]。然后，通过可分离卷积将其转换为更高分辨率的特征图。最后，使用 Softmax 函数对生成的特征图进行归一化，并将其分成较小的块。对于上部特征图中的每个像素及其相应的较小块进行点积运算，得到大小为$\sigma H \times \sigma W \times C$ 的输出特征图。

4.5 改进损失函数

损失函数可以量化模型对结果的预测程度，并估计检测模型与真实数据之间的差距。选择适当的损失函数有助于开发更好的模型并加速训练过程中的收敛。通过计算损失函数的梯度并进行反向传播，更新网络的权值。因此，选择有效的损失函数可以显著优化神经网络的性能。YOLOv5 在 DIoU Loss 的基础上增加了一个影响因子，通过考虑引入中心点距离和高宽比，将完全交并比损失函数作为损失函数，具体计算方法如下：

$$L_{CIoU} = 1 - CIoU \tag{4-9}$$

$$CIoU = IoU - \frac{A^2}{B^2} - \frac{16\left(\tan^{-1}\frac{w^{gt}}{h^{gt}} - \tan^{-1}\frac{w^{p}}{h^{p}}\right)^4}{\pi^4\left[\left(1-IoU\right)+v\right]} \tag{4-10}$$

式中：A 表示预测框与目标框中心点距离，B 表示最小外接矩形对角线的长度，$\frac{w^{gt}}{h^{gt}}$ 和 $\frac{w^{p}}{h^{p}}$ 分别是预测框和目标框的高宽比。

尽管 L_{CIoU} 增加了一个影响因子来考虑引入中心点距离和高宽比，但是仍然无法有效地解决边界回归中的样本不平衡问题，可能会导致收敛缓慢而且回归不准确，因此引入 Focal EIoU 损失函数可有效解决以上问题，如下所示：

$$L_{FocalEIoU} = IoU^{\gamma} L_{EIoU} \tag{4-11}$$

$$\begin{aligned} L_{EIoU} &= LIoU + L_{dis} + L_{asp} \\ &= 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \frac{\rho^2(w, w^{gt})}{{c_w}^2} + \frac{\rho^2(h, h^{gt})}{{c_h}^2} \end{aligned} \tag{4-12}$$

式中：γ 表示控制异常值抑制程度的参数，c_w 和 c_h 分别是预测框和目标框最小外界框的宽和高。

Focal EIoU 损失函数将纵横比的损失项拆分成预测的宽度和高度分别与最小外接框宽度和高度之间的差异，从而提高了模型的收敛速度和回归精度[101,102]。然而，Focal EIoU 的聚焦机制是静态的，未充分利用非单调动态聚焦机制的潜力。由于训练数据不可避免地包含低质量示例，几何因素会加重对这些低质量示例的惩罚，进而降低了模型的泛化性能。当 anchor box 与目标框高度

重叠时，损失函数会减弱几何因素的惩罚。因此，本文在原有基础上进行了改进，引入了构建非单调聚焦系数的方法，并将 Wise-IoU[103]引入作为损失函数，具体计算如下所示。

$$L_{WIoU} = rR_{W_{IoU}} L_{IoU} \tag{4-13}$$

$$R_{WIoU} = \exp\left(\frac{\left(x-x_{gt}\right)^2+\left(y-y_{gt}\right)^2}{\left(c_w^{\ 2}+c_h^{\ 2}\right)^*}\right) \tag{4-14}$$

$$r = \frac{\beta}{\delta\alpha^{\beta-\delta}} \tag{4-15}$$

$$\beta = \frac{L^*_{IoU}}{L_{IoU}} \tag{4-16}$$

式中：c_w 和 c_h 是最小外接矩形的宽和高，β 表示锚框的离群度，离群度小，意味着锚框的质量高，为防止 R_{WIoU} 产生阻碍收敛的梯度，将 c_w 和 c_h 从计算中分离出来(*表示此操作)，可有效消除阻碍收敛的因素，由于 L_{IoU} 是动态的，锚框的离群度也是动态的，这使得 Wise-IoU 避免了将长宽比纳入损失函数。通过估计锚定框的离群度，并定义动态的非单调聚焦机制，对质量较高的锚定框分配了适度 β 的小梯度增益，允许中等质量的锚定框成为回归过程的焦点。同时，对于质量较低的锚定框，给予了较大 β 的最小梯度增益，从而有效地降低了低质量情况下锚定框回归的惩罚。Focal EIoU 损失函数训练时收敛曲线如图 4-13 所示。

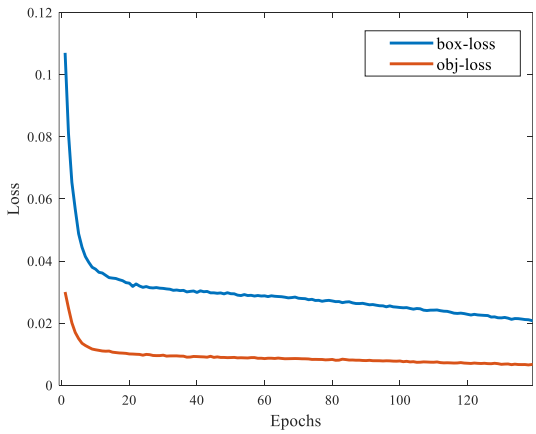


图 4-13 损失函数

在图 4-13 中，box-loss 为定位损失，用于计算锚框和相应的校准分类是否正确。当 box-loss 值较低时，意味着模型能够准确预测目标的位置；obj-loss 为置信度损失，表示模型是否成功地检测到了目标。当 obj-loss 值较低时，模型将一个物体误判为背景或未检测到目标时，obj-loss 会增加，反之，当模型正确地检测到目标时，该损失会减小。在图 4-13 中，box-loss 和 obj-loss 的值都很小，收敛速度很快。

4.6 超参数设置

本文从中国交通标志标线的公共数据集 TT100k 中选取了 9600 幅不同交通场景图像作为训练和测试数据集，按照 8:2 的比例进行划分。数据集包含 34 个类别。实验使用的硬件平台包括 Intel Core i7-8358P 处理器和 Nvidia GeForce GTX 3090 图形处理器。具体参数见表 4-3 所示。

表 4-3 有关训练参数

参数	数值
图像大小	640 ×640
batch size	16
Epoch	150
初始学习率	0.0001
Weight decay	0.0005
优化器	SGD
PyTorch	1.8.1
cuDNN	11.1
Python	3.8

4.7 实验结果与分析

4.7.1 对比实验分析

为了证明所提出网络在交通标志识别领域中的优势，本文将所提出的方法与其他网络进行了比较。Tan 等人提出的 Efficientdet 网络近年来取得了优异的成果，Efficientdet 网络训练的参数只有 3.752G 个参数，其权重只有 15.15M；Andrew 等人提出的 MobileNet-SSD 被用作比较实验之一；MobileNet 被广泛应用于移动终端的神经网络主干，对神经网络移植到移动终端具有里程碑意义。此外，YOLOV3 还将 Darknet53 网络引入骨干网。与之前的 YOLOv2 和 YOLOv1 相比，YOLOV3 具有更强大的功能和更高的精度，有助于提高模型的特征提取能力，从而提高目标的检测性能。

表 4-4 对比实验结果

模型	参数量(G)	权重大(M)	FPS	mAP@0.5
YOLOv5	7.11	14.6	76	93.6
MobileNet-SSD	25.06	118	22	32.0
Efficientdet	3.75	15.1	26	57.8
YOLOX	69.0	9.0	59	68.6
YOLO-GCC	3.7	8.3	85.8	91.9

由表 4-4 可知，与 MobileNet-SSD、Efficientdet、YOLOX 网络相比，本章所提出的 YOLO-GCC 轻量化算法具有参数量仅为 3.7G，权重仅为 8.3M，FPS 增加到 85.8fps， mAP@0.5 提升至 91.9%的优势，因此，YOLO-GCC 模型在所有模型中表现和准确性最佳。YOLO-GCC 模型的训练结果如图 4-14 所示。评估图表显示在 100 个 epochs 后，精确度、召回率、mAP、训练损失和验证损失的值都趋于收敛。

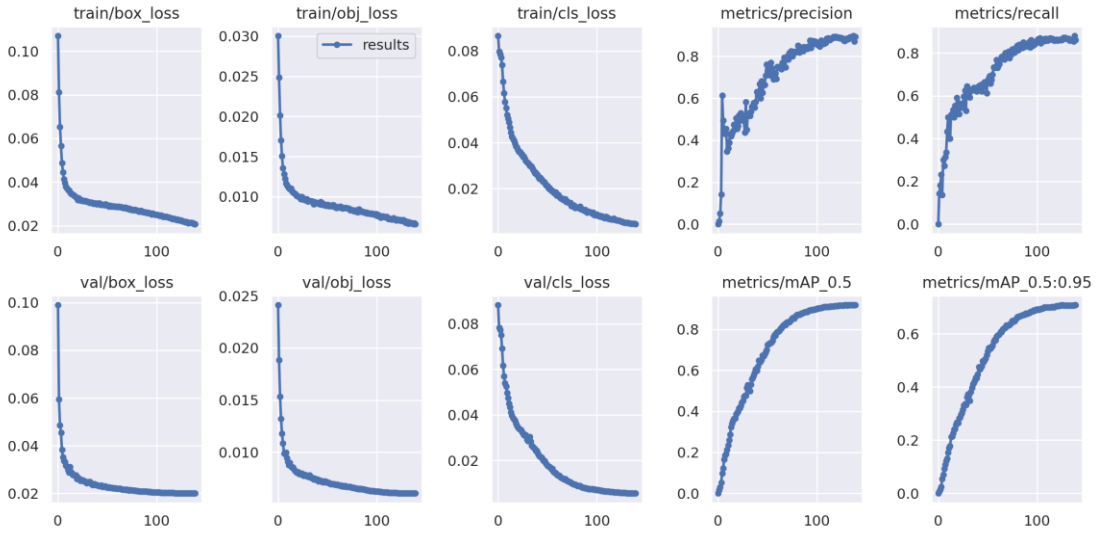


图 4-14 YOLO-GCC 模型的训练结果

4.7.2 消融实验分析

为了分析本文对 YOLOv5 网络改进方法的有效性，针对主要改动部分在 TT100k 数据上进行消融实验，具体做法是在 YOLOv5 网络中分别加入 GhostConv 和 SAGN，探究 Ghost 卷积对改进前后网络的影响。再根据 YOLOv5 网络骨干网络改进的基础上，依次加入改进的损失函数和上采样模块。其中，G 表示 GhostConv，C 表示 CARAFE 上采样模块，W 表示 WIoU 损失函数，√意味着在 YOLOv5 网络中添加了对应的模块。

通过对比分析：(1) YOLOv5＋GhostConv；(2) YOLOv5＋SAGN；(3) YOLOv5＋SAGN+CARAFE 上采样模块；(4) YOLOv5＋SAGN+CARAFE 上采样模块+ WIoU 损失函数；具体的消融结果见表 4-5 和曲线对比如图 4-15、图 4-16、图 4-17 和图 4-18。

表 4-5 消融实验结果

Models	G	SAGN	C	W	P	R	mAP@0.5	mAP@0.5:0.95	FPS
实验 1	√				89.9	84.6	90.1	69.5	82
实验 2		√			90.4	85.8	91.3	70.8	73.5
实验 3		√	√		89.3	87.8	91.4	70.7	70.9
实验 4		√	√	√	91.9	89.06	91.9	70.8	85.8

下图为四种改进后的的模型曲线效果对比图。

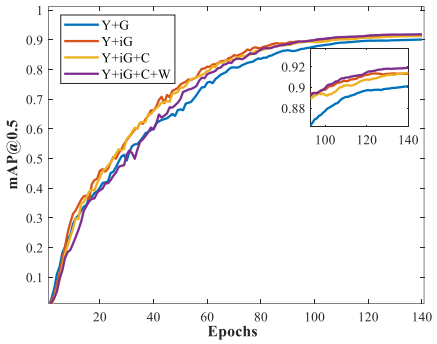


图 4-15 改进后的模型 mAP@0.5 对比图

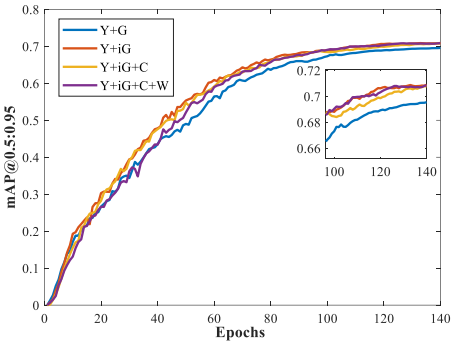


图 4-16 改进后的模型 mAP@0.5:0.95 对比图

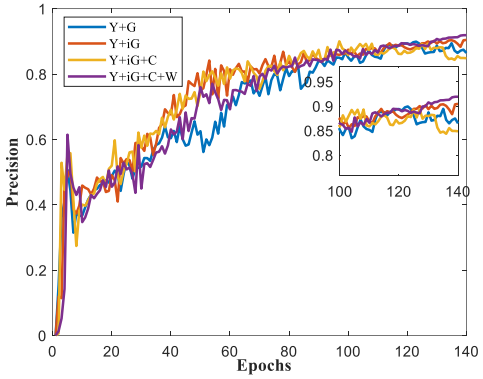


图 4-17 改进后的模型 Precision 对比图

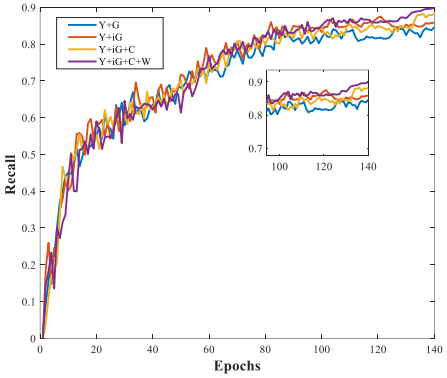


图 4-18 改进后的模型 Recall 对比图

从表 4-5 和图 4-15、图 4-16、图 4-17 和图 4-18 可知，改进后的模型曲线和改进前的模型曲线对比曲线波动较小，模型对添加的 Ghost 卷积进行改进后，精确率、召回率和平均精确率均有所提很高，在此基础上，引入 CARAFE 网络作为上采样模块，虽然精确率略有下降，但召回率和平均精确率均有所提高，说明加入后的模块能更好地捕获小目标的特征信息，再引入 WIoU 作为模型的

损失函数，引入后模型的精确率、召回率和平均精确率均得到有效提升。将三种改进的策略结合后效果最佳，即改进后的 YOLOv5 模型 mAP 和检测速度 FPS 分别提高了 1.8 个百分点和 3.8 fps，进一步验证了本文方法的可行性。

除了直观的损失性能之外，本文还绘制了分类混淆矩阵，如图 4-19 所示。混淆矩阵用于总结分类结果并表示准确性评估矩阵，颜色越深，目标的识别率越高。

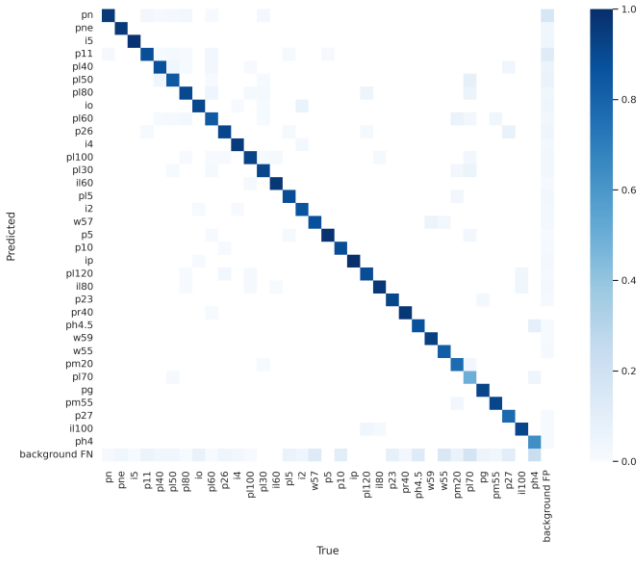


图 4-19 YOLO-GCC 在训练集上的混淆矩阵

4.7.3 部分检测结果示例

图 4-20 部分实例检测结果

为了验证本文提出的算法在交通标志检测中的性能，本文选择了不同尺度的交通标志图像，包括一些人眼难以识别的交通标志图像，如图 4-20 所示。由第 2 章第 2.3.2 小节可知，pl30 代表限速 30km/h，pl60 代表限速 60 km/h，pl80 代表限速 80 km/h，pl120 代表限速 120 km/h，il60 代表最低车速 60 km/h，il80 代表最低车速 80 km/h，il100 代表最低车速 100 km/h，pn 代表禁止停车，ph4.5 代表限高 4.5m。测试结果表明，该算法能够有效地检测出道路上的小尺度目标，且置信度高。

图 4-21 复杂环境实例检测结果

在夜晚、雨天和大雾等复杂环境下，车载设备所拍摄的交通标志图片清晰度较低，从而导致交通标志检测与识别所受影响较大。图 4-21 为对复杂天气行车记录仪拍摄的交通标志进行检测，结果表明，本文提出的算法可以精准定位和检测部分复杂环境下的交通标志。

4.8 本章小结

本章对比了 MobileNetV3、Shufflenetv2 和 GhostConv 三种轻量化网络在 YOLOv5 中的改进效果，最终选取 GhostNet 做为 YOLOv5 算法的轻量化网络，再接着对轻量化 GhostNet 网络进行改进，融合 GhostNet 网络和 CBAM 注意力机制级联结构来优化 YOLOv5 主干特征提取网络，提出一种轻量型的道路交通标志检测模型，降低了模型复杂度，提高了交通标志的检测速度。采用 WIoU 作为边框损失函数，并在在特征融合层中引入 CARAFE 上采样模块以更好的恢复检测目标局部细节。训练结果表明，改进后的 YOLOv5 模型，mAP 提升 1.8%，为 91.9%，权重下降为原来的 65.1%，参数量下降为原来的 58.8%，检测速率提升了 8.6fps，本章提出的轻量化算法相对于其他算法更易于移植到移动设备端，且检测精度明显优于大部分主流算法，可以在车辆智能辅助驾驶中进行实际应用。

第 5 章 总结与展望

5.1 总结

本文基于 YOLOv5 算法，针对目前交通标志识别算法存在检测精度低，模型尺寸大，检测速度慢等问题，提出了一种基于深度学习的轻量化交通标志识别算法。传统方法主要依赖于手工提取特征和经典的计算机视觉技术，但随着深度学习技术的兴起，基于卷积神经网络的方法在交通标志检测领域取得了显著的成就。深度学习方法通过端到端的训练，能够自动学习图像中的特征，从而在复杂多变的道路环境中更为准确地检测和识别交通标志。本文的主要工作总结如下：

(1) 针对当前交通标志识别技术不能满足智能辅助驾驶系统实时性和准确性要求的问题，提出了一种改进的 YOLOv5 算法。将卷积注意力模型和 CSP1_3CBAM 模型融合在一起，建立了一个新的 CSP1_3CBAM 模型。选择 ACONC 作为 YOLOv5 的激活函数，从而提高了模型对有用信息特征的提取能力。最后通过在 TT100K 交通标志数据集上的实验表明，改进后的 YOLOv5 算法的损失函数值比改进前明显降低。改进后的 YOLOv5 模型在 TT100K 数据集交通标志检测方法与其他检测方法相比，mAP 提高了 6.2%、FPS 每秒提升了 3.54 帧。

(2) 对 YOLOv5 网络进行轻量化处理，提出一种轻量化的 YOLO-GCC 交通标志检测模型。首先分别采用 MobileNetV3、Shufflenetv2 和 GhostNet 三种网络作为 YOLOv5 的特征提取网络，分别对网络进行训练验证，结果表明 GhostNet 作为特征提取网络，更能明显降低网络模型的权重，因此选取 GhostNet 做为轻量化主干网络，接着融合 GhostNet 网络和 CBAM 注意力机制，引入 WIoU 作为边框损失函数，并在特征融合层中引入 CARAFE 上采样模块。最后在 TT100k 交通标志数据集测试结果表明，改进后的 YOLOv5 模型，mAP 提升 1.8%，为 91.9%，权重下降为原来的 65.1%，参数量下降为原来的 58.8%，FPS 每秒提升了 8.6 帧。

5.2 展望

本文提出了一种轻量化的 YOLO-GCC 交通标志检测模型。该模型选取 GhostNet 作为特征提取网络，并融合 CBAM 注意力机制，引入 WIoU 作为边框损失函数，以及 CARAFE 上采样模块进行特征融合，该算法在实现轻量化的同时取得了显著的性能提升，但仍存在一些可以进一步改进的方向。

(1) 本文所使用的数据集为中部分实例数量较少，且大多数场景为白天和晴天，考虑到复杂天气情况下的交通标志识别，因此可以整合其他来源的交通标志数据集，如公开数据集、自行采集数据等，增加交通标志的种类和数量，以增加数据样本的多样性和丰富性，从而提升模型的泛化能力。

(2) 注意机制仅和 GhostNet 网络进行融合，未来可以探索更多先进的轻量化网络结构和注意力机制融合，以进一步提升模型的特征提取能力和鲁棒性。

(3) 交通标识牌易被污损和遮挡，本文虽在特征提取网络中融入了注意力机制，但仍存在部分遮挡面积较大和褪色的交通标志无法准确检测，因此未来可以考虑引入更多的先验知识或者上下文信息，例如道路场景信息等，以提升模型在复杂环境下的适应能力。

(4) 随着自动驾驶技术的发展和应用场景的多样化，考虑整合与交通标志相关的其他领域数据，如道路标线、车辆、行人等信息，以提供更丰富的语境信息，帮助模型更好地理解交通环境，提高交通标志检测的准确性和实用性，实现更加全面和智能的交通场景理解和决策。

参考文献

[1] Salah Zaki P, Magdy William M, Karam Soliman B, et al. Traffic Signs Detection and Recognition System using Deep Learning[J]. arXiv e-prints, 2020: arXiv: 2003.03256.

[2] Ai W, Fang D, Fu J, et al. Hopf bifurcation analysis and control of the continuum model considering the new energy vehicles effect[J]. The European Physical Journal B, 2024, 97(1): 2.

[3] Hu Y, Li Y, Huang H, et al. A high-resolution trajectory data driven method for real-time evaluation of traffic safety[J]. Accident Analysis & Prevention, 2022, 165: 106503.

[4] Li S, Zhang J, Chen Z, et al. Theoretical Analysis of Cooperative Driving at Idealized Unsignalized Intersections[J]. Tsinghua Science and Technology, 2023, 29(1): 257-270.

[5] Yu J, Ye X, Tu Q. Traffic sign detection and recognition in multiimages using a fusion model with YOLO and VGG network[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(9): 16632-16642.

[6] Mishra J, Goyal S. An effective automatic traffic sign classification and recognition deep convolutional networks[J]. Multimedia Tools and Applications, 2022, 81(13): 18915-18934.

[7] Kim T, Park S, Lee K. Traffic Sign Recognition Based on Bayesian Angular Margin Loss for an Autonomous Vehicle[J]. Electronics, 2023, 12(14): 3073.

[8] Dewi C, Chen R C, Liu Y T, et al. Synthetic Data generation using DCGAN for improved traffic sign recognition[J]. Neural Computing and Applications, 2022, 34(24): 21465-21480.

[9] Gan Y, Li G, Togo R, et al. Zero-shot traffic sign recognition based on midlevel feature matching[J]. Sensors, 2023, 23(23): 9607.

[10] 高涛, 邢可, 刘占文, 等. 基于金字塔多尺度融合的交通标志检测算法[J]. 交通运输工程学报, 2022, 22(3): 210-224.

[11] Ouyang Z, Niu J, Ren T, et al. MBBNet: An edge IoT computing-based traffic light detection solution for autonomous bus[J]. Journal of Systems Architecture, 2020, 109: 101835.

[12] 王海, 王宽, 蔡英凤, 等. 基于改进级联卷积神经网络的交通标志识别[J]. 汽车工程, 2020, 42(9): 1256-1262.

[13] 赵子婧, 刘宏哲, 曹东璞. 基于 Libra R-CNN 改进的交通标志检测算法[J]. 机械工程学报, 2021, 57(22): 255-265.

[14] 陈飞,刘云鹏,李思远.复杂环境下的交通标志检测与识别方法综述[J].计算机工程与应用,2021,57(16):65-73.

[15] Zhu Y, Zhang C, Zhou D, et al. Traffic sign detection and recognition using fully convolutional network guided proposals[J]. Neurocomputing, 2016, 214: 758-766.

[16] 张淑芳,朱彤.基于残差单发多框检测器模型的交通标志检测与识别[J].浙江大学学报(工学版), 2019, 53(05):940-949.

[17] Huang Z, Yu Y, Gu J, et al. An efficient method for traffic sign recognition based on extreme learning machine[J]. IEEE transactions on cybernetics, 2016, 47(4): 920-933.

[18] Abdel-Salam R, Mostafa R, Abdel-Gawad A H. RIECNN: real-time image enhanced CNN for traffic sign recognition[J]. Neural Computing and Applications, 2022, 34(8): 6085-6096.

[19] Dalai R, Senapati K K. A MASK-RCNN Based Approach Using Scale Invariant Feature Transform Key points for Object Detection from Uniform Background Scene[J]. Advances in Image and Video Processing, 2019, 7(5): 01-08.

[20] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Ieee, 2005, 1: 886-893.

[21] Kerim A, Efe M Ö. Recognition of traffic signs with artificial neural networks: A novel dataset and algorithm[C]//2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC). IEEE, 2021: 171-176.

[22] 冯润泽,江昆,于伟光,等.基于两阶段分类算法的中国交通标志牌识别[J].汽车工程,2022,44(03):434-441+448.

[23] Sapijaszko G, Alobaidi T, Mikhael W B. Traffic sign recognition based on multilayer perceptron using DWT and DCT[C]//2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, 2019: 440-443.

[24] Weng H M, Chiu C T. Resource efficient hardware implementation for real-time traffic sign recognition[C]//2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018: 1120-1124.

[25] Lim X R, Lee C P, Lim K M, et al. Recent advances in traffic sign recognition: approaches and datasets[J]. Sensors, 2023, 23(10): 4674.

[26] 梁敏健,崔啸宇,宋青松,等.基于 HOG-Gabor 特征融合与 Softmax 分类器的交通标志识别方法[J].交通运输工程学报,2017,17(03):151-158.

[27] Tan M, Pang R, Le Q V. Efficientdet: Scalable and efficient object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10781-10790.

[28] Ruta A, Li Y, Liu X. Robust class similarity measure for traffic sign recognition[J]. IEEE Transactions on Intelligent Transportation Systems, 2010, 11(4): 846-855.

[29] Mishra J, Goyal S. An effective automatic traffic sign classification and recognition deep convolutional networks[J]. Multimedia Tools and Applications, 2022, 81(13): 18915-18934.

[30] Bay H, Tuytelaars T, Van Gool L. Surf: Speeded up robust features[C]//Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9. Springer Berlin Heidelberg, 2006: 404-417.

[31] Ren F X, Huang J, Jiang R, et al. General traffic sign recognition by feature matching[C]//2009 24th international conference image and vision computing New Zealand. IEEE, 2009: 409-414.

[32] He S, Chen L, Zhang S, et al. Automatic recognition of traffic signs based on visual inspection[J]. IEEE Access, 2021, 9: 43253-43261.

[33] Tabernik D, Skočaj D. Deep learning for large-scale traffic-sign detection and recognition[J]. IEEE transactions on intelligent transportation systems, 2019, 21(4): 1427-1440.

[34] Kumar M, Ramalingam S, Prasad A. An optimized intelligent traffic sign forecasting framework for smart cities[J]. Soft Computing, 2023, 27(23): 17763-17783.

[35] Zhang K, Sheng Y, Li J. Automatic detection of road traffic signs from natural scene images based on pixel vector and central projected shape feature[J]. IET Intelligent Transport Systems, 2012, 6(3): 282-291.

[36] Qu S, Yang X, Zhou H, et al. Improved YOLOv5-based for small traffic sign detection under complex weather[J]. Scientific reports, 2023, 13(1): 16219.

[37] Min W, Liu R, He D, et al. Traffic sign recognition based on semantic scene understanding and structural traffic sign location[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(9): 15794-15807.

[38] Zhang J, Zhang H, Lang D, et al. Research on rainy day traffic sign recognition algorithm based on PMRNet[J]. Mathematical biosciences and engineering, 2023, 20(7): 12240-12262.

[39] Elman J L. Finding structure in time[J]. Cognitive science, 1990, 14(2): 179-211.

[40] 吴新辉,毛政元,翁谦,等.利用基于残差多注意力和 ACON 激活函数的神经网络提取建筑物[J].地球信息科学学报,2022,24(04):792-801.

[41] Dennis C, Engelbrecht A P, Ombuki-Berman B M. An analysis of activation function saturation in particle swarm optimization trained neural networks[J]. Neural Processing Letters, 2020, 52: 1123-1153.

[42] Emanuel R H K, Docherty P D, Lunt H, et al. The effect of activation functions on accuracy, convergence speed, and misclassification confidence in CNN text classification: a comprehensive exploration[J]. The Journal of Supercomputing, 2024, 80(1): 292-312.

[43] Duan F, Chapeau-Blondeau F, Abbott D. Optimized injection of noise in activation functions to improve generalization of neural networks[J]. Chaos, Solitons & Fractals, 2024, 178: 114363.

[44] Uijlings J R R, Van De Sande K E A, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104: 154-171.

[45] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[46] Arora N, Kumar Y, Karkra R, et al. Automatic vehicle detection system in different environment conditions using fast R-CNN[J]. Multimedia Tools and Applications, 2022, 81(13): 18715-18735.

[47] 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.

[48] 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. 2016: 779-788.

[49] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.

[50] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

[51] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.

[52] Ultralytics.YOLOv5[EB/OL]. [2022-4-10].

[53] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]. IEEE, Las Vegas, NV, USA, 2016: 770-777.

[54] 杨坚,钱振,张燕军,等.采用改进 YOLOv4-tiny 的复杂环境下番茄实时识别[J]. 农业工程学报,2022,38(09):215-221.

[55] Pattanaik A, Balabantaray R C. Enhancement of license plate recognition performance using Xception with Mish activation function[J]. Multimedia tools and applications, 2023, 82(11): 16793-16815.

[56] Zhang S, Lu J, Zhao H. Deep Network Approximation: Beyond ReLU to Diverse Activation Functions[J]. Journal of Machine Learning Research, 2024, 25.

[57] Zhu Z, Liang D, Zhang S, et al. Traffic-sign detection and classification in the wild[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2110-2118.

[58] Houben S, Stallkamp J, Salmen J, et al. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark[C]//The 2013 international joint conference on neural networks (IJCNN). Ieee, 2013: 1-8.

[59] Zhang J, Zou X, Kuang L D, et al. CCTSDB 2021: a more comprehensive traffic sign detection benchmark[J]. Human-centric Computing and Information Sciences, 2022, 12.

[60] 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.

[61] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.

[62] Liu S, Qi L, Qin H, et al. Path aggregation network for instance segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 8759-8768.

[63] Xu R, Tao Y, Lu Z, et al. Attention-mechanism-containing neural networks for high-resolution remote sensing image classification[J]. Remote Sensing, 2018, 10(10): 1602.

[64] 刘莫尘,褚镇源,崔明诗,等.基于改进 YOLO v8-Pose 的红熟期草莓识别和果柄检测[J].农业机械学报,2023,54(S2):244-251.

[65] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.

[66] Park J, Woo S, Lee J Y, et al. Bam: Bottleneck attention module[J]. arXiv preprint arXiv:1807.06514, 2018.

[67] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.

[68] Wei J, Wang S, Huang Q. F³Net: fusion, feedback and focus for salient object detection[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34(07): 12321-12328.

[69] Jiang P T, Zhang C B, Hou Q, et al. Layercam: Exploring hierarchical class activation maps for localization[J]. IEEE Transactions on Image Processing, 2021, 30: 5875-5888.

[70] Ding B, Qian H, Zhou J. Activation functions and their characteristics in deep neural networks[C]//2018 Chinese control and decision conference (CCDC). IEEE, 2018: 1836-1841.

[71] Sodhi S S, Chandra P. Bi-modal derivative activation function for sigmoidal feedforward networks[J]. Neurocomputing, 2014, 143: 182-196.

[72] Ma N, Zhang X, Liu M, et al. Activate or not: Learning customized activation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 8032-8042.

[73] Wen C, Yang X, Zhang K, et al. Improved loss function for image classification[J]. Computational intelligence and neuroscience, 2021, 2021: 1-8.

[74] Zhang X, Lu W, Pan Y, et al. Empirical study on tangent loss function for classification with deep neural networks[J]. Computers & Electrical Engineering, 2021, 90: 107000.

[75] 方遒,李伟林,梁卓凡,等.基于多尺度复合卷积和图像分割融合的车道线检测算法[J].北京理工大学学报,2023,43(08):792-802.

[76] Zhang R, Zheng K, Shi P, et al. Traffic sign detection based on the improved YOLOv5[J]. Applied Sciences, 2023, 13(17): 9748.

[77] Wu S, Yang J, Wang X, et al. Iou-balanced loss functions for single-stage object detection[J]. Pattern Recognition Letters, 2022, 156: 96-103.

[78] Rezatofighi H, Tsoi N, Gwak J Y, et al. Generalized intersection over union: A metric and a loss for bounding box regression[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 658-666.

[79] Liang T J, Bao H. A optimized YOLO method for object detection[C]//2020 16th International Conference on Computational Intelligence and Security (CIS). IEEE, 2020: 30-34.

[80] Gevorgyan Z. SIoU loss: More powerful learning for bounding box regression[J]. arXiv preprint arXiv:2205.12740, 2022.

[81] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.

[82] Ma L, Wu Q, Zhan Y, et al. Traffic sign detection based on improved YOLOv3 in foggy environment[C]//International conference on wireless communications, networking and applications. Singapore: Springer Nature Singapore, 2021: 685-695.

[83] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[84] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.

[85] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[86] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.

[87] Sandler M, Howard A, Zhu M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 4510-4520.

[88] 孙伟,潘森,黄恒.基于改进 YOLOv4 模型的无人机目标检测算法[J].传感技术学报,2023,36(03):456-461.

[89] Gao S, Pei Z, Zhang Y, et al. Bearing fault diagnosis based on adaptive convolutional neural network with Nesterov momentum[J]. IEEE Sensors Journal, 2021, 21(7): 9268-9276.

[90] Jin G, Zhu T, Akram M W, et al. An adaptive anti-noise neural network for bearing fault diagnosis under noise and varying load conditions[J]. IEEE access, 2020, 8: 74793-74807.

[91] Zhang X, Zhou X, Lin M, et al. Shufflenet: An extremely efficient convolutional neural network for mobile devices[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6848-6856.

[92] Ma N, Zhang X, Zheng H T, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 116-131.

[93] Zhu Y, Liu R, Hu G, et al. Accurate identification of cashmere and wool fibers based on enhanced ShuffleNetV2 and transfer learning[J]. Journal of Big Data, 2023, 10(1): 152.

[94] 张学立,贾新春,王美刚,等.安全帽与反光衣的轻量化检测：改进 YOLOv5s 的算法[J].计算机工程与应用,2024,60(01):104-109.

[95] Fan Y, Cai Y, Yang H. A detection algorithm based on improved YOLOv5 for coarse-fine variety fruits[J]. Journal of Food Measurement and Characterization, 2024, 18(2): 1338-1354.

[96] 金鑫,庄建军,徐子恒.轻量化 YOLOv5s 网络车底危险物识别算法[J].浙江大学学报(工学版),2023,57(08):1516-1526+1561.

[97] Wang X, Liu J. Vegetable disease detection using an improved YOLOv8 algorithm in the greenhouse plant environment[J]. Scientific Reports, 2024, 14(1): 4261.

[98] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.

[99] Wang J, Chen K, Xu R, et al. Carafe: Content-aware reassembly of features[C]// Proceedings of the IEEE/CVF international conference on computer vision. 2019: 3007-3016.

[100] Shi W, Caballero J, Huszár F, et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network[C] //Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 1874-1883.

[101] 彭道刚,潘俊臻,王丹豪,等.基于改进 YOLO v5 的电厂管道油液泄漏检测[J]. 电子测量与仪器学报,2022,36(12):200-209.

[102] Yang E, Tang Y, Zhang A A, et al. Policy Gradient Based Focal Loss to Reduce False Negative Errors of Convolutional Neural Networks for Pavement Crack Segmentation[J]. Journal of Infrastructure Systems, 2023, 29(1): 04023002.

[103] Tong Z, Chen Y, Xu Z, et al. Wise-IoU: bounding box regression loss with dynamic focusing mechanism[J]. arXiv preprint arXiv:2301.10051, 2023.

攻读硕士学位期间发表的学术论文目录

已发表或录用论文

[1] **郑昆明**，张荣芸，李浩然，邱天.Baja 赛车制动系统设计与分析[J].佳木斯大学学报(自然科学版).(已发表)

[2] Rongyun Zhang, **Kunming Zheng**, Peicheng Shi, Ye Mei, Haoran Li, and Tian Qiu. 2023. Traffic Sign Detection Based on the Improved YOLOv5. Applied Sciences 13.(已发表，对应本文第 3 章)

[3] Rongyun Zhang, **Kunming Zheng**, Peicheng Shi. A Lightweight Model for Detecting Traffic Signs. International Journal of Vehicle Design.(返修中，对应本文第 4 章)

致 谢

欲买桂花同载酒，终不似，少年游。

行文至此，落笔为终。心中千语，唯恐难诉。此刻，是我在安徽工程大学求学的第七年。七载时光，转瞬即逝，再回首，无数回忆涌上心头。有过车队比赛成绩的焦虑，有过课题瓶颈的惘然，有过未来生活的胆怯……这七年，有失去，也有获得。这七年，是释怀，也是成长。

高山仰止，景行行止。感谢我的恩师张荣芸老师，从本科到研究生阶段，给予我无数的指导与关怀。在您的指导下，我不仅提升了科研能力，还学会了如何更有效地解决问题和面对挑战。您为人正直，高风亮节，学术上严谨治学，精益求精，这些都深深地感染了我，让我在今后的学术道路上更加自信。在撰写论文期间，都离不开您的亲力亲为和伏案推敲，您细致的修改意见和耐心的指导让我受益匪浅。

春晖寸草，山高海深。年岁愈长，愈能看清来自农村的自己不过是踩在父母的肩膀上享受了他们未曾体验过的生活。他们虽学历不高，但坚信知识能改变命运，在农村早点工作结婚的风气下，他们仍坚定供我读本科与研究生。他们身上的朴实让我时刻以一颗谦卑之心前进，我时刻提醒自己，纵使走得再远，也别忘了陪伴自己长大的父母，感谢他们。

所爱隔山海，山海亦可平。感谢小顾，从 2014 年相识至今，整整十年。人生总会遇到黑暗与低谷，在我难过与低迷时给我鼓励与赞赏。因为有你，平凡的日子才会发光。细水流长，不负相遇，来日可期。

山水一程，三生有幸。感谢我在求学期间遇到的同学和朋友，很幸运与你们共度青春时光。

最后，前路漫漫，感谢走的很慢但一直向前的自己！

再见了，北京中路的安徽工程大学！

尚德教学 唯实惟新