空间分组内卷积轻量级目标检测算法

卢 迪[1,2†]，赵 庆[1,2]

(1. 哈尔滨理工大学 测控技术与通信工程学院，哈尔滨 150080；
2. 哈尔滨理工大学 模式识别与信息感知黑龙江省重点实验室，哈尔滨 150080)

摘 要: 针对现有轻量级目标检测算法存在检测精度不足、特征融合能力较弱及检测速度较慢等问题, 在YOLOv8n 基础上提出基于空间分组内卷积的轻量级目标检测算法. 首先, 在内卷积基础上提出一种新型空间分组内卷积 (SGWInvo), 克服内卷积空间信息建模方面的不足, 并基于 SGWInvo 进一步设计一种轻量化主干网络 SCNet 替换 YOLOv8n 主干网络; 其次, 提出一种双向路径聚合网络, 以提高多尺度目标的特征融合能力; 最后, 采用深度可分离卷积对检测头进行轻量化, 结合 YOLO2YOLO 分步训练策略, 消除 NMS 带来的推理时延. 研究包括两种检测方法: 一对多匹配的 SGWInvo-YOLO 和一对一匹配的 SGWInvo-YOYO. 在 COCO 数据集上的实验表明, 与 YOLOv8n 相比, 两种算法参数量均降低 23.3%, SGWInvo-YOLO 与之推理速度相当, mAP0.5 精度提升 3.0%; SGWInvo-YOYO 推理时延减少 10.5%, mAP0.5 精度提升 2.3%.

关键词: 空间分组内卷积；双向路径聚合网络；分步训练；YOLOv8n；轻量化；目标检测

中图分类号: TP391.41 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2025.0035

引用格式: 卢迪, 赵庆. 空间分组内卷积轻量级目标检测算法 [J]. 控制与决策, 2025, 40(10): 3127-3135.

Lightweight object detection algorithm based on SGWInvo

LU Di[1,2†]，*ZHAO Qing*[1,2]

(1. School of Measurement and Control Technology and Communication Engineering，Harbin University of Science and Technology，Harbin 150080，China；2. Heilongjiang Province Key Laboratory of Pattern Recognition and Information Perception，Harbin University of Science and Technology，Harbin 150080，China)

Abstract: To address the limitations of existing lightweight object detection algorithms, such as inadequate detection accuracy, weak feature fusion capability, and suboptimal inference speed, this paper proposes a lightweight object detection algorithm based on spatial group-wise involution, built upon the YOLOv8n framework. A novel spatial group-wise involution (SGWInvo) is introduced to enhance spatial information modeling and overcome the limitations of standard involution operations. Based on SGWInvo, a lightweight backbone network named SGWInvo and Conv Net (SCNet) is designed to replace the original YOLOv8n backbone. Additionally, a dual path aggregation network (DPAN) is proposed to enhance the feature fusion capability for multi-scale objects. Finally, depth-wise separable convolutions are adopted to lighten the detection head, and a step-by-step training strategy, YOLO2YOLO, is adopted to eliminate inference latency caused by non-maximum suppression (NMS). Two detection methods are presented: SGWInvo-YOLO, with one-to-many matching, and SGWInvo-YOYO, with one-to-one matching. Experiments on the COCO dataset show that, compared to YOLOv8n, both proposed algorithms reduce the parameter count by 23.3%. SGWInvo-YOLO achieves comparable inference speed with a 3.0% improvement in mAP0.5, while SGWInvo-YOYO reduces inference latency by 10.5% and improves mAP0.5 by 2.3%

Keywords: SGWInvo；DPAN；step-by-step training；YOLOv8n；lightweight；object detection

0 引 言

目标检测是计算机视觉的核心问题之一, 主要任务是定位图像中的目标并识别其类别, 广泛应用于无人驾驶、区域监控、医疗检测等多个领域[1]. 然而, 随着互联网的快速发展和移动设备的普及, 许多现有的目标检测模型由于依赖大量计算资源, 难以在计算资源有限的边缘设备上部署. 因此, 轻量级模型的研究成为当前研究热点之一.

目标检测算法可分为二阶算法和一阶算法. 二阶算法首先生成初步候选区域, 而后进行分类和位

收稿日期: 2025-01-09；**录用日期:** 2025-04-22.
责任编委: 张丹.
†通信作者. E-mail: ludizeng@hrbust.edu.cn.

置回归, Faster R-CNN[2] 是代表性算法. 尽管二阶算法能提供较高精度, 但由于庞大的参数量, 往往难以在资源有限的终端设备上部署, 且推理速度较慢. 一阶算法则不需要候选框, 直接利用神经网络得到分类及回归结果, 节省大量的时间. 因此, 轻量级模型的研究主要集中在一阶算法的改进上, 其中 YOLO 系列[3-8] 算法作为典型代表, 受到学术界和工业界广泛关注. 研究者们通过模块间的合理组合, 引入深度可分离卷积、注意力机制等来轻量化网络. 例如, 文献 [9] 提出的 CSP 结构在实现轻量化的同时, 能够保持较高的准确性, 降低计算成本, 并在 YOLOv4-11 中得到广泛应用. 何湘杰等[10] 将 ECA 注意力机制引入 YOLOv4-Tiny 主干网络, 并通过空洞卷积优化 SPP 结构, 实现轻量化. 周葳楠等[11] 利用改进的 ShuffleNetV2[12] 网络替换 YOLOv5 的主干网络, 并嵌入 SE 注意力模块与 Inception 结构, 提出 SA-YOLO 模型, 实现主干网络的轻量化. 金立生等[13] 采用嵌入通道注意力机制的倒残差网络, 并引入深度可分离卷积, 有效减少了特征提取网络的参数量. 张小艳等[14] 通过引入空间通道重建注意力卷积和高效多尺度注意力改进 YOLOv8n 模型, 有效减少了参数量, 提升了检测速度. Zhong 等[15] 采用 PsConv 轻量化 YOLOv8n 网络, 并设计增强型跨尺度特征融合颈部网络, 提升了不同尺度特征融合能力.

YOLOv8n 是 YOLO 系列中综合性能较为平衡的模型之一, 但仍存在一些不足, 限制了其在边缘设备上的应用. 具体原因如下: 1) 卷积网络的局限性: 作为一种基于卷积的网络, 其性能受到卷积固有特性制约. 为了提高检测精度, 通常需要使用较大卷积核或堆叠多个小卷积核增强特征表达能力, 导致参数量显著增加; 此外, 卷积网络使用共享卷积核进行信息提取, 未能充分考虑不同空间位置之间的特征差异, 从而限制其有效性和灵活性. 2) 多尺度融合问题: 颈部网络 PAN 的连接方式较为简单, 导致特征信息融合不充分, 容易造成细节信息丢失. 3) 非极大值抑制 (NMS) 问题: 在预测过程中, 需通过 NMS 去除重叠的预测框, 引入了额外的计算开销, 进而影响推理速度.

针对上述问题, 本文在 YOLOv8n 基础上提出基于空间分组内卷积的轻量级目标检测算法. 首先, 在内卷积[16] 基础上提出一种新型的空间分组内卷积 (spatial group-wise involution, SGWInvo), 克服了内卷积未充分考虑空间信息的问题, 弥补卷积神经网络因共享卷积核在特征表达上的局限性, 并基于 SGWInvo 设计一种轻量化主干网络 SCNet 替换 YOLOv8n 主干网络; 其次, 提出一种双向路径聚合网络 (DPAN) 构建颈部网络, 增强多尺度特征信息融合; 最后, 采用深度可分离卷积替代标准卷积, 以进一步轻量化检测头, 并结合 YOLO2YOLO 的分步训练策略, 消除 NMS 带来的推理延时.

1 本文算法

1.1 空间分组内卷积轻量级网络结构

网络结构如图 1 所示, 主要包括主干网络、颈部网络和检测头. 主干网络 SCNet 结合 SGWInvo 和传统卷积, 用于高效的特征提取. 颈部网络 DPAN 负责实现多尺度特征融合, 以增强模型在复杂场景中的表现. 检测头连接不同融合层, 用于检测大、中、小尺度目标. 检测方法包括两种: 一对多匹配的 SGWInvo-YOLO 和一对一匹配的 SGWInvo-YOYO.

1.2 主干网络

1.2.1 内卷积原理

文献 [16] 提出一种内卷积算法, 旨在为特征图中每个像素点生成一个专属于该像素点的内卷积核, 从而克服共享卷积核在特征表达上的局限性, 包含两个部分: 核函数生成和内卷积操作, 如图 2 所示. 在宽W高H通道数为C的特征图上 (i, j) 所在位置对应向量$X_{i,j}$, 核生成函数Φ对$X_{i,j}$进行两次全连接操作, 生成K^2G个卷积核元素, 而后重塑成G个$K\times K$大小卷积核, 其中G为超参数. 内卷积操作过程是: 将特征图上 (i, j) 所在$K\times K$邻域, 在通道维度上平均分为G段, 每段数据与对应的内卷积核作哈达玛积, 最后在通道维度上进行求和.

1.2.2 空间分组内卷积

上述内卷积计算过程仅考虑了通道维度, 未能充分利用空间信息, 因此本文对核函数的生成进行改进, 提出一种新型空间分组内卷积 (SGWInvo), 将核生成函数分为两个部分: 通道分组信息交互模块和空间信息交互模块.

1) 通道分组信息交互模块.

通道分组信息交互模块如图 3 所示. 首先采用一个全连接层进行通道信息压缩, 将输入从C维降至C/r维, 以建立全局通道关系. 随后, 结合层归一化和激活函数 ReLU, 提升模型泛化能力并加速收敛. 最后, 通道信息分组层将其分为G组, 每组包含K^2元素, 进一步重塑为G组$K\times K$大小卷积核.

2) 空间信息交互模块.

空间信息交互模块赋予内卷积核空间特征相关

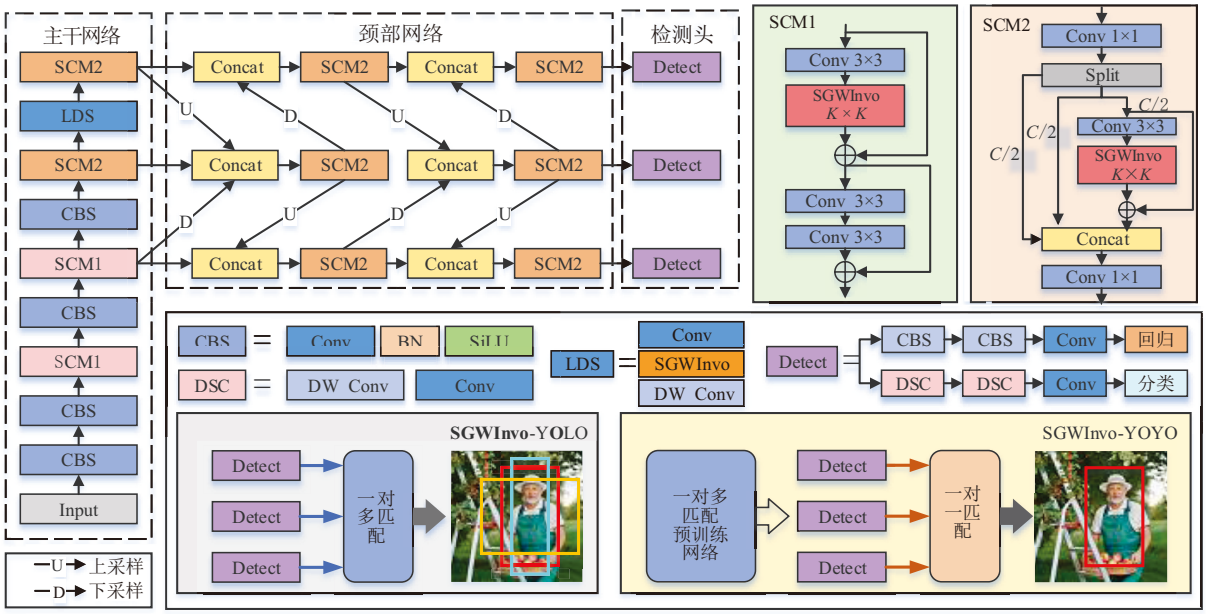


图1　网络结构及检测方法

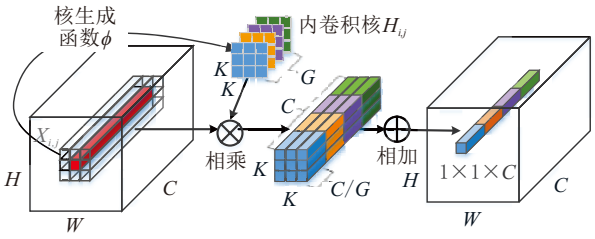


图2　内卷积原理

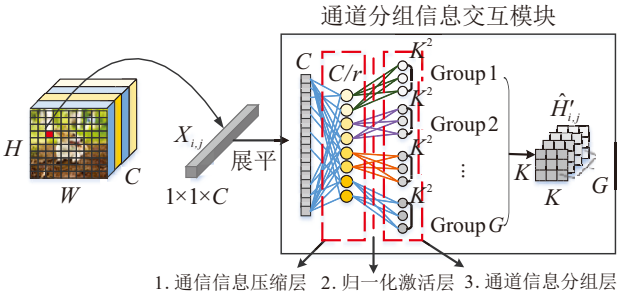


图3　通道信息交互模块

信息, 如图 4 所示. 特征图上以 (i, j) 为中心的 $K\times K$ 邻域, 经过通道信息交互模块生成各自包含通道信息的内卷积核 $\hat{H}'_{i+v,j+u}$, $u,v\in[-\lfloor K/2\rfloor, \lfloor K/2\rfloor]$. 选取 $\hat{H}'_{i+v,j+u}$ 各自相同层 $K\times K$ 个内卷积核 $\hat{H}'_{i+v,j+u,g}$, $g\in[1,G]$, 按顺序分成 G 组, 每组包含 K^2 个 $K\times K$ 核. 每个空间信息交互单元负责生成一个融合空间域信息卷积核 $\hat{H}_{i,j,g}$. 具体过程是: 第 g 组内卷积核 $\hat{H}'_{i+v,j+u,g}$, 从左上角开始, 每次选取相同位置元素构成一个 $1\times K$ 行向量, 与空间信息交互权重 $K\times 1$ 列向量做内积, 得到一个标量元素. 依次进行 K^2 次操作, 得到 K^2 个元素, 从左上角开始依次展开, 得到一个 $K\times K$ 的卷积核 $\hat{H}_{i,j,g}$. G 个空间信息交互单元生成内卷积核按序排放, 组成 G 组 $K\times K$ 大小的卷积核, 即经过空间信息交互模块得到的属于像素点 (i, j) 的卷积核 $\hat{H}_{i,j}$.

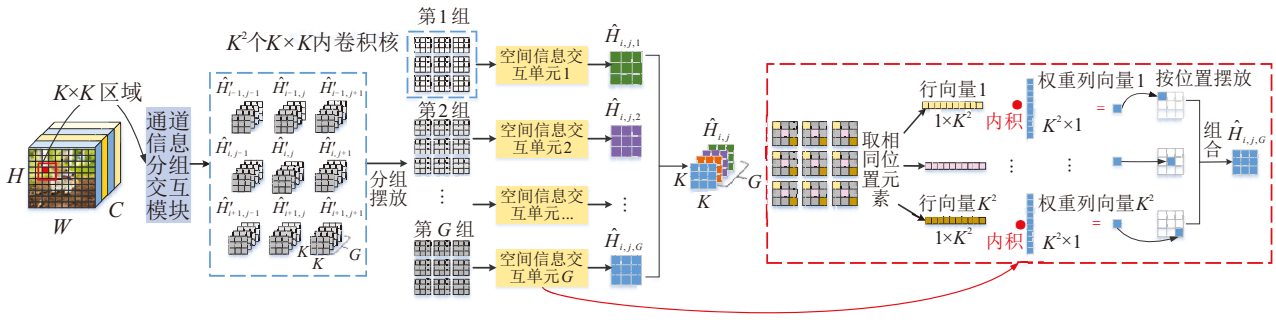
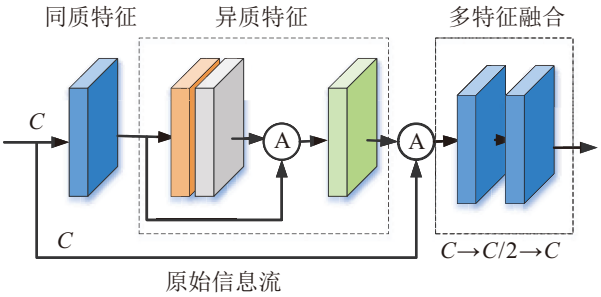


图4　空间信息交互模块

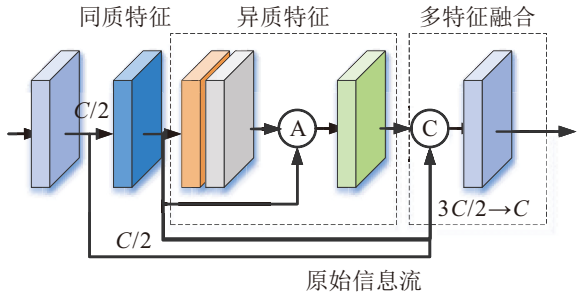
1.2.3　SCM 模块

SGWInvo 能够有效弥补共享卷积核在特征表达上的局限性, 并且在使用大尺寸卷积核时, 能够避免参数量的显著增加. 基于这些特性, 本文提出多特征提取融合模块 SCM (SGWInvo-Conv module), 如图 5 所示. 通过卷积操作提取局部同质特征, 利用 SGWInvo 提取局部异质特征, 并结合来自输入的原始信息流, 借助由卷积构成的融合层, 实现多特征提

取与融合. SCM1 模块采用串行连接方式逐步获取不同层次特征信息, 并最终将其融合. 为进一步减少网络深层参数量, SCM2 模块在 SCM1 基础上, 通过跨局部连接方式实现更轻量化设计. 图 1 中主干网络 SCNet 主要由 SCM 和 CBS 卷积等模块组成.



(a) SCM1



(b) SCM2

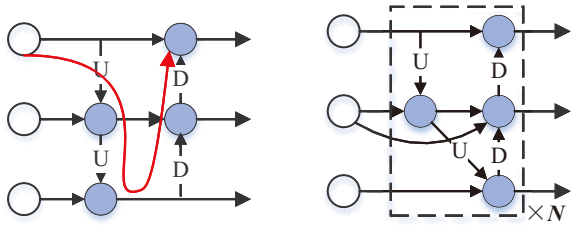
图5 SCM 模块

1.3 颈部网络

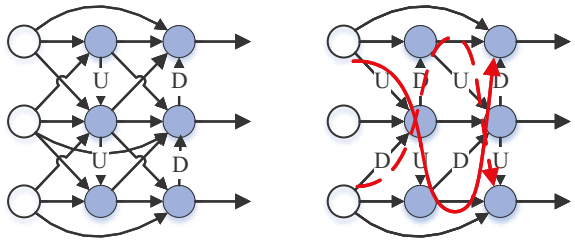
YOLOv8n 颈部网络 PAN[17] 如图 6(a) 所示, 其连接较为单薄, 导致特征信息融合不足, 容易丢失细节信息等问题, 目前的研究通常以增加结点和连接路径来改善该问题, 如图 6(b) 和 (c) 所示. 为研究更优的连接方式, 本文在 PAN 基础上设计一种双路径聚合网络 (dual path aggregation network, DPAN), 如图 6(d) 所示. DPAN 首先在 PAN 基础上增加一条先自下而上, 再自上而下的融合路径, 强化深层向浅层的信息互融; 其次, 增加来自主干网络的跳跃连接, 引入主干网络提取的原始特征信息, 从而避免在多层采样和特征融合过程中可能发生的信息丢失或过度抽象, 进而增强网络的鲁棒性. DPAN 通过复用共享路径和节点, 在不增加太多成本的前提下, 实现更多尺度的特征信息融合.

1.4 检 测 头

本文借鉴文献 [7] 方法, 采用深度可分离卷积替代 YOLOv8n 检测头中分类分支的标准卷积, 以减少模型参数量, 如图 1 中 Detect 部分所示. 针对 YOLOv8n 依赖后处理 NMS 影响推理时延问题, 文献 [7] 提出一对多和一对一联合训练策略, 其中一对多训练为



(a) YOLO-PAN (b) Bi-PAN



(c) GPAN (d) DPAN

图6 颈部网络及改进

一对一训练提供丰富的监督信息, 在推理阶段仅进行一对一匹配, 从而避免 NMS 带来的额外计算开销. 然而, 联合训练方法在实际应用中, 尽管推理时不进行一对多匹配, 但相关分支结构参数仍被部署模型保留, 对推理时延产生影响. 为解决这一问题, 本文提出 YOLO2YOLO 的分步训练方法 (以下简称 YOYO), 如图 7 所示. 该方法首先通过一对多匹配训练获得 SGWInvo-YOLO 模型, 然后使用该模型训练好的主干网络和颈部网络进行初始化, 并冻结这些部分的权重, 最后仅重新训练一对一匹配的检测头部分, 从而得到最终的 SGWInvo-YOYO 模型.

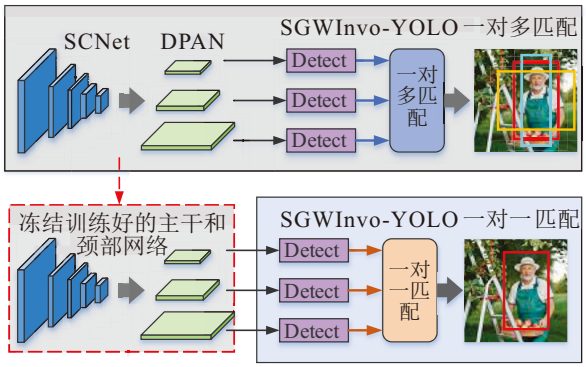


图7 分步训练策略

1.5 损失函数

YOLOv8 采用 CIoU 损失函数, 其惩罚项v反映预测框与真值框的宽高比, 优点是收敛快, 但当预测框与真实框的宽高呈现线性比例时, 惩罚项v为 0, 导致该项不起作用. 为解决该问题, 本文在 CIoU 基础上引入真实宽高的差值项v_{wh}, 改进后的损失函数简称 I-CIoU (improved CIoU), 其公式如下:

$$L_{\text{I-CIoU}} = 1 - \text{IoU} + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v + (1-\alpha)v_{wh}, \tag{1}$$

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

$$v_{wh} = \frac{2}{\pi^2}[(\arctan w^{gt} - \arctan w)^2 + (\arctan h^{gt} - \arctan h)^2, \tag{3}$$

$$\alpha = \frac{v}{(1-\text{IoU}) + v}. \tag{4}$$

其中: ρ为欧氏距离; b、w、h和b^{gt}、w^{gt}、h^{gt}分别为真实框和预测框的中心点、宽、高; c为包裹真实框和预测框的最小矩形. 当宽高比相差大时, 宽高比v起主要作用, 实现快速调节; 当宽高比例相差小时, 宽高差值v_{wh}起主要作用, 实现基于真实宽高差值的精准调节.

2　实验与结果分析

本文实验基于 PyTorch 框架实现, 操作系统为 Linux, CPU 为 AMD EPYC 9754, GPU 为 NVIDIA GeForce RTX 4090D; 实验采用公开数据集 PASCAL VOC 和 COCO 数据集. PASCAL VOC 数据集包含 20 个类别, 其中训练集包含图片 16 551 张, 验证集包含图片 5 011 张. COCO 数据集包含 80 个类别, 训练集包含 118 287 张图片, 验证集包含 5 000 张图片. 参数设置采用 YOLOv8n 默认参数, 批次大小设置: PASCAL VOC 数据集为 64, COCO 数据集为 128. 采用均值平均精度 (mean average precision, mAP)、参数量 (parameters), 计算复杂度 (floating point operations, FLOPs) 和时延 (latency) 等作为评价指标.

表1　颈部改进网络对比实验结果

模型	参数量/Mbit	mAP0.5/%	mAP0.5:0.95/%
PAN[17]	1.7	73.2	53.8
Bi-FPN$_{N=2}$[18]	2.8	74.6	54.7
GFPN[19]	2.8	74.1	54.2
DPAN	2.6	74.9	54.9

2.1　颈部网络改进对比实验

为验证不同颈部网络的连接效率, 并确保对比的公平性, 所有实验均采用未改进 YOLOv8n 模型, 仅更换颈部网络的连接方式. 结果如表 1 所示, 其中 $N = 2$ 表示 Bi-FPN 的级联数量为 2; 表中所示参数量仅包括颈部网络参数, 不包含网络其他部分. 实验结果表明, 相较于 PAN, 所有改进的颈部网络在检测精度上均有显著提升. 从综合表现看, 本文提出的 DPAN 在增加最少参数量的情况下, 达到了更高的精度提升, 表明 DPAN 有着更优的连接效率.

2.2　消融实验

为验证本文算法改进部分的有效性, 在 PASCAL VOC 数据集上进行消融实验, 结果如表 2 所示. 以 YOLOv8n 为基线模型, "√" 表示用该部分替换 YOLOv8n 原来部分, SCNet 为主干网络, DPAN 为颈部网络, Light-Head 为轻量化分类分支检测头, 一对一匹配 (One2One) 简称 O2O. O2O$^{\#}$为本文提出分步训练方法; 第 1 组为 YOLOv8n 基线模型结果; 第 6 组、第 7 组分别为所提出 SGWInvo-YOLO、SGWInvo-YOYO 模型结果. 实验结果表明, SCNet 和 Light-Head 都能在轻量化的同时有效提升检测精度, 验证了这两项改进的有效性. DPAN 解决了 PAN 网络在特征融合方面的不足, 显著提升了检测精度, 尽管不可避免地带来一定的参数增加, 然而通过系统性地整合各项改进, 其他轻量化措施能够弥补新增参数开销, 且整体检测精度实现 "1 + 1 > 2" 的提升效果. 采用 I-CIoU 损失函数后 mAP0.5 和 mAP0.5 : 0.95分别增加 0.2% 和 0.1%, 虽然提升幅度不大, 但更能表达出预测框与真值框的宽高差值, 且不影响模型的整体结构, 也不会增加额外的参数量. 本文两种改进算法在参数量方面均比 YOLOv8n 降低 23.3%, 其中, SGWInvo-YOLO 采用 YOLO 典型的一对多匹配策略显著提升了检测精度, mAP0.5 和 mAP0.5 : 0.95 较 YOLOv8n 提升 2.3% 和 2.9%; SGWInvo-YOYO 采用一对一匹配策略, 避免了

表2　消融实验结果

序号	SCNet	DPAN	Light-HeadA	I-CIoU	O2O$^{\#}$	mAP0.5/%	mAP0.5:0.95/%	参数量/Mbit	GFLOPs	Latency/ms
1						73.2	53.8	3.0	8.4	5.7
2	√					73.8	54.2	2.4	6.8	5.5
3		√				74.8	54.9	3.2	8.9	5.9
4			√			73.6	54.1	2.7	7.2	5.6
5				√		73.4	53.9	3.0	8.4	5.7
6	√	√	√	√		75.5	56.7	2.3	6.7	5.7
7	√	√	√	√	√	73.4	54.3	2.3	6.7	5.1

表3 检测方法改进对比实验结果

网络	O2O	参数量/Mbit			mAP0.5/%	mAP0.5 : 0.95/%	Latency/ms		
		训练	推理	部署			前处理	推理	NMS
YOLOv10n[7]	×	2.3	2.3	2.3	74.7	54.9	0.2	4.8	0.7
YOLOv10n[7]	√	2.7	2.3	2.7	73.2	54.0	0.2	5.2	0
YOLOv11n[8]	×	2.6	2.6	2.6	73.9	54.7	0.2	4.7	0.7
YOLOv11n[8]	√	3.0	2.6	3.0	72.2	53.7	0.2	5.1	0
SGWInvo-YOLO	×	2.3	2.3	2.3	75.5	56.4	0.2	4.8	0.7
SGWInvo-YOLO	√	2.7	2.3	2.7	73.8	54.7	0.2	5.2	0
SGWInvo-YOYO	√	2.3	2.3	2.3	73.4	54.3	0.2	4.9	0

NMS 处理所带来的额外计算开销, 在略微提高精度的同时, 推理时延降低 10.5%.

2.3 检测方法改进对比实验

一对一匹配与一对多匹配检测方法对比结果如表 3 所示. O2O 项代表一对一匹配. YOLOv10n、YOLOv11n 和 SGWInvo-YOLO 均采用文献 [7] 的联合训练方法, SGWInvo-YOYO 采用本文提出的分步训练方法. 实验结果表明, 采用联合训练方式的一对多检测算法, 尽管在推理过程中不进行一对多匹配, 其影响仍然体现在实际部署的模型中. 具体而言, 在参数量方面, 部署模型的参数量比推理模型大约增加了 0.4 Mbit; 在时延方面, 虽然总体时延减少, 主要是由于省掉了后处理 NMS 所耗时延, 但推理时延较采用一对多匹配的方法仍然增加了约 0.4 ms. 相比之下, 本文提出的 SGWInvo-YOYO 方法通过分步训练方式, 实现了部署与推理模型参数量的一致, 并在推理时延上表现出更低的延迟.

此外, 选择 RT-DETR[20] 中最小的模型 RT-DETR-R18 进行对比, 结果如表 4 所示. RT-DETR-R18 参数量约为 SGWInvo-YOLO 的 8 ～ 9倍, 为了在相同参数量级上进行对比, 实验中将 DETR 模型轻量化至与本文模型接近的参数量, 具体做法是: 将 DETR 缩至 3 层 Transformer, 同时通道数减半; DETRtiny 表示轻量化后的模型. SGWInvo-DETRtiny 比 SGWInvo-YOLO, mAP0.5 和 mAP0.5 : 0.95 精度分别为下降 7.7% 和 6.4%, 实验结果表明, 随着 Transformer 层数缩减, DETR 检测精度也显著下降. 第 4 组和第 5 组实验均采用 YOLO2DETR 分步训练模式, 即在 SGWInvo-YOLO 训练好的权重基础上再继续训练 DETR 的检测头. 在第 4 组实验中, 应用训练好的主干和颈部网络后, 检测精度明显下降, 这表明训练好的 YOLO 网络未必能很好地适应 DETR 结构. 综上所述, 直接将 DETR 应用于 SGWInvo-YOLO 进行检测, 在检测精度和模型规模方面均未表现出明显优势.

表4 与 DETR 对比实验结果

模型	预训练	参数量 (Mbit)	mAP0.5 (%)	mAP0.5:0.95 (%)
RT-DETR-R18[20]	×	20.1	74.9	57.0
SGWInvo-YOLO	×	2.3	75.5	56.4
SGWInvo-DETRtiny	×	2.6	67.8	50.0
SGWInvo-DETRtiny	主干+颈部	2.6	64.3	46.6
SGWInvo-DETRtiny	主干	2.6	67.0	49.1

2.4 与其他 YOLO 算法对比实验

为评估本文算法性能, 在 PASCAL VOC 数据集上对比其他同量级的 YOLO 算法, 实验结果如表 5 所示. “*” 标记为一对一匹配检测模型, 未标记均为一对多匹配检测模型; “/ ” 前后数值表示训练和推理数值. 除 SGWInvo-YOYO 外, 其他模型均从零开始训练 200 轮, SGWInvo-YOYO 采用分步训练策略, 在 SGWInvo-YOLO 训练 200 轮的基础上继续训练. 在一对多匹配检测模型中 SGWInvo-YOLO 取得最佳精度, 比 YOLOv11n 参数量降低约 11.5%, mAP0.5 和 mAP0.5:0.95 分别提升 1.6% 和 1.7%; 在一对一匹配检测模型中 SGWInvo-YOYO 取得最佳精度, 比 YOLOv10n, mAP0.5 和 mAP0.5:0.95 分别提升 0.2% 和 0.3%, 推理时延降低 5.6%. 为更系统地展示表 5 内容, 各算法综合性能对比如图 8 所示. 横轴表示推理时延, 值越小推理时延越低; 纵轴表示

表5 PASCAL VOC 数据集上对比实验结果

模型	mAP0.5 (%)	mAP0.5:0.95 (%)	参数量 (Mbit)	GFLOPs	Latency (ms)
YOLOv5n[3]	64.2	38.9	1.8	4.3	6.3
YOLOv6n[4]	72.7	52.6	4.2	11.6	5.9
YOLOv7-tiny[5]	71.4	45.2	6.2	13.9	6.8
YOLOv8n[6]	73.2	53.8	3.0	8.2	5.7
*YOLOv10n[7]	73.2	54.0	2.7/2.3	8.4/6.8	5.4
YOLOv11n[8]	73.9	54.7	2.6	6.3	5.6
*YOLOv11n-O2O[8]	72.2	53.7	3.0/2.6	8.3/6.3	5.3
*SGWInvo-YOYO	73.4	54.3	2.3	6.7	5.1
SGWInvo-YOLO	75.5	56.4	2.3	6.7	5.7

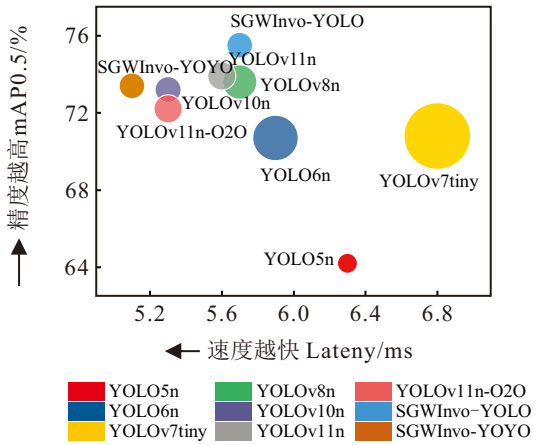


图8　各算法综合性能对比

mAP0.5 精度, 值越大检测精度越高; 圆半径表示模型参数量, 半径越大参数量越大.

为了更直观地评估检测效果, 将本文算法与 YOLOv8n、YOLOv10n 和 YOLOv11n 算法的可视化进行对比, 结果如图 9 所示. 1) 在背景目标区分度较高或场景较为简单的情况下, 如图 9(a) 和 (b) 所示, 各算法的检测效果差异较小. 2) 背景目标颜色纹理相近、区分度低场景如图 9(c) 所示, 栅格遮挡场景如图 9(h) 和图 9(i) 所示, 对比算法普遍存在漏检或误检的情况. 3) 烟雾遮挡如图 9(d) 所示, 对比算法定位精度较差, 存在定位不准的问题. 4) 光线不足或暗环境下如图 9(e) 所示, 多目标交叉重叠等复杂场景如图 9(f) 和图 9(g) 所示, 对比算法普遍存在漏检现象. 相比之下, 本文算法通过提取丰富的局部信息和增强的多尺度融合策略, 能够有效减少背景目标颜色和纹理相近、光线不足等场景中的漏检问题, 并且在交叉重叠、遮挡等复杂场景下表现优异.

为进一步验证本文算法的泛化能力, 在更具挑战性的 COCO 数据集上进行对比实验, 结果如表 6 所示. 表中参数量和检测精度指标, 除本文算法外, 其他算法结果均来自各文献中报告的最佳性能, AP_S、AP_M 和 AP_L 为 COCO 数据集的评价指标, 表示小、中和大类目标的平均精度. 针对可获取测试条件的对比模型, 统一在本文实验的硬件平台上, 基于其预训练权重复现了推理时延 (表中 Latency 项), "—" 表示对应模型未提供可复现条件.

实验结果表明, 在一对多匹配检测模型中, SGWInvo-YOLO 的 mAP0.5 和 mAP0.5 : 0.95 精度较文献 [23] 分别低 1.3% 和 1.0%, 经深入分析, 该差异主要源于小目标检测指标 AP_S 低 0.8%, 而中、大目标检测指标 AP_M 和 AP_L 高 0.6% 和 1.9%. 此外, SGWInvo-YOLO 模型参数量较文献 [23] 少约 20.7%. 相较于 YOLOv8n 基准模型, SGWInvo-YOLO 在检测性能上呈现更均衡的提升, 同时模型更轻量; 一对一匹配检测模型中 SGWInvo-YOYO 和 YOLOv10n 检测精度相当; 由于本文 SCM 模块提取了更丰富的局部特征信息以及 DPAN 更好的多尺度融合, 使得 SGWInvo-YOLO 和 SGWInvo-YOYO 算法在小目标指标 AP_S 上有较好的提升.

本文算法在 COCO 数据集上训练过程的

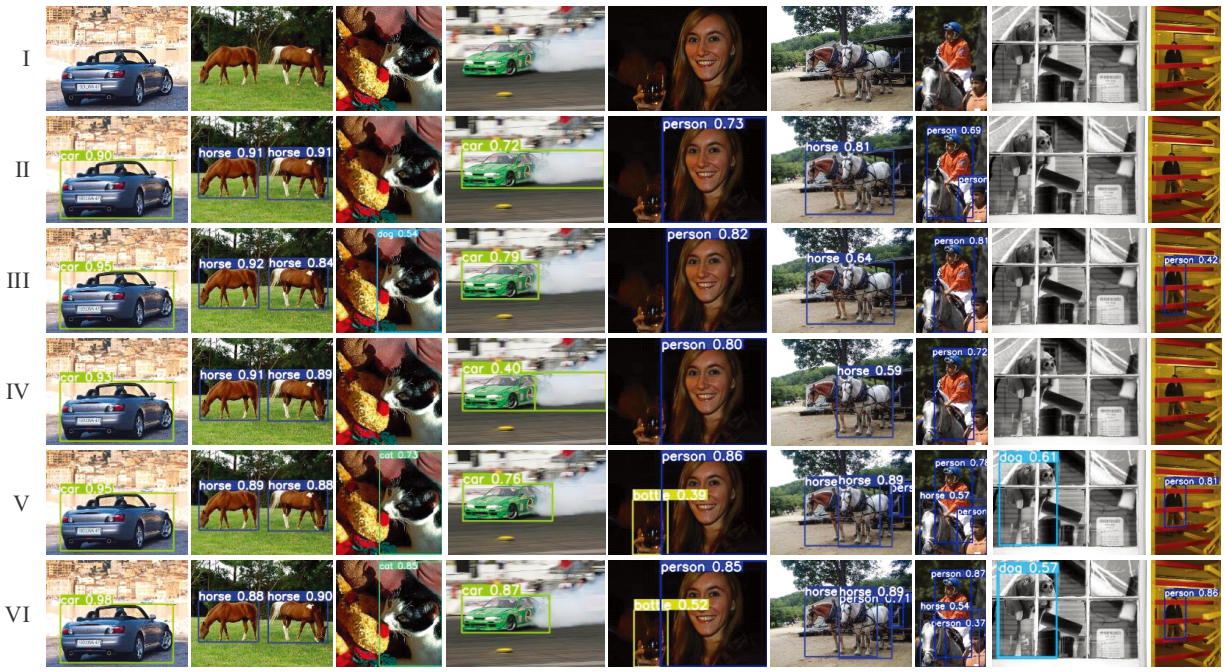


图9　检测结果可视化对比

表6 COCO 数据集上对比实验结果

模型	参数量/Mbit	mAP0.5/%	mAP0.5:0.95/%	AP_S/%	AP_M/%	AP_L/%	Latency/ms
YOLOv5n[3]	1.9	46.2	28.0	14.1	32.2	36.7	6.5
YOLOv6n[4]	4.3	50.7	35.2	15.5	39.5	52.5	6.2
YOLOv7-tiny[5]	6.2	47.3	30.8	10.0	31.9	52.2	7.1
YOLOv8n[6]	3.2	52.5	37.3	18.6	41.0	53.5	5.9
*DEYO-tiny[21]	4.2	52.8	37.6	17.9	41.3	54.2	6.8
*YOLOv10n[7]	2.7/2.3	53.8	38.5	18.9	42.4	54.6	5.6
YOLOv11n[8]	2.7	55.3	39.4	19.9	43.3	57.0	5.8
DCT-YOLOv8n[22]	3.0	53.9	38.4	18.3	41.5	52.2	—
MYOLOv8n[23]	2.9	56.8	40.6	22.5	42.7	52.3	—
*SGWInvo-YOYO	2.3	54.8	38.2	20.2	42.3	53.2	5.3
SGWInvo-YOLO	2.3	55.5	39.6	21.3	43.3	54.2	5.9

mAP 曲线如图 10 所示. 因为 SGWInvo-YOYO 采用分步训练, 在 SGWInvo-YOLO 训练完的基础上继续训练, 所以图 10 中 SGWInvo-YOYO 的初始 mAP0.5 数值高于 SGWInvo-YOLO 曲线. 可以看出, SGWInvo-YOYO 在 SGWInvo-YOLO 基础上继续训练 50 ～ 100轮左右便能达到最佳性能, 采用分步训练方法, 虽然增加了约 10% ～ 20% 的额外训练成本, 但消除了 NMS 带来的推理延时, 且在第 2 步训练时, 只需对一对一匹配的检测头训练, 训练成本的增加并不显著.

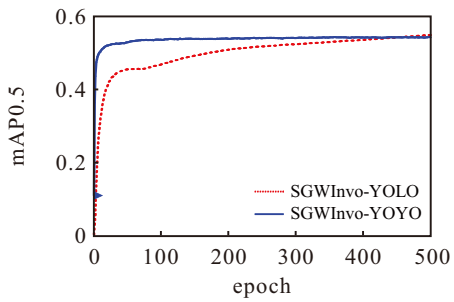


图10 COCO 数据集上训练过程的 mAP 曲线

3 结 论

针对现有轻量级目标检测算法在检测精度、特征融合能力不足、难以满足边缘设备部署需求的问题, 本文在 YOLOv8n 基础上进行改进. 首先基于 SGWInvo 设计了一种轻量化主干网络 SCNet 替换 YOLOv8n 主干网络, 轻量化的同时增强了局部特征的表达能力; 其次提出一种双向路径聚合网络 (DPAN), 以提高多尺度目标的特征融合能力; 最后实现了一对多匹配检测算法 SGWInvo-YOLO 和一对一匹配检测算法 SGWInvo-YOYO. 实验结果表明, 与 YOLOv8n 相比, 两种模型均在参数量方面减少了 23.3%; 在检测速度方面, SGWInvo-YOLO 与 YOLOv8n 相当, SGWInvo-YOYO 推理时延降低了 10.5%; 在检测精度方面, 基于 mAP0.5 指标的评估结果表明, SGWInvo-YOLO 和 SGWInvo-YOYO 在 PASCAL VOC 数据集上分别提升了 2.3% 和 0.2%, 在 COCO 数据集上分别提升了 3.0% 和 2.3%. 本文提出的两种轻量级模型相较于 YOLOv8n 在各项指标上均有一定程度的提升, SGWInvo-YOLO 更适用于对精度要求较高的应用场景, 而 SGWInvo-YOYO 更适合对速度要求较高的场景. 此外实验还表明, 无论是采用联合训练还是分步训练策略的一对一检测模型, 其检测精度均不及一对多检测模型的精度, 因此, 进一步提升一对一匹配检测的精度仍是本文未来研究的重要方向. 综上所述, 本文研究对于轻量级目标检测算法研究有一定的参考意义.

参考文献 (References)

[1] 王红梅, 王晓鸽, 王晓燕. 基于深度学习的复杂背景下目标检测[J]. 控制与决策, 2022, 37(12): 3115-3121. (Wang H M, Wang X G, Wang X Y. Target detection under complex background based on deep learning[J]. Control and Decision, 2022, 37(12): 3115-3121.)

[2] Ren S Q, He K M, Girshick R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[3] Jocher G. YOLOv5[EB/OL]. (2020-06-10)[2022-11-23]. http://github.com/ultralytics/YOLOv5.

[4] Li C, Li L, Jiang H, et al. YOLOv6: A single-stage object detection framework for industrial applications[J/OL]. 2022, arXiv: 2209.02976.

[5] Wang C Y, Bochkovskiy A, Liao H M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[C]. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Vancouver, 2023: 7464-7475.

[6] Jocher G. YOLOv8[EB/OL]. (2023-01-10)[2024-01-24]. https://github.com/ultralytics/ultralytics.

[7] Wang A, Chen H, Liu L H, et al. YOLOv10: Real-time end-to-end object detection[J/OL]. 2024, arXiv: 2405.14458.

[8] Jocher G. YOLOv11[EB/OL]. (2024-09-30)[2024-09-30]. https://github.com/ultralytics/ultralytics.

[9] Wang C Y, Mark Liao H Y, Wu Y H, et al. CSPNet: A new backbone that can enhance learning capability of CNN[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Seattle, 2020: 1571-1580.

[10] 何湘杰, 宋晓宁. YOLOv4-Tiny 的改进轻量级目标检测算法[J]. 计算机科学与探索, 2024, 18(1): 138-150.
(He X J, Song X N. Improved YOLOv4-tiny lightweight target detection algorithm[J]. Journal of Frontiers of Computer Science and Technology, 2024, 18(1): 138-150.)

[11] 周葳楠, 吴治海, 张正道, 等. 基于弱特征增强的轻量化小目标检测方法[J]. 控制与决策, 2024, 39(2): 381-390.
(Zhou W N, Wu Z H, Zhang Z D, et al. Lightweight small target detection method based on weak feature enhancement[J]. Control and Decision, 2024, 39(2): 381-390.)

[12] Ma N N, Zhang X Y, Zheng H T, et al. ShuffleNet V2: Practical guidelines for efficient CNN architecture design[M]. Computer Vision — ECCV 2018. Cham: Springer International Publishing, 2018: 122-138.

[13] 金立生, 张舜然, 郭柏苍, 等. 基于改进 YOLOv4 的轻量化路侧视角多目标检测算法[J]. 控制与决策, 2024, 39(9): 2885-2893.
(Jin L S, Zhang S R, Guo B C, et al. Lightweight roadside view multi object detection algorithm based on improved YOLOv4[J]. Control and Decision, 2024, 39(9): 2885-2893.)

[14] 张小艳, 王苗. 改进的 YOLOv8n 轻量化景区行人检测方法研究[J]. 计算机工程与应用, 2025, 61(2): 84-96.
(Zang X Y, Wang M. Research on improved YOLOv8n light-weight pedestrian detection method in scenic spots[J]. Computer Engineering and Applications, 2025, 61(2): 84-96.)

[15] Zhong J Q, Qian H M, Wang H L, et al. Improved real-time object detection method based on YOLOv8: A refined approach[J]. Journal of Real-Time Image Processing, 2025, 22(1): 1-13.

[16] Li D, Hu J, Wang C H, et al. Involution: Inverting the inherence of convolution for visual recognition[C]. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville, 2021: 12316-12325.

[17] Liu S, Qi L, Qin H F, et al. Path aggregation network for instance segmentation[C]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, 2018: 8759-8768.

[18] Tan M X, Pang R M, Le Q V. EfficientDet: Scalable and efficient object detection[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 10778-10787.

[19] Zhao G M, Ge W F, Yu Y Z. GraphFPN: Graph feature pyramid network for object detection[C]. 2021 IEEE/CVF International Conference on Computer Vision. Montreal, 2021: 2743-2752.

[20] Zhao Y A, Lv W Y, Xu S L, et al. DETRs beat YOLOs on real-time object detection[C]. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2024: 16965-16974.

[21] Ouyang H D. DEYO: DETR with YOLO for step-by-step object detection[J/OL]. 2022, arXiv: 2211.06588.

[22] 王涛, 张笃振. DCT-YOLOv5: 从频率角度设计目标检测算法[J]. 计算机技术与发展, 2024, 34(10): 69-76.
(Wang T, Zhang D Z. DCT-YOLOv5: Designing object detection algorithms from a frequency perspective[J]. Computer Technology and Development, 2024, 34(10): 69-76.)

[23] 张正勃, 曹爱岷, 王兴盛. 基于 MYOLOv8 的目标检测方法[J]. 计算机测量与控制, 2025, 33(1): 93-98.
(Zhang Z B, Cao A M, Wang X S. Object detection method based on MYOLOv8[J]. Computer Measurement and Control, 2025, 33(1): 93-98.)

作者简介

卢迪 (1971−), 女, 教授, 硕士生导师, 主要研究方向为数据融合与图像处理, E-mail: ludizeng@hrbust.edu.cn;

赵庆 (1987−), 男, 硕士生, 主要研究方向为深度学习与图像处理, E-mail: zhaoqing3156@163.com.