计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
2025,34(9):151−161 [doi: 10.15888/j.cnki.csa.009962] [CSTR: 32024.14.csa.009962]

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

改进 YOLOv8 的轻量化无人机航拍目标检测①

严嘉旭[1,2], 苏天康[1,2], 宋慧慧[1,2]

[1](南京信息工程大学 江苏省大数据分析技术重点实验室, 南京 210044)
[2](江苏省大气环境与装备技术协同创新中心, 南京 210044)
通信作者: 宋慧慧, E-mail: songhuihui@nuist.edu.cn

摘　要: 无人机航拍场景下的目标检测因目标尺寸小、物体间遮挡严重、尺度变化大等因素, 常出现漏检与误检的问题. 此外, 受限于无人机平台的计算性能, 实现高精度与轻量化的实时目标检测具有较大挑战. 为此, 本文提出了一种改进 YOLOv8 的轻量级目标检测算法. 该算法采用轻量化的分割一切模型 (segment anything model, SAM): MobileSAM 的图像编码器作为 YOLOv8 的骨干网络, 能够有效地提取多尺度特征, 提升模型对小目标的检测效率, 同时提高泛化能力, 增强在不同任务和数据集上的表现. 针对检测头部分, 进行了轻量化设计, 提出基于共享卷积与自适应特征缩放的轻量化检测头 LSCD (lightweight scalable shared convolutional detection head) 以减少参数量和计算开销, 保持检测精度的同时降低模型参数量. 最后, 使用归一化高斯 Wasserstein 距离 (normalized Gaussian Wasserstein distance, NWD) 损失, 提升小目标检测能力. 所提算法在 VisDrone-DET2019 数据集上对小目标的检测精度和召回率相较于原始 YOLOv8s 模型有较大提升; 相比于原始 YOLOv8s 模型, mAP50 提高了 3.2%, 达到 41.4%, 且参数量减少了 33.9%. 在 DOTA v1.0 数据集上, mAP50 达到 48.8%, 提升了 8%, 表明算法具有较好的泛化能力.

关键词: 无人机航拍场景; 目标检测; YOLOv8; 轻量化; 归一化高斯 Wasserstein 距离; MobileSAM; LSCD

引用格式: 严嘉旭,苏天康,宋慧慧.改进 YOLOv8 的轻量化无人机航拍目标检测.计算机系统应用,2025,34(9):151–161. http://www.c-s-a.org.cn/1003-3254/9962.html

Lightweight UAV Aerial Target Detection Based on Improved YOLOv8

YAN Jia-Xu[1,2], SU Tian-Kang[1,2], SONG Hui-Hui[1,2]

[1](Jiangsu Key Laboratory of Big Data Analysis Technology, Nanjing University of Information Science & Technology, Nanjing 210044, China)
[2](Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology, Nanjing 210044, China)

Abstract: Target detection from a UAV perspective often faces challenges such as small object size, severe occlusion, and significant scale variation, which result in missed and false detections. Moreover, constrained by the limited computational capacity of UAV platforms, achieving real-time detection with both high accuracy and lightweight deployment remains difficult. To address these challenges, a lightweight version of YOLOv8 is proposed. The image encoder of MobileSAM, a lightweight adaptation of the segment anything model (SAM), is integrated as the backbone network, enabling effective extraction of multi-scale features, improved detection efficiency for small targets, and enhanced generalization across diverse tasks and datasets. A lightweight detection head, termed lightweight scalable shared convolutional detection head (LSCD), is proposed. Based on shared convolution and adaptive feature scaling, LSCD reduces the parameter count and computational overhead while maintaining detection accuracy. In addition, the normalized Gaussian Wasserstein distance (NWD) loss is employed to further improve detection performance on small objects. Experimental results on the VisDrone-DET2019 dataset demonstrate that the proposed algorithm significantly enhances both precision and recall for small targets compared to the original YOLOv8s model. The mAP50 is increased

① 基金项目: 国家自然科学基金 (61872189)
收稿时间: 2025-02-18; 修改时间: 2025-03-10, 2025-03-18, 2025-03-28; 采用时间: 2025-04-07; csa 在线出版时间: 2025-07-25
CNKI 网络首发时间: 2025-07-28

by 3.2%, reaching 41.4%, while the number of parameters is reduced by 33.9%. On the DOTA v1.0 dataset, the proposed model achieves a mAP50 of 48.8%, representing an 8% improvement, indicating strong generalization capability.

Key words: UAV aerial perspective; target detection; YOLOv8; lightweight; normalized Gaussian Wasserstein distance (NWD); MobileSAM; lightweight scalable shared convolutional detection head (LSCD)

随着通信技术、高清图采、智能飞控等技术的发展, 无人机 (unmanned aerial vehicle, UAV) 技术得到了飞速发展, 推动了其在多个领域的应用, 如灾害评估、交通监控、农业测绘、电力测绘、军事侦察等领域[1–3]. 与此同时, 目标检测技术作为无人机智能化的核心支撑, 越来越受到研究者的关注. 目标检测的任务是识别图像中目标的类别并确定其位置.

早期的目标检测技术主要基于传统图像处理方法, 例如背景减除、光流法和帧间差分等[4]. 在正常环境下, 它们能展现出良好的性能表现. 然而, 在复杂环境背景下, 由于光照变化及目标遮挡等干扰的影响, 其检测精度将显著降低.

随着深度学习技术兴起, 基于卷积神经网络 (CNN) 的目标检测算法取得了重大进展. 主流目标检测算法分为两大类: 双阶段 (two-stage) 和单阶段 (one-stage) 方法. 双阶段方法以 R-CNN[5] 系列为代表, 包括改进后的 Fast R-CNN[6]和 Faster R-CNN[7]. 这类算法通过生成候选框并对其进行分类和回归, 实现高精度检测. 然而, 由于双阶段方法的计算复杂度较高, 检测速度相对较慢, 因此不适合实时性要求较高的无人机任务. 相比之下, 单阶段检测方法以 YOLO (you only look once)[8] 系列和 SSD (single shot multibox detector)[9]为代表. 该类方法仅需一次前向传播 (forward propagation, FP) 即可同步完成目标的定位与分类, 具有检测速度快、实时性强的突出优点. 其中, 特别是 YOLO 系列算法, 在精度与性能之间实现了较好的均衡, 并通过不断改进模型结构, 在实时性和检测性能方面持续提升.

然而, 在无人机航拍场景中, 由于镜头距离物体较远, 视角中存在众多小目标; 它们一般存在尺寸小、细节特征模糊以及噪声干扰等问题, 使得以上方法难以完全适用. 因此, 小目标检测问题成为无人机目标检测的难点之一.

针对小目标检测问题, 韩俊等人[10]开发了多尺度特征提取模块 LM-fem 和混合域注意力模块 S-ECA, 这些技术显著增强了 YOLOv5 在特征提取方面的效能, 提升了小目标检测能力. 史涛等人[11]在 SPPF (spatial pyramid pooling fast) 模块中创新性地融入了大核可分离注意力机制. 该机制通过相对较低的计算成本, 显著提升了对小型目标的特征表达能力. 然而, 这一进步并非没有代价, 它在一定程度上导致了目标检测的漏检与误检现象. 尽管上述手段能在某种程度上提升模型对微小目标的识别能力, 但同时也会不可避免地增加噪声, 使得识别性能的提高并不显著. Wang 等人[12]在 YOLOv8 的网络中嵌入了小目标检测结构 (small target detection structure, STC), 虽然提高了小目标检测能力, 但同时也导致了参数量的大幅增加, 导致计算性能下降. 吴明杰等人[13]使用带自注意力 (self-attention) 机制的检测头替换原始检测头, 提高了检测层的特征表示能力, 解决了特征丢失的问题, 提高了小目标的检测能力, 但同时也不可避免地造成了参数量的稍大提升. 尽管以上方法针对小目标问题取得不错效果, 但都不可避免地增大了计算量. 无人机计算资源有限, 需要同时满足检测速度和小目标问题.

针对计算资源有限的问题, 众多学者专注于开发轻量化的网络模型, 以便在不牺牲过度精度的情况下提高检测算法的性能, 学者们提出了如 MobileNet[14–16], ShuffleNet[17]和 GhostNet[18]等轻量级网络结构, 其采用了深度卷积以及组卷积技术来有效提取空间特征, 虽然降低了计算量, 但也会增加访存次数从而降低碎片化计算的效率, 这一点在设计轻量化网络的时候需要权衡. Wang 等人[19]通过将 YOLOv5 的骨干网络替换成 MobileNetV3, 并结合专门针对小目标的检测层, 有效的轻量化了模型并提高了在移动端上的运行速度. Liu 等人[20]提出了名为 EdgeYOLO 的框架, 该框架设计了一种更为轻量且高效的解耦式头部结构, 使其能够被部署于边缘计算平台, 从而实现在无人机平台上更为迅速和精准的检测. Lee 等人[21]提出基于幅度的层自适应剪枝 (layer-adaptive magnitude-based pruning, LAMP) 算法. 该算法通过引入自适应幅值剪枝分数, 有效精简了权重并显著降低了模型参数量和计算量[22].

总的来说, 为了提高小目标检测精度, 通常需要采用多尺度特征提取和注意力机制等方法, 这虽然能增强检测能力, 但也增加了计算复杂度和模型参数, 影响运行速度. 然而, 模型轻量化通常通过减少参数和计算量提升速度, 但这可能削弱模型对小目标细节的捕捉能力, 影响检测精度. 因此, 无人机视角下目标检测仍有很大进步空间, 为了解决以上问题, 本文分析得出以下改进方案.

(1) 考虑引入 MobileSAM 的图像编码器到骨干网络中, MobileSAM 的图像编码器经过专门的设计和优化, 能够处理多种复杂场景并能提取高质量的多尺度特征, 从而帮助更有效的检测小目标; 并且, 编码器经过海量数据的大规模训练, 具有很好的泛化能力, 引入后模型更能适应多样化的数据分布, 提高在不同任务和数据集上的表现.

(2) 设计了新的损失函数, 引入 NWD (normalized Gaussian Wasserstein distance) 损失, 提升小目标检测能力.

(3) 针对检测头部分, 设计了一种基于共享卷积与自适应特征缩放的轻量化检测头 LSCD (lightweight shared convolutional detection head) 以减少参数量和计算开销.

1 YOLOv8 算法简介

YOLOv8 是 YOLO 系列中较新的目标检测模型, 由 Ultralytics 团队于 2023 年 1 月推出, 旨在进一步提升检测性能并适应多样化的应用需求. 模型延续了 YOLO 架构的经典设计, 包括输入 (Input)、主干网络 (Backbone)、颈部网络 (Neck) 和检测头 (Head) 这 4 个模块, 并基于不同网络深度与宽度提供了 5 种规模 (n/s/m/l/x) 的模型, 以平衡检测精度和计算资源的需求.

图 1 展示了官方 YOLOv8 网络的结构, 在输入阶段, 通过 Mosaic 数据增强提高模型的鲁棒性, 并且支持尺寸自适应缩放, 为模型适配不同输入分辨率提供了便利. 主干网络基于 DarkNet53 结构以实现有效的特征提取, 并将上个版本中的 C3 模块替代成了 C2f, C2f 中包含了多个残差瓶颈结构, 此举有助于信息的深度传递和表示. SPPF 模块通过采用空间金字塔池化 (spatial pyramid pooling) 技术, 能够在不同的尺度上提取特征. 这种多尺度池化方式使得网络能够更好地捕捉到不同尺寸目标的信息, 从而提高检测性能, 尤其是在处理不同大小的目标时.

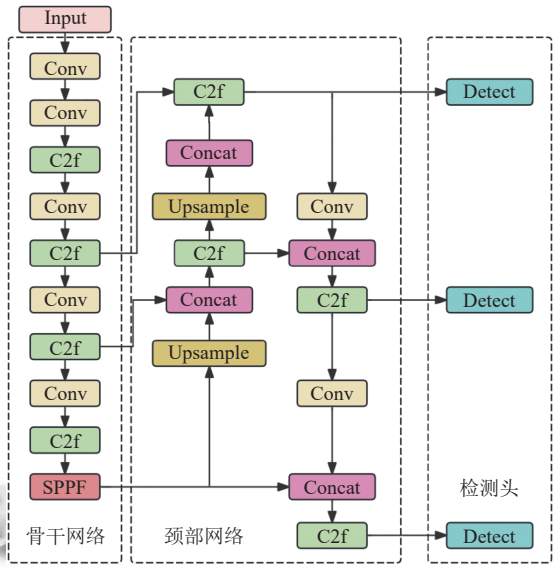


图 1 YOLOv8 网络结构

颈部 (Neck) 网络负责对特征图进行上采样, 其结合 FPN (feature pyramid network)[23]和 PAN (path aggregation network)[24]来融合从骨干网络中获得的特征, 丰富多种尺度的语义信息和定位信息, 此举有助于提升了对不同大小目标的检测能力.

检测头采用了解耦头 (decoupled-head) 结构, 将分类和回归任务分离, 提高了检测精度; 同时检测头包含 3 个尺寸各异的特征图, 用以检测不同大小的目标.

2 MobileSAM-YOLO 模型

针对算力资源有限情况下的实时无人机图像小目标检测存在精度低, 模型规模大, 部署难等问题, 本文在 YOLOv8s 的基础上, 提出了 MobileSAM-YOLO 模型, 替换轻量化的 MobileSAM 的图像编码器作为新的骨干网络, 设计全新的检测头, 引入新的归一化高斯 Wasserstein 损失 (normalized Gaussian Wasserstein loss), 构建轻量化小目标检测网络. 该模型整体结构如图 2 所示.

2.1 MobileSAM 图像编码器

MobileSAM[25]的图像编码器是一种基于视觉 Transformer (ViT) 的自监督学习的轻量化图像编码器, 其经过大规模数据训练, 具备强大特征表示能力, 可以有效地识别并提取小目标的特征, 提高检测精度, 且能适应不同任务和数据集合, 具备优秀的泛化性. MobileSAM 的思想源自分割一切模型 (segment anything model, SAM)[26], 其大致原理如图 3 所示.

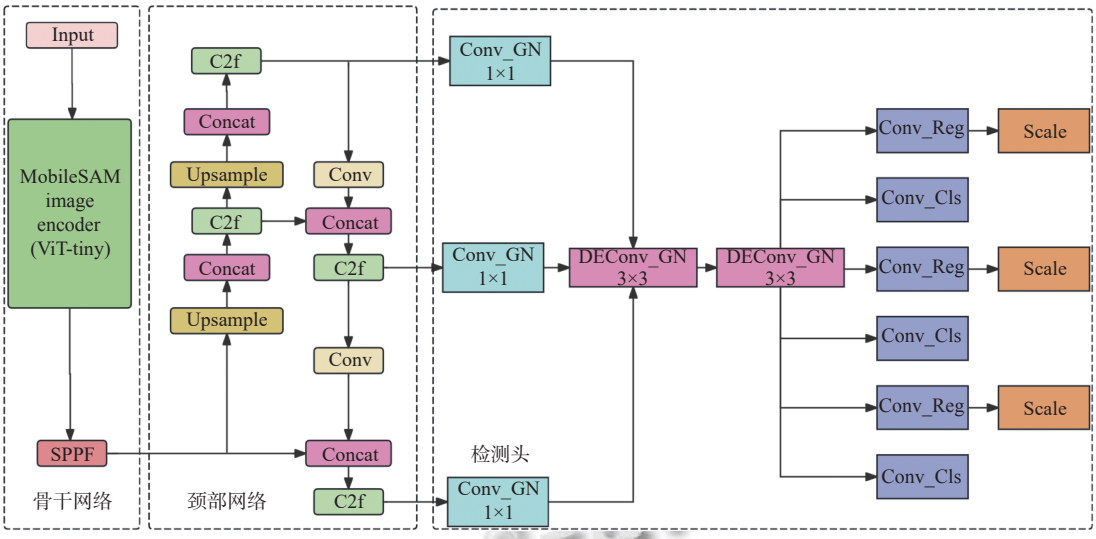


图 2 MobileSAM-YOLO 网络结构图

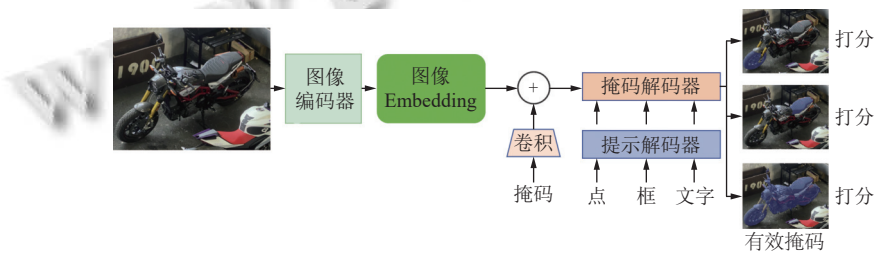


图 3 SAM 模型原理

SAM 通过巨型图像编码器 (ViT-H) 在 SA-1B 数据集 (包含 1 100 万张图像和 10 亿掩码) 大规模预训练 (pre-trained) 提取全局特征, 随后利用轻量级解码器根据提示生成分割掩码.

不难看出, ViT-H 在全局特征提取、小目标细节捕捉以及减少背景干扰等细节上表现出色, 若能够将该模块整合到 YOLOv8 的骨干网络中, 将会有效地提高无人机视角下小目标检测的性能. 但是, 由于 ViT-H 的参数量达到了 632M, 难以直接部署到无人机此类资源受限的移动端设备上; 因此, 本文提出利用轻量化的 SAM——MobileSAM, 其编码器将原始的 ViT-H 替换为更小的 ViT-tiny, 其将原来的 16 层 Transformer 缩减到 6 层, 每层的注意力头从 16 个降低到 4 个, 隐藏层维度从 1 280 降低到 256, 参数量从 640M 降至 5.78M; 该编码器还使用分组卷积 (grouped convolution) 代替标准卷积, 并引入了动态稀疏注意力机制, 减少了 FLOPs 数. 为了将 SAM 的知识迁移到新的 ViT-Tiny 编码器中, MobileSAM 通过解耦知识蒸馏 (decoupled knowledge distillation, KD)[27]保留 SAM 的特征表达能力, 如图 4 所示.

其大概思想与传统的知识蒸馏不同, 传统方法通常将教师 (Teacher) 模型和学生 (Student) 模型端到端联合训练, 但 MobileSAM 采用分阶段解耦策略: 首先针对图像编码器进行知识蒸馏, 冻结教师模型 (SAM) 的解码器, 仅训练学生模型 (MobileSAM) 的编码器, 然后固定学生编码器的权重, 单独微调轻量化的解码器以适应下游的分割任务. 采用这种方式能够避免同时优化编码器与解码器的参数冲突, 降低训练复杂度; 并且能保证编码器的蒸馏过程不受下游任务 (如分割解码器) 的干扰, 确保特征提取能力的纯粹性, 从而保留编码器的泛化性.

MobileSAM 极大地保留了 SAM 学习到的多尺度特征和小目标细节信息, 能够有效地提高小目标的检测能力; 同时保持模型的轻量化, 有利于保持无人机平台上的检测性能.

2.2 YOLOv8 网络在头部的改进

本文提出了一种基于共享卷积与自适应特征缩放的轻量化检测头 LSCD (lightweight shared scalable

convolutional detection head), 旨在实现模型参数量与计算成本显著降低的同时维持检测精度. 该设计通过改进常规检测头有效维持了轻量化与精度之间的平衡.

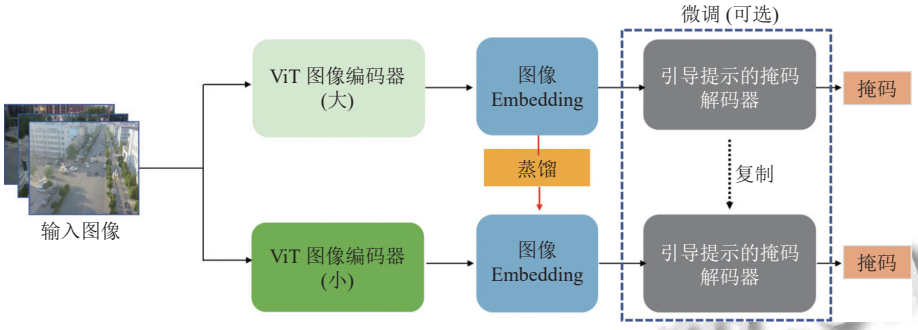


图 4 MobileSAM 的解耦知识蒸馏

常规检测头中各检测层由相互独立的卷积运算构成, 如图 5 所示, 每个分支都需要两个 3×3 卷积和用于分类和目标框回归的卷积, 这虽然能够提高检测的精确率, 但同时也增加了参数量.

与常规检测头不同, 为了降低对各个分支分别进行卷积的冗余计算开销. LSCD 采用共享卷积的策略, 如图 6 所示. 首先, 将输出特征图 P3、P4 和 P5 分别进行 1×1 的 Conv_GN 的操作. 具体如图 6(a) 所示: 首先进行 1×1 的卷积操作, 接着进行组归一化 (group normalization, GN)[28], 最后通过激活函数 Mish 进行输出. 这样做是为了将特征图的通道数压缩以便在进一步处理前混合特征, 同时保留特征图的空间维度. 这里使用 Mish[29]作为激活函数, Mish 对精度和泛化性有良好的优化能力, 能够允许更优质的信息流入神经网络.

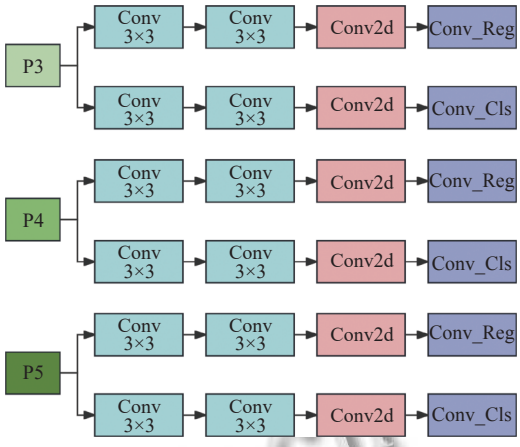


图 5 常规检测头

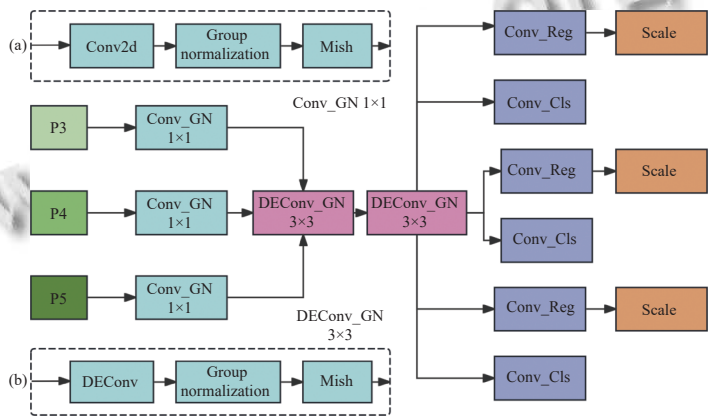


图 6 LSCD 共享卷积检测头

接着, 上一步输出的混合特征经过 2 个 3×3 的 DEConv_GN 共享卷积操作, 如图 6(b) 所示: 首先进行 3×3 的 DEConv 卷积操作, 接着进行组归一化 GN, 最后同样通过激活函数 Mish 进行输出. 其中, DEConv (detail enhanced convolution)[30]是一种能够增强细节感知的卷积操作, 相较于普通卷积主要是提取低频信息, DEConv 通过引入中心差分卷积、角度差分卷积、垂直差分卷积和水平差分卷积这 4 种差分卷积来帮助增

强高频信息 (如边缘和轮廓) 的表示. 具体来说, DEConv 将 4 个差分卷积和 1 个普通卷积并行部署用于特征提取, 然后将提取到的特征简单相加得到 DEConv 的输出 此举有助于 DEConv 可以同时考虑低频和高频信息, 以更好地捕捉图像的整体特征和详细特征, 从而提高对细节特征的感知. DEConv 的结构图如图 7 所示. 值得注意的是, 并行部署 5 个卷积层进行特征提取会导致参数量的增加进而降低检测速度, 在此情况下, 可以利用卷积的有用特性: 如果尺寸、步长和填充都相同的多个二维卷积核对相同的输入进行卷积操作得到的输出进行相加操作, 那么可以在对应位置上将这些卷积核相加, 得到一个等效的卷积核, 该卷积核将产生相同的最终输出. 因此 DEConv 利用重新参数化技术, 在训练阶段通过多个并行卷积层学习丰富的特征表示, 并利用卷积的可加性在反向传播中分别更新这些并行卷积核的权重, 然后将这些核权重相加以获得前向传播中的等效核权重, 这些并行卷积层等效转换为一个标准的 3×3 卷积层, 从而在不牺牲性能的情况下提高了模型的检测精度.

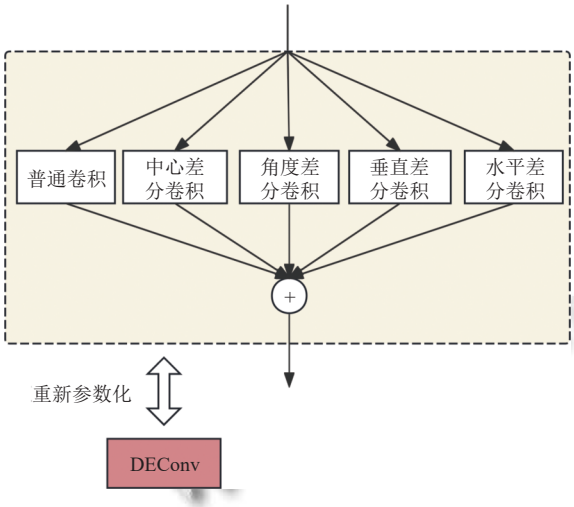


图 7 DEConv 结构图

随后, 经过两个 DEConv_GN 共享卷积融合得到的输出特征分别输入到两个分支中: Conv_Reg 分支负责预测框回归, Conv_Cls 负责目标分类. 最后, Scale 层对 Conv_Reg 层的输出进行缩放, 该层定义了一个可学习的缩放因子用来逐元素调整输出特征图的特征大小, 进而能够动态地调整后续的卷积层, 这项改进稳定了预测框的回归, 从而在训练过程中提高了预测框准确性.

为了有效应对模型训练过程中梯度消失的现象, 网络架构设计中一般会考虑采用批量归一化 (batch normalization, BN) 技术[31], 其通过将输入数据转换至近似标准正态分布, 提升了激活函数的敏感度, 从而使得梯度信息能够有效地在网络中传播. 然而, BN 有效性显著受到批量大小的影响. 特别是在小批量的情况下, 批量归一化能会导致输出结果的误差增加, 从而影响模型的性能. 为了克服批量大小对归一化效果的影响, LSCD 采用了组归一化 (GN) 替换 YOLOv8 检测头原有的 BN. GN 与 BN 不同, GN 将输入通道分组并在每个组内进行归一化, 从而减轻了批量大小变化对模型性能的影响, 特别是对于高精度图像, 即使在小批量情况下也能确保稳定性. 这一改进不仅有助于提高模型的收敛速度和预测准确性, 还增强了其鲁棒性和泛化能力. GN 在每个通道组内计算均值和方差, 有效减少了噪声. GN 的归一化过程如式 (1) 所示.

$$GN(x)=\frac{x-\mu}{\sqrt{\theta^2+\varepsilon}} \tag{1}$$

其中, x是输入特征, μ是每个组内计算出的均值, θ^2是每个组内计算出的方差, ε是一个很小的常数用来防止分母为零. 具体来说, 假设有一个形状为$[N,C,H,W]$的特征x, 其中N是样本数也就是批量大小, C是通道数, H和W是空间维度, 首先, 将通道数分成G组, 每组有C/G个通道, 然后对每组内的特征进行归一化.

文献[32]的实验表明, 在分类和回归任务中避免使用 GN 会导致模型精度下降 1% 左右, 因此 GN 被证实可以增强检测头在定位和分类方面的性能. 所以, 为了弥补参数量下降可能导致的检测头性能下降, GN 模块被应用到 LSCD 的所有卷积层中, 包括 DEConv_GN 和 Conv_GN. 此改进不仅加速了模型训练的收敛速度, 还提高了检测精度.

LSCD 通过将特征融合部分的参数共享, 有效减轻了模型的计算量, 提取的多尺度特征通过细节增强卷积的整合, 为后续任务提供了丰富的特征信息. 参照第 3.4.2 节的实验数据, LSCD 使模型在数据集 VisDrone-DET2019 上 mAP50 精度提升了 0.5%, 计算量下降了 0.7 FLOPs, 参数量下降了约 81%. VisDrone-DET2019 数据集包含超过 260 000 个目标, mAP50 提升 0.5% 意味着额外检测出约 1 300 个漏检目标, 并且这种提升是在参数量下降 81% 的情况下获得的, 这意味着模型每个参数的有效性提升了约 5.3 倍. 相比于其他轻量化方

法[14–18]在减少参数量的同时精度会不可避免的略微下降, LSCD 在降低 81% 参数量的同时仍提升了 0.5% 的精度, 体现出该模块的有效性.

2.3 归一化高斯 Wasserstein 损失函数

YOLOv8 中默认的预测框回归损失函数为 CIoU (complete intersection over union)[33], CIoU 是一种改进的 IoU 损失, 它考虑了重叠区域、中心点距离以及长宽比的差异. CIoU 的公式包括 3 个部分: IoU 损失、中心点距离的惩罚项和长宽比的惩罚项. CIoU 在处理边界框回归时表现不错, 尤其是在目标尺寸较大且边界框重叠较多的情况下. CIoU 具体公式如下:

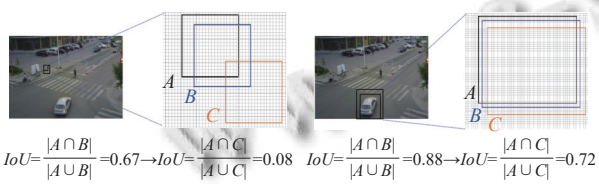
$$L_{\mathrm{CIoU}} = 1 - IoU + \frac{\rho^2(B, B^{\mathrm{gt}})}{C^2} + \alpha v \tag{2}$$

$$v = \frac{4}{\pi^2}\left(\arctan\left(\frac{w^{\mathrm{gt}}}{h^{\mathrm{gt}}}\right) - \arctan\left(\frac{w^{\mathrm{pred}}}{h^{\mathrm{pred}}}\right)\right)^2 \tag{3}$$

$$IoU = \frac{|B \cap B^{\mathrm{gt}}|}{|B \cup B^{\mathrm{gt}}|} \tag{4}$$

$$\alpha = \frac{v}{1 - IoU + v} \tag{5}$$

其中, IoU 是预测框与真实框的交并比, $\rho(B, B^{\mathrm{gt}})$是预测框中心点B与真实框中心点B^{gt}之间的欧氏距离, α是平衡系数, v是长宽比一致性惩罚项, 常数C是预测框与真实框的最小外接矩形的对角线的长度. w和h分别是预测框的宽度和高度, w^{gt}和h^{gt}分别代表真实框的宽度和高度. 然而, 当处理小目标或者边界框之间几乎没有重叠时, IoU 及其变体系列 (如 CIoU、GIoU[34]) 可能会出现问题, 如图 8 所示.



(a) 小尺度目标 (b) 大尺度目标

图 8 大小目标先验框 IoU 对比

面对大目标时, 预测框 B 和 C 与真实框 A 的 IoU 变化不大 (从 0.88 降至 0.72), 但对于小目标来说, IoU 会因为几个像素点的偏离而产生很大的变化 (从 0.67 降至 0.08). 这样会导致梯度在这些情况下可能会消失, 导致优化困难.

因此, 本文考虑将现有的 CIoU 损失调整为归一化高斯 Wasserstein 距离 (normalized Gaussian Wasserstein distance, NWD)[35], 与 CIoU 的思想不同的是, NWD 利用边界框的高斯分布, 计算这些分布间的 Wasserstein 距离, 增强对微小目标的感知. NWD 的计算分为两步.

第 1 步计算边界框的高斯分布, w_A和h_A代表边界框 A 的宽度和高度, (c_{xA}, c_{yA})为中心坐标. 可得边界框的高斯分布$N(\mu, \sigma)$:

$$\mu = \begin{bmatrix} c_{xA} \\ c_{yA} \end{bmatrix}, \quad \sigma = \begin{bmatrix} \frac{w_A^2}{4} & 0 \\ 0 & \frac{h_A^2}{4} \end{bmatrix} \tag{6}$$

第 2 步计算两个框之间的 Wasserstein 距离并归一化, 同理, w_B和h_B为边界框 B 的宽度和高度, 中心坐标为(c_{xB}, c_{yB}). 由此简化得出高斯分布N_A和N_B之间 Wasserstein 距离为:

$$W_2^2(N_A, N_B) = \left\| \left(\left[c_{xA}, c_{yA}, \frac{w_A}{2}, \frac{h_A}{2} \right]^{\mathrm{T}}, \left[c_{xB}, c_{yB}, \frac{w_B}{2}, \frac{h_B}{2} \right]^{\mathrm{T}} \right) \right\|_2^2 \tag{7}$$

随后, 进行归一化, 即可得出边界框之间的相似性度量:

$$NWD(N_A, N_B) = \exp\left(-\frac{\sqrt{W_2^2(N_A, N_B)}}{C}\right) \tag{8}$$

其中, 常数 C 与图像尺寸密切相关, 在实际应用中 一般设置为对应数据集中图像的平均绝对尺寸以获得最优性能. 最后, 可以得出 NWD 损失的定义为:

$$L_{\mathrm{NWD}} = 1 - NWD(N_A, N_B) \tag{9}$$

使用 NWD 的优势在于, 即使预测框与真实框之间没有重叠区域, NWD 仍然能够提供有效的梯度信号, 从而显著提升模型在小目标检测和低重叠场景下的性能. 此外, NWD 对边界框的尺度变化不敏感, 能够更好地平衡多尺度目标的优化过程. 相关研究的实验表明, 在密集小目标检测任务中, NWD 相较于传统的 IoU 方法具有更高的鲁棒性和检测精度.

3 实验结果与分析

实验分别在两个数据集上进行, 分别是 VisDrone-DET2019[36]和 DOTA v1.0[37], 前者主要用来验证本文方法的有效性, 后者主要测试本方法的泛化性.

3.1 实验环境

本文实验基于 64 位 Ubuntu 20.04.5 LTS 操作系统, 具体实验环境如表 1 所示.

表 1 实验环境

参数	设置/版本
操作系统	Ubuntu 20.04.5 LTS
Python	3.8.10
CUDA	11.8
GPU	RTX 3080 (20 GB)
CPU	Intel(R) Xeon(R) Platinum 8352V CPU
内存	48 GB
PyTorch	2.0.0

3.2 数据集

采用 VisDrone-DET2019 和 DOTA v1.0 为本文实验数据集.

VisDrone-DET2019 数据集由天津大学 AISKYEYE 团队收集. 包含了 10 类航拍视角下的目标, 有城市道路上的行人、个人、自行车、汽车等. 其中, 极小目标 (小于 16×16 像素) 的占比达到 12%, 而小目标 (小于 32×32 像素) 的占比更是高达 45%. 特别是在人和行人类别中, 小目标的占比分别高达 77% 和 65%. 该数据集对于本文关注的小目标检测算法具有重要研究意义.

DOTA v1.0 数据集是由武汉大学 2018 年发布的专为航拍图像目标检测设计的大规模数据集. 该数据集旨在解决航拍图像中目标检测的独特挑战, 如目标方向多样性、尺度变化大以及小目标密集分布等问题. DOTA v1.0 包含 2 806 张航拍图像, 每张图像尺寸在 800×800–4000×4000 像素之间, 尺度范围涵盖了从飞机和船只等小尺度目标到足球场和飞机场等大尺度目标, 本文使用该数据集验证所提算法的泛化性.

3.3 评价指标

本研究采纳了目标检测领域内经典评估指标, 使用精确度 (precision, P)、召回率 (recall, R) 和全类平均精确率 (mean average precision 50, mAP50) 衡量模型检测精度; 为直观反映模型轻量化效果, 在部分实验中, 亦纳入了浮点运算次数 (FLOPs)、参数量 (parameters) 及模型体积 (model size) 和 FPS (frames per second) 衡量模型检测速度.

精确度 (P) 是指在所有被判定为正类的样本中, 实际为正类的样本所占的比例; 召回率 (R) 则是指在所有实际为正类的样本中, 成功被识别为正类的样本所占的比例. mAP50 表示在预测框与真实框的交并比 (*IoU*) 阈值为 0.5 时, 所有类别平均精度 (AP) 的平均值. FLOPs、参数量、帧数和模型大小则用于评估模型的计算需求、复杂性以及存储空间需求.

3.4 实验结果与分析

3.4.1 不同数据集对比实验

为体现本文改进算法的有效性, 选取相似规模的主流目标检测算法进行对比, 表 2 和表 3 展示了相同条件下, 不同算法在 VisDrone-DET2019 和 DOTA v1.0 上的对比结果. 在 VisDrone-DET2019 数据集上, 本文提出的 MobileSAM-YOLO 相较于其他相似规模的检测算法平均检测精度更高, 达到 41.4%, mAP50 指标分别提升了 25.1%、23.6%、3.2% 和 2.7%.

表 2 不同算法在 VisDrone-DET2019 数据集上的对比实验 (%)

算法	mAP50											P	R
	Ped	Peo	Bic	Car	Van	Truck	Tri	Awn	Bus	Mot	all		
YOLOv5s	40.1	32.4	11.1	73.1	36.1	27.3	19.2	9.8	45.4	38.5	33.1	45.1	32.8
YOLOX-s	39.1	33.2	11.8	72.1	37.2	29.2	17.3	9.7	46.1	39.5	33.5	35.6	40.6
YOLOv8s	43.4	33.9	13.9	**79.7**	45.5	37.1	27	15.6	59.7	44.7	40.1	52.1	38.9
YOLOv10s	43.6	34.3	13.8	78.5	**46.2**	**37.5**	27.6	15.4	**60.3**	**46**	40.3	50.6	37.5
Ours	**45.4**	**35.8**	**16.1**	79.4	44.9	36.8	**30.2**	**17.5**	60	45.6	41.4	**58.6**	**42.3**

表 3 不同算法在 DOTA v1.0 数据集上的对比实验 (%)

算法	mAP50																P	R
	Pla	Sh	Sto	Bad	Ten	Bas	Gro	Har	Bri	Lar	Sma	Hel	Rou	Soc	Swi	all		
YOLOv5s	65.8	55.5	27.9	39.6	88.8	37.9	33.8	67.2	6.3	75.3	55.3	28.9	12.9	33.2	33	44.1	68.8	35.6
YOLOX-s	70.1	56.1	30.4	40.2	87.5	38.6	33.6	67.4	6.7	75.5	58.6	**33.2**	13.6	35.4	**35**	45.5	69.2	36.8
YOLOv8s	70.1	56.3	28.7	42.2	**91.6**	38.2	**34.6**	66.9	6.7	77.6	54.3	30.9	12.6	35	32.9	45.2	70.1	41.4
YOLOv10s	73.2	57.2	29.2	42.1	91.2	35.6	33.2	**70.1**	7.1	**79.2**	54.5	32.1	13.1	**36.3**	34.8	45.9	72.1	43.2
Ours	**73.4**	**62.3**	**32.4**	**45.3**	91.5	**40.3**	33.4	69.9	**15.2**	79	**64.6**	32.9	**20.2**	36.2	34.7	48.8	**80.2**	**55.2**

从表 2 可以看出, MobileSAM-YOLO 的精度 P 和召回率 R 相较于其他算法都有很大的提升, 意味着其在检测任务中既能准确地识别出目标, 又能尽量减少漏检, 具有很好的鲁棒性. 对于行人、人、自行车等小目标, MobileSAM-YOLO 的检测精度远高于其他模型, 验证了 MobileSAM 编码器针对小目标检测方面的能

力. 然而, 在汽车、面包车、卡车等类别中, MobileSAM-YOLO 模型未能达到最高的精度, 但也维持了相对较好的水平, 是在可接受的范围之类, 可能原因大概是由于参数量的降低导致了模型没法充分表达组内和组间差异, 后续工作中针对此问题可以开展进一步的研究.

在 DOTA v1.0 数据集上, 由表 3 可以看出, 本文方法在储藏罐、小型车、桥梁等小目标相比于其他算法有较大的提升, 精度 P 和召回率 R 也得到了相应的提高. 体现出本文方法在不同数据集上都能有效改进小目标检测性能, 具有很好的泛化性.

3.4.2 消融实验

为验证各模块的性能以及贡献程度, 进行了单模块消融实验和多模块消融实验.

通过表 4 的单模块消融可以看出, M (MobileSAM) 的加入使模型参数量有了明显的下降, 下降了约 21.4%; 虽然计算量 FLOPs 提高了, 不过由于 MobileSAM 优秀的架构设计, 推理速度 FPS 不降反升; 得益于强大的特征提取能力, mAP50 提高了 1.7%, 证明该模块的有效性. L (LSCD) 模块通过共享卷积和特征缩放在降低参数量的同时并提高了特征融合能力, 使得 mAP50 提升了 0.5%, 证明其在降低参数量的同时保持精度的有效性. N (NWD) 没有对参数量和计算量产生影响, 但仍轻微提高了 mAP50, 体现了其能够有效提高精度.

表 4 VisDrone-DET2019 单模块消融实验

Baseline	M	L	N	Para (M)	FLOPs (G)	mAP50 (%)	FPS (f/s)
—	—	—	—	11.2	28.6	40.1	133
√	√	—	—	8.8	38.2	40.8	243
√	—	√	—	9.4	25.9	40.3	145
√	—	—	√	11.2	28.6	40.2	133

通过表 5 的在 VisDrone-DET2019 上多模块消融可以看出, MobileSAM 和 LSCD 两者结合, MobileSAM 提取的优质特征和小目标细节经过 LSCD 进一步的特征融合, 有效提高了精度并降低了参数量. mAP50 提高了 2.5% 的同时参数量降低了 33.9%. LSCD 和 NWD 的结合, 由于没能接收优质的小目标细节特征, 检测效果没能达到最佳, 参数量和检测速度也没有较大提升. MobileSAM 和 NWD 的结合, 由于没有 LSCD 的参数共享设计和高效的特征融合, 精度和计算性能的都没有较大的提高. 这 3 个模块的共同作用, 解决了小目标和轻量化的问题, 参数量下降了 33.9%, mAP50 提升了 3.2%, 达到了预期目标.

为验证泛化性, 增加了在 DOTA v1.0 数据集上的多模块消融实验, 通过表 6 的实验结果可以看出, 在 DOTA v1.0 数据集上, 实验数据具有相似的特征: 各个模块都对算法有或大或小的贡献, 因此, 可以得出本文方法具有优秀的泛化性.

表 5 VisDrone-DET2019 多模块消融实验

Baseline	M	L	N	Para (M)	FLOPs (G)	mAP50 (%)	FPS (f/s)
—	—	—	—	11.2	28.6	40.1	133
√	√	—	—	8.8	38.2	40.8	243
√	√	√	—	7.4	35.6	41.1	250
√	√	—	√	8.8	38.2	40.2	243
—	—	√	√	9.4	25.9	40.3	145
—	√	√	√	7.4	35.6	41.4	250

表 6 DOTA v1.0 多模块消融实验

Baseline	M	L	N	Para (M)	FLOPs (G)	mAP50 (%)	FPS (f/s)
—	—	—	—	11.2	28.6	45.2	288
√	√	—	—	8.8	38.2	47.9	320
√	√	√	—	7.4	35.6	48.5	333
√	√	—	√	8.8	38.2	47.9	320
—	—	√	√	9.4	25.9	46.3	291
—	√	√	√	7.4	35.6	48.8	333

3.4.3 可视化分析

为全面对比 MobileSAM-YOLO 和 baseline 模型 YOLOv8s 在不同场景下的检测效果, 从我们 VisDrone-DET2019 数据集中选取了 3 张具有代表性的场景: 密集场景、垂直场景和遮挡场景, 并可视化了在这些场景中的检测结果, 如图 9 所示.

由图 9 可以看出, 在第 1 张垂直、密集且有遮挡的场景中, 由于镜头距离较远, 路两边的行人和摩托车之类的目标像素占比很少, YOLOv8s 存在较多的漏检现象, 而本文方法能够有效地缓解垂直角度下的小目标漏检现象. 图片左下位置有一辆被树遮挡的小汽车, 本文方法成功地检测到了而 YOLOv8s 未能检测出来. 在对图中所有目标的检测中, 本文方法检测出来的目标都具有更高的置信度, 体现出不俗的鲁棒性. 在第 2 张夜间、密集的复杂场景中, 本文方法成功地检测出右下角的露半个身子的小目标行人和最远处站着的小目标行人, 体现出强大的小目标检测能力; 对于手推车上的婴儿, YOLOv8s 错误的检测成了摩托车而本文方法却能够成功检出, 体现出强大的细节感知能力. 在第 3 张有遮挡的斜拍的日常街景中, 本文方法成功检出 YOLOv8s 未能检出的最远处被树遮挡的停在路边的小汽车. 总的来说, 在背景复杂、有遮挡、小目标密集场景下的精确率和检出率方面, 本文方法明显优于基线模型, 在模型轻量化的情况下具备更优秀的无人机视角下小目标检测性能.

(a) 垂直、密集、遮拦 (b) 夜间、密集 (c) 遮挡

图 9 不同场景检测效果对比

4 结束语

本文针对当前目标检测算法在无人机图像上小目标检测精度与轻量化之间难以平衡的问题, 本文提出了一种改进的基于 YOLOv8 的轻量级无人机图像目标检测算法. 具体来说, MobileSAM 的引入显著减少了模型参数, 能够提取高质量的多尺度特征和小目标细节信息, 从而更有效的检测小目标; 并且, 其编码器经过海量数据的大规模训练, 具有很好的泛化能力, 引入后能提高在不同任务和数据集上的表现. LSCD 的加入进一步降低了参数量, 同时保持了精度的提升, 这表明 LSCD 在减少模型复杂度的同时, 也保留了关键特征信息. 最后, NWD 的引入虽然对参数量和 FLOPs 没有直接影响, 但通过优化回归损失函数, 实现了精度的微小提升, 进一步证明了本文改进方法的有效性. 综合来看, 本文提出的改进方法不仅提高了模型的检测精度和速度, 还展示了良好的泛化性能, 为无人机图像小目标检测提供了新的思路.

参考文献

1 Bouguettaya A, Zarzour H, Kechida A, *et al*. Vehicle detection from UAV imagery with deep learning: A review. IEEE Transactions on Neural Networks and Learning Systems, 2022, 33(11): 6047–6067. [doi: 10.1109/TNNLS.2021.3080276]

2 李利霞, 王鑫, 王军, 等. 基于特征融合与注意力机制的无人机图像小目标检测算法. 图学学报, 2023, 44(4): 658–666.

3 Kisantal M, Wojna Z, Murawski J, *et al*. Augmentation for small object detection. arXiv:1902.07296, 2019.

4 Zou ZX, Chen KY, Shi ZW, *et al*. Object detection in 20 years: A survey. Proceedings of the IEEE, 2023, 111(3): 257–276. [doi: 10.1109/JPROC.2023.3238524]

5 Girshick R, Donahue J, Darrell T, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014. 580–587.

6 Girshick R. Fast R-CNN. Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015. 1440–1448.

7 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149. [doi: 10.1109/TPAMI.2016.2577031]

8 Redmon J, Divvala S, Girshick R, *et al*. You only look once: Unified, real-time object detection. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 779–788.

9 Liu W, Anguelov D, Erhan D, *et al*. SSD: Single shot multibox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 21–37.

10 韩俊, 袁小平, 王准, 等. 基于 YOLOv5s 的无人机密集小目标检测算法. 浙江大学学报 (工学版), 2023, 57(6): 1224–1233.

11 史涛, 崔杰, 李松. 优化改进 YOLOv8 实现实时无人机车辆检测的算法. 计算机工程与应用, 2024, 60(9): 79–89. [doi:

10.3778/j.issn.1002-8331.2312-0291]
12 Wang G, Chen YF, An P, *et al*. UAV-YOLOv8: A small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. Sensors, 2023, 23(16): 7190. [doi: 10.3390/s23167190]
13 吴明杰, 云利军, 陈载清, 等. 改进 YOLOv5s 的无人机视角下小目标检测算法. 计算机工程与应用, 2024, 60(2): 191–199. [doi: 10.3778/j.issn.1002-8331.2307-0223]
14 Howard A, Sandler M, Chen B, *et al*. Searching for MobileNetV3. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 1314–1324.
15 Howard AG, Zhu ML, Chen B, *et al*. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861, 2017.
16 Sandler M, Howard A, Zhu ML, *et al*. MobileNetV2: Inverted residuals and linear bottlenecks. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 4510–4520.
17 Zhang XY, Zhou XY, Lin MX, *et al*. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 6848–6856.
18 Han K, Wang YH, Tian Q, *et al*. GhostNet: More features from cheap operations. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 1577–1586.
19 Wang XF, Wu ZW, Jia M, *et al*. Lightweight SM-YOLOv5 tomato fruit detection algorithm for plant factory. Sensors, 2023, 23(6): 3336. [doi: 10.3390/s23063336]
20 Liu SH, Zha JL, Sun J, *et al*. EdgeYOLO: An edge-real-time object detector. Proceedings of the 42nd Chinese Control Conference (CCC). Tianjin: IEEE, 2023. 7507–7512.
21 Lee J, Park S, Mo S, *et al*. Layer-adaptive sparsity for the magnitude-based pruning. arXiv:2010.07611, 2020.
22 唐克, 魏飞鸣, 李东瀛, 等. 基于改进 YOLOv8 的轻量化无人机图像目标检测算法. 计算机工程, 1–11. https://link.cnki.net/urlid/31.1289.tp.20241204.0953.003. (2024-12-04).
23 Lin TY, Dollár P, Girshick R, *et al*. Feature pyramid networks for object detection. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 936–944.
24 Liu S, Qi L, Qin HF, *et al*. Path aggregation network for instance segmentation. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 8759–8768.
25 Zhang CN, Han DS, Qiao Y, *et al*. Faster segment anything: Towards lightweight SAM for mobile applications. arXiv:2306.14289, 2023.
26 Kirillov A, Mintun E, Ravi N, *et al*. Segment anything. Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision. Paris: IEEE, 2023. 3992–4003.
27 Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv:1503.02531, 2015.
28 Wu YX, He KM. Group normalization. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018. 3–19.
29 Misra D. Mish: A self regularized non-monotonic activation function. arXiv:1908.08681, 2019.
30 Chen ZX, He ZW, Lu ZM. DEA-Net: Single image dehazing based on detail-enhanced convolution and content-guided attention. IEEE Transactions on Image Processing, 2024, 33: 1002–1015. [doi: 10.1109/TIP.2024.3354108]
31 Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. Proceedings of the 32nd International Conference on Machine Learning. Lille: JMLR.org, 2015. 448–456.
32 Tian Z, Shen CH, Chen H, *et al*. FCOS: A simple and strong anchor-free object detector. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(4): 1922–1933.
33 Zheng ZH, Wang P, Ren DW, *et al*. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. IEEE Transactions on Cybernetics, 2022, 52(8): 8574–8586. [doi: 10.1109/TCYB.2021.3095305]
34 Rezatofighi H, Tsoi N, Gwak JY, *et al*. Generalized intersection over union: A metric and a loss for bounding box regression. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 658–666.
35 Wang JW, Xu C, Yang W, *et al*. A normalized Gaussian Wasserstein distance for tiny object detection. arXiv:2110.13389, 2021.
36 Du DW, Zhu PF, Wen LY, *et al*. VisDrone-DET2019: The vision meets drone object detection in image challenge results. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop. Seoul: IEEE, 2019. 213–226.
37 Xia GS, Bai X, Ding J, *et al*. DOTA: A large-scale dataset for object detection in aerial images. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 3974–3983.

(校对责编: 张重毅)