doi:10.6041/j.issn.1000-1298.2026.03.007

基于 YOLO－DRR 与实时处理 FPGA 边缘计算平台的低空视角下柑橘冠层分割

吕石磊[1,2]　陈洁瑜[1]　高　鹏[1,2]　李　震[1,2]　刘雪雅[1,3]　李子杰[1]
陈嘉鸿[1]　高松茂[1]　陈毅聪[1,3]

（1.华南农业大学人工智能与低空技术学院，广州 510642；2.国家柑橘产业技术体系机械化研究室，广州 510642；
3.人工智能与数字经济广东省实验室（广州），广州 510335）

摘要：冠层是柑橘果树光合作用的主体，直接影响果树的生长、产量与果实品质，是果树健康与丰产的基础。通过对冠层结构的监测，可以及时调整修剪、灌溉、施肥等种植管理措施，从而优化冠层内部环境，促进果树的健康生长和发育。针对柑橘果园种植密集、冠层之间重叠遮挡影响果树生长效率和产量品质等问题，本研究构建了自然环境下的柑橘果树冠层数据集，提出了 YOLO－DRR（YOLO v5s－seg－DSConv－RFEM－RIME）轻量化分割模型，同时为了提高实时性、降低功耗和便于在果园使用，将该模型部署到便携的边缘计算平台中。本研究以 YOLO v5s－seg 模型为基础，使用多尺度特征提取模块对骨干网络进行改进，提高了多尺度目标的分割精度；使用分布移位卷积模块替换了颈部网络中的 C3 模块，降低了卷积核中的内存使用量，从而提高了运算速度；采用霜冰优化算法优化 YOLO－DRR 模型超参数，利用群智能的迭代机制进一步提升模型性能。最后，将 YOLO－DRR 模型移植部署至 FPGA 边缘计算平台，利用 FPGA 强大的边缘计算能力，确保数据处理的实时性，更加有效地利用硬件资源，减少功耗和散热问题，实现复杂背景下柑橘果树冠层长时间实时分割监测的要求。实验结果表明，YOLO－DRR 模型对冠层进行分割的精确度达到 86.34%，召回率达到 88.68%，mAP@0.5 达到 93.41%，mAP@0.5:0.95 达到 63.13%，移植至边缘计算平台后，检测速度达到了 19 f/s，功耗仅为 22 W，这表明本研究提出的模型具有在复杂背景下对柑橘果树冠层进行实时分割的能力，能够满足果园实时监测冠层生长环境的需求。

关键词：柑橘果树冠层；分割；YOLO－DRR；超参数优化；FPGA

中图分类号：TP391　　**文献标识码**：A　　**文章编号**：1000-1298(2026)03-0067-10　　**OSID**：

Implementation of Low-altitude Citrus Canopy Segmentation Based on YOLO－DRR and Real-time FPGA Edge Computing

LÜ Shilei[1,2]　CHEN Jieyu[1]　GAO Peng[1,2]　LI Zhen[1,2]　LIU Xueya[1,3]　LI Zijie[1]
CHEN Jiahong[1]　GAO Songmao[1]　CHEN Yicong[1,3]

(1. *College of Artificial Intelligence and Low-Altitude Technology*, *South China Agricultural University*, *Guangzhou* 510642, *China*
2. *Division of Citrus Machinery*, *China Agriculture Research System*, *Guangzhou* 510642, *China*
3. *Guangdong Artificial Intelligence and Digital Economy Laboratory* (*Guangzhou*), *Guangzhou* 510335, *China*)

Abstract: The canopy is the primary source of photosynthesis in citrus fruit trees, and it has a direct impact on the growth, yield, and fruit quality of the trees. It is the foundation for healthy and productive fruit trees, and efficient and accurate monitoring of canopy growth is especially important. Monitoring the canopy structure allows planting management measures such as pruning, irrigation, and fertilization to be adjusted promptly, optimizing the internal environment of the canopy and promoting healthy fruit tree growth and development. A dataset of citrus fruit tree canopies in a natural environment was created and a lightweight YOLO－DRR segmentation model (YOLO v5s－seg－DSConv－RFEM－RIME) was proposed to address the issues of dense planting in citrus orchards and overlap shading between canopies,

收稿日期：2025－09－12　修回日期：2025－10－15

基金项目：国家自然科学基金项目（32271997、31971797）、广州市重点研发计划项目（2024B03J1309）和国家现代农业产业技术体系项目（CARS－26）

作者简介：吕石磊（1984—），男，教授，博士，主要从事农业信息化研究，E-mail：lvshilei@scau.edu.cn

通信作者：李震（1981—），男，教授，博士，主要从事农业工程与智慧农业研究，E-mail：lizhen@scau.edu.cn

which affect fruit tree growth efficiency and yield quality. Meanwhile, the model was deployed on a portable edge computing platform to improve real-time performance, reduce power consumption, and make it easier to use in inter-orchard scenarios. Firstly, the segmentation accuracy of multi-scale targets was improved based on the YOLO v5s - seg model by using the scale-aware RFE module (RFEM) for backbone networks. Secondly, the use of the distribution shifting convolution module (DSConv) to replace the C3 module in the neck network reduced memory usage in the convolutional kernel, thereby increasing the speed of operations. Thirdly, the rime optimization algorithm (RIME) was used to optimize the hyperparameters of the YOLO - DRR model, and the iterative mechanism of swarm intelligence was utilized to further improve the model performance. Finally, the YOLO - DRR model was transplanted and implemented on the FPGA edge computing platform. The FPGA device was highly environmentally adaptable and can operate reliably in a wide range of temperature and humidity conditions, ensuring the device's dependability and stability in the complex and changing environment of the orchard. Simultaneously, the powerful edge computing capability of FPGA was used to ensure real-time data processing, more efficient use of hardware resources, reduction of power consumption and heat dissipation issues, and the realization of the requirement for long-term real-time segmentation monitoring of citrus fruit tree canopies in complex environments. The experimental results showed that the YOLO - DRR model segmented the canopy with 86.34% precision, 88.68% recall, 93.41% mAP@0.5, and 63.13% mAP@0.5:0.95. After porting it to the edge computing platform, the detection speed was increased to 19 f/s while consuming only 22 W of power. This suggested that the model proposed was capable of segmenting the canopy of a citrus fruit tree in the complex context of real-time canopy segmentation, which can meet the demand for real-time monitoring of the canopy growth environment in the orchard.

Key words: canopy of citrus trees; segment; YOLO - DRR; hyperparameter optimization; FPGA

0 引言

柑橘在全球范围内种植面积和总产量均位居水果之首[1],根据国家统计局及行业报告,截至 2022 年,我国柑橘园面积达到了 299.58 万公顷,2023 年柑橘产量进一步增长到了 6 433.76 万吨。随着消费者健康意识的不断增强,以及线上贸易的持续繁荣,柑橘鲜果市场需求十分旺盛,如何有效提升柑橘的产量并培育出更高品质的果实,已成为广大果农关注与亟待解决的问题。其中,柑橘果树最重要的组成部分是冠层,冠层是果树进行主要生理活动的重要场所,直接影响到果树的光合作用效率、果实品质以及整体产量,准确获取果树冠层信息还有助于果园监测树木长势、预防树木病虫害等[2-4]。然而,我国柑橘果园主要分布在丘陵和山地地区,果园地形复杂,果树之间种植密集[5],冠层相对较小,之间也存在互相遮挡情况,传统的柑橘果树冠层人工调查和野外测量方法费时费力,不能满足冠层信息的高效、高精度采集。随着科技发展,利用无人机进行遥感已成为协助果农进行精准农业的重要新技术,与卫星遥感相比,无人机采集的遥感图像具有更高的分辨率和更好的时效性[6-7]。因此,针对我国柑橘果园这一特殊环境,采用无人机遥感技术获取果树冠层图像,对冠层的生长进行监测,进而对冠层实施精准分割,可以更准确地计算冠层面积、冠幅等参数来判断冠层的生长状况,为果园管理快速提供科学依据,减少人工干预,从而做出最优的管理决策,优化生长过程[8-10]。

传统的冠层图像分割算法如 OTSU 法[11]、分水岭[12]、图割法[13]和 K-means 聚类[14]等,通常依赖颜色、纹理、灰度等基本特征进行分割,在背景简单的场景中表现良好,但难以应对复杂自然环境。这类方法往往忽略上下文信息,易受背景干扰,导致分割精度低,无法有效处理重叠或遮挡冠层区域。相比于深度学习方法,往往会忽略图像的上下文信息,导致分割准确率低下,难以处理复杂背景的干扰,出现遗漏果树、无法分割重叠或遮挡区域果树的情况,不能满足精准分割的要求[15-16]。近年来,随着深度学习技术的不断开发,与传统依赖先验知识的特征提取算法相比,深度卷积神经网络对几何变换、变形和光照具有一定的不变性,有效克服了物体外观变化带来的识别困难,基于深度学习的分割算法性能大大提升,对大规模实现精准农业监测具有重要意义[17-21]。在山地果园这类地理位置偏远、地形复杂且条件受限的农业区域,部署大型计算平台面临着诸多挑战与不便,因此要选择合适的模型,充分利用边缘计算等先进技术,以实现对果园果树冠层的实时分割监测。

在深度学习中,针对特定数据集设计和调整模型架构对提升精度至关重要,但超参数优化同样不

可或缺。然而，超参数组合复杂，传统手动调参效率低下且难以获得最优解，因此常采用智能优化算法[22]。部分智能算法参数多、迭代次数高，会显著增加计算成本，因此需权衡优化效果与计算效率，避免资源过度消耗。

综上，目前果树冠层分割技术面临着三大难题：①精度要求，果树冠层互相遮挡的情况下，达到理想分割标准。②速度要求，要开发一种高效运行的轻量级模型，能够部署在便携的边缘计算平台上，同时确保处理速度满足实时分割要求。③便携要求，服务器运算平台通常被部署在数据中心或特定机房，不适合在山地果园移动使用。因此，本文构建自然环境下的柑橘果树冠层数据集，提出改进的 YOLO－DRR 模型，在主干网络中的 SPFF 模块后引入多尺度特征提取模块，使模型学习特征图的不同感受野，增强特征金字塔表示，从而提高模型分割能力。将 YOLO－DRR 模型颈部网络中的 C3 模块替换为内存使用量更低、计算速度更高、即插即用的分布移位卷积模块，从而降低模型的内存占用量。通过霜冰优化算法（Rime optimization algorithm，RIME）对 YOLO－DRR 模型的超参数进行优化，不断实验得出最适用于柑橘果树冠层分割任务的超参数，进一步提升模型的性能。将优化 YOLO－DRR 模型进行量化、编译处理，部署到实时性高、功耗低、便携的 FPGA（Field programmable gate array）平台[23－25]，采集冠层画面以实时进行柑橘果树冠层的分割。

1　实验数据采集与处理方法

本研究实验数据主要采集于广东省广州市从化区柑橘星球果园（23.64°N，113.50°E），采集时间为 2023 年 12 月 13 日，使用大疆御 MAVIC AIR2 型无人机在自然环境下对柑橘果树进行采集。该果园涵盖了平原与山地 2 种典型地形，所选树冠冠层密度属于当地中等水平，能够代表柑橘果树冠层的普遍情况。考虑柑橘果树高为 2.5 m 左右，为确保每幅图像都能拍摄到目标果树的完整树冠，无人机距离果园地面上方 10 m，摄像头 90°垂直向下进行拍摄。

构建的柑橘果树冠层数据集共有 237 幅原始图像，如图 1 所示。通过加入调节亮度、高斯模糊和添加椒盐噪声等数据增强方法将数据集图像扩充到 948 幅。使用 ISAT with segment anything 交互式半自动图像标注工具对数据集进行标注，标注后的数据集以比例 8∶2 划分为训练集（756 幅）和验证集（192 幅）。数据划分时严格遵守了“果树冠层”的划分原则，即同一棵果树上的所有图像只会出现在训练集或验证集中的一方，绝不会交叉。



(a) 密集目标



(b) 稀疏目标

图 1　部分柑橘果树冠层的航拍图像

Fig. 1　Aerial imagery of partial citrus tree canopies

2　YOLO－DRR 模型设计

针对果园自然环境下密集分布的柑橘果树冠层难以精准分割等问题，本研究提出了柑橘果树冠层轻量化 YOLO－DRR 分割模型，如图 2 所示为该分割模型的网络结构示意图。首先，模型以 YOLO v5s－seg 为基础，在骨干网络中添加了多尺度特征提取模块，其通过扩展卷积来充分利用特征映射中感受野的优势，增加了特征图的感受野，从而提高了多尺度目标分割的精度。其次，使用分布移位卷积模块替换颈部网络中的 C3 模块，降低了卷积核中的内存使用量，并提高了运算速度。最后，使用霜冰优化算法，通过模拟软霜粒子的运动和硬霜粒子之间的穿刺行为，对模型的超参数进行优化，进一步提升模型的分割性能。

2.1　多尺度特征提取模块 RFEM

在很多场景下，图像中通常存在不同尺度的果树冠层，分割出不同尺度的冠层是本算法一个非常重要的任务。引入多尺度训练，感受野增加的同时，语义信息会相应丰富，但局部邻域信息可能会相应缺失。因此，引入了 RFEM 模块[26]来学习特征图的不同感受野，增强特征金字塔网络表示。RFEM 模块通过扩展卷积来充分利用特征映射中感受野的优势，其可分为基于扩展卷积的多分支和采集加权层两部分，如图 3 所示。多分支部分分别取 1、2、3 作为不同扩展卷积的速率，它们都使用固定大小的卷积核。此外，还增加了残差连接，以防止训练过程中梯度爆炸和消失的问题。采集加权层用于收集来自不同分支的信息，并对每个分支的特征进行加权。

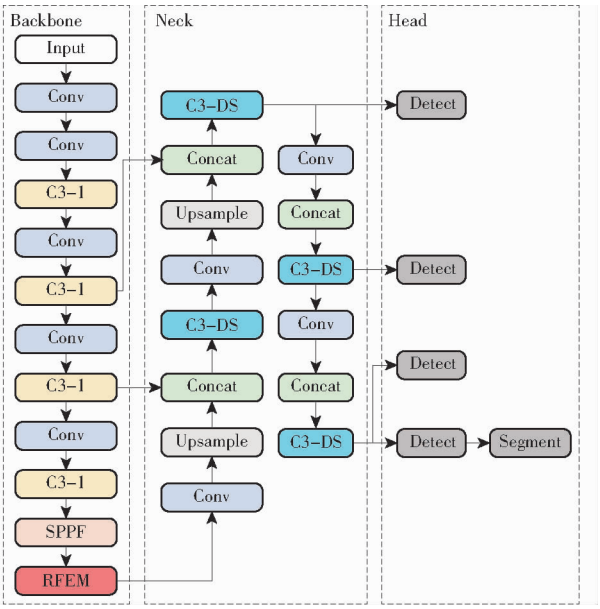

图 2 YOLO－DRR 模型网络结构示意图

Fig. 2 Schematic of network structure of YOLO－DRR model

因此，RFEM 模块一方面通过显著减少模型所需参数的数量，有效地降低模型训练过程中潜在的过拟合风险，另一方面，确保每个训练样本都能被充分利用，从而提高模型的泛化能力和数据利用效率。

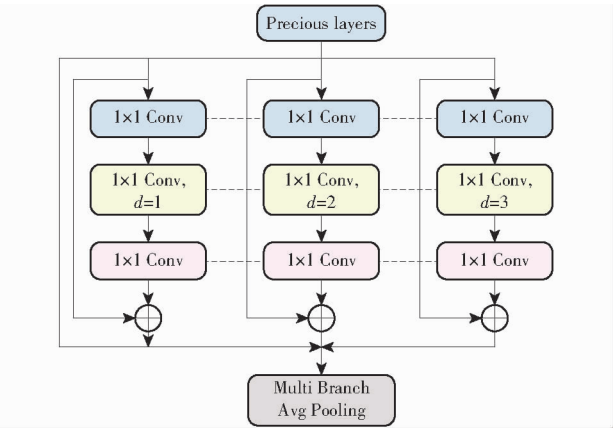

图 3 多尺度特征提取模块 RFEM

Fig. 3 Multi-scale feature extraction module (RFEM)

2.2 分布移位卷积模块 DSConv

运行卷积神经网络所需的大部分内存和计算工作都用于卷积层，为了使网络运行得更快、更高效，必须提高卷积层的计算，而 DSConv 模块[27]的内存使用量低至标准卷积的 1/10，速度高达标准卷积的 10 倍，同时能即插即用，因此，将颈部网络中的 C3 模块替换为 DSConv 模块。DSConv 是一种深度可分离卷积模块的变体，它由 Variable Quantized Kernel (VQK) 和 Distribution Shift 两部分组成。DSConv 的设置如图 4 所示，设置原始卷积张量为 (C_{ho}, C_{hi}, k, k)，其中 C_{ho} 为下一层的通道数，C_{hi} 为当前层的通道数，k 是卷积核的宽度和高度。

可变量化核 VQK 是 DSConv 的量化组件，它能够执行更快且内存高效的乘法运算。VQK 的张量仅包含可变位数的整数值，大小与原始卷积张量 (C_{ho}, C_{hi}, k, k) 相同，参数值被设定为原始浮点模型量化的值，一旦被设定就无法更改。

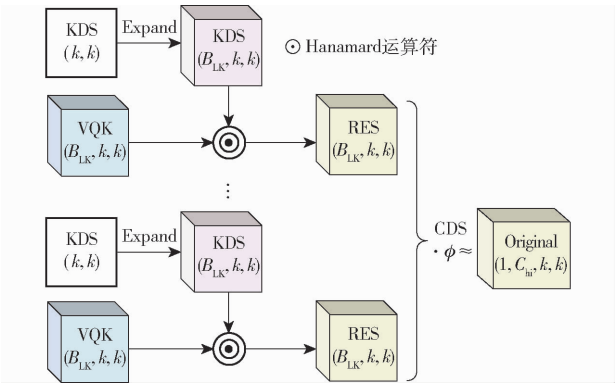

图 4 分布移位卷积模块 DSConv

Fig. 4 Distribution shift convolution module (DSConv)

分布移位分为核分布偏移器 (Distribution shift in the kernel, KDS) 和通道分布偏移器 (Distribution shift in the channels, CDS)，它目的是通过使用量化和分布移位来模拟卷积核的分布。KDS 在 VQK 的每个 (1, B_{LK}, 1, 1) 切片中进行分布偏移，其中 B_{LK} 为块大小超参数，张量大小为 $2\left(C_{ho}, \mathrm{CEIL}\left(\frac{C_{hi}}{B_{LK}}\right), k, k\right)$，其中 CEIL($x$) 是向上取整运算符，保存单精度值。对 KDS 和 CDS 进行初始化十分必要，在使网络最接近最优值前，都需要不断微调，这里，KDS 中用 ξ 表示缩放，用 ξ_s 表示偏置，CDS 中用 ϕ 和 ϕ_s 表示。

2.3 YOLO－DRR 模型超参数优化

为了获得最佳的模型性能，往往需要通过调整超参数进行优化。本研究采用了霜冰优化算法，RIME[28]对 YOLO－DRR 模型超参数进行优化设计，利用群智能的寻优迭代机制快速收敛并获得最优超参数组合，进而提升模型性能，解决人工调参过于依赖经验的问题。

RIME 算法通过模拟软霜粒子的运动来进行算法搜索。首先，在微风环境中，软霜粒子生长具有强烈的随机性，使得霜粒子能够自由覆盖对象表面，但在同一方向上生长速度缓慢，通过对软霜的运动特性进行模拟，如图 5 所示，提出软霜搜索策略，使算法能够快速覆盖整个搜索空间，避免陷入局部最优解。

其次，在强风条件下，硬霜的生长更加简单而具有规律，但因为生长方向相同，各个霜冰之间容易交叉，出现霜冰穿刺现象，通过对穿刺现象进行模拟，如图 6 所示，提出硬霜穿刺机制，改善算法的收敛性

和跳出局部最优的能力。

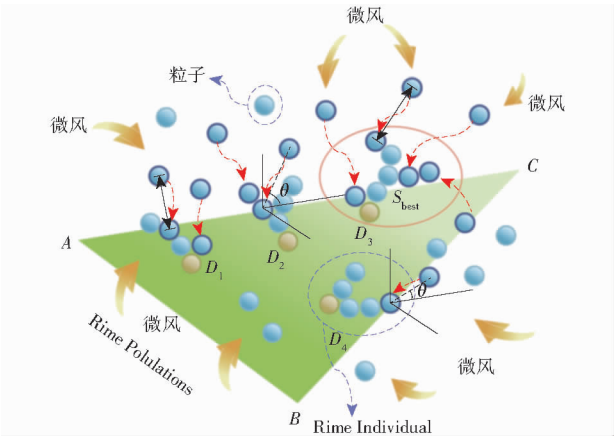

图 5　软霜粒子运动

Fig. 5　Movement of soft frost particles

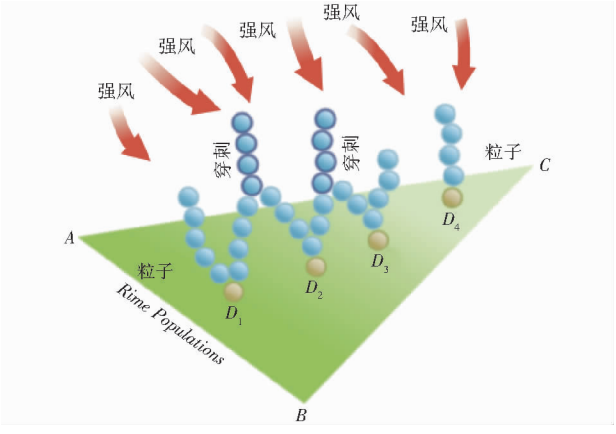

图 6　硬霜粒子运动

Fig. 6　Hard frost particle movement

最后，提出参与种群更新的积极贪婪选择机制，将霜粒子更新后的适应度与其更新前的适应度进行比较，如果更新后的适应度优于更新前，则执行替换操作，同时替换 2 个霜粒子的位置，确保种群在每次迭代中朝更优方向演化。

基于 RIME 算法的 YOLO－DRR 模型超参数优化流程如图 7 所示，计算步骤主要包括：

(1)数据集划分：采集数据集，并将其划分为训练集和验证集。

(2)初始化参数和种群：设置相关参数，包括种群数目 N_{popnum}，最大迭代次数 T，超参数的上下边界 b_u 和 b_l，初始化种群位置，每个个体代表一组超参数。

(3)超参数优化：种群中每个霜粒子更新位置信息，更新迭代次数 $t = t + 1$。

(4)适应度更新：比较每个霜粒子的更新适应度与更新前的值，记录最优适应度，并记录当前最优粒子。

(5)算法终止：满足算法终止条件，将优化后的超参数 x_{gbest} 训练模型，得到最终模型。

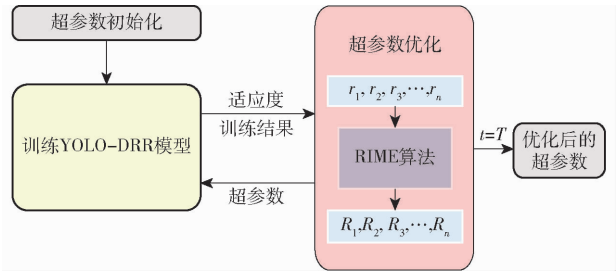

图 7　基于 RIME 算法的 YOLO－DRR 模型超参数优化流程图

Fig. 7　Flowchart of hyperparameter optimization for YOLO－DRR model based on RIME algorithm

3　实验结果分析

3.1　实验环境与评价指标

YOLO－DRR 模型是一个基于深度学习的分割框架，利用 Pytorch 1.13.0 框架和 Python 3.9.18 编程语言，在强大的 Intel Core i9－10900K CPU 和 NVIDIA GeForce RTX 3080 GPU 支持下，实现了高效的深度学习训练。在训练过程中，采用随机梯度下降(Stochastic gradient descent，SGD)优化器，并基于余弦退火策略动态调整学习率。训练共进行 500 次迭代，Batch Size 设置为 16。为有效防止过拟合，引入了早停策略：若模型在连续 100 个迭代中性能未得到改善，则提前终止训练。该配置不仅确保了模型的充分训练，还有效节约了计算资源。

本研究使用的主要评价指标为精确度(P)，IoU 阈值为 0.5 时平均精度均值(mAP@0.5)，浮点运算量(GFLOPs)，模型内存占用量(Size)，召回率(R)，在多个 IoU 阈值(从 0.5 到 0.95，步长为 0.05)下计算的平均精度均值(mAP@0.5:0.95)。

3.2　消融实验

为进一步分析不同改进策略的性能，基于相同的训练集、训练方法和测试环境，依次对 YOLO－DRR 模型的改进模块进行消融实验。其中，以 YOLO v5－seg 模型为基准模型，分别在模型中逐步添加分布移位卷积模块(DSConv)、多尺度特征提取模块(RFEM)，形成多个改进模型，实验结果如表 1 所示。

表 1　消融实验结果

Tab. 1　Results of ablation experiment

模型	P/%	mAP@0.5/%	mAP@0.5:0.95/%	浮点运算量
YOLO v5－seg	84.63	89.70	61.08	2.57×10^{10}
添加 DSConv	84.79	90.65	62.80	2.61×10^{10}
添加 RFEM	84.88	91.61	62.88	2.57×10^{10}
YOLO－DRR	85.51	92.86	63.03	2.61×10^{10}

由表 1 数据可以得到,各种改进策略均对模型的训练结果产生了正面作用。其中,添加了 DSConv 模块之后,模型分割的精确度、mAP@0.5 和 mAP@0.5:0.95 相较于基准模型分别提高 0.19%、1.06% 和 2.82%,说明 DSConv 模块能够保持模型的精度。添加了 RFEM 模块后,模型分割的精确度、mAP@0.5 和 mAP@0.5:0.95 与基准模型相比分别提升 0.30%、2.13% 和 2.95%,说明 RFEM 模块捕捉到输入图像中的远程依赖关系和上下文信息,通过融合多尺度特征,RFEM 能够更有效地处理不同尺度的目标,提高了模型对复杂场景的适应能力和分割能力。最后,在综合以上改进策略后,YOLO-DRR 模型的分割精确度、mAP@0.5 和 mAP@0.5:0.95 均得到显著提升,较基准模型分别提升 1.04%、3.52%、3.19%。

3.3 热力图分析

为深入探讨 YOLO-DRR 模型在复杂场景下的特征提取优势,利用 Grad-CAM[29] 技术,结合已完成训练的模型权重,生成了热力图。YOLO v5s-seg 模型和 YOLO-DRR 模型的分割效果和热力图结果如图 8、9 所示。

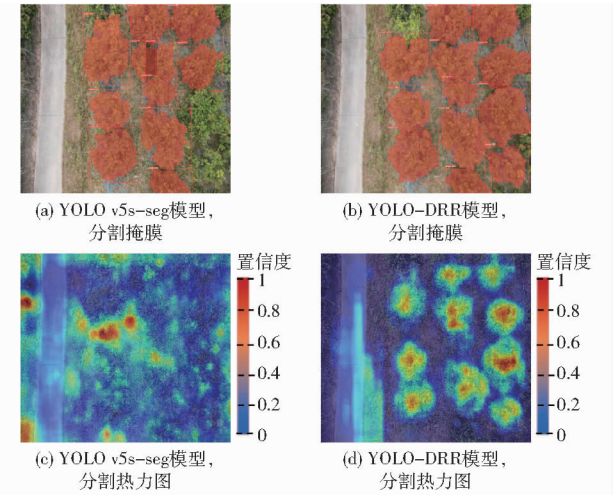

图 8 YOLO v5s-seg 模型和 YOLO-DRR 模型对稀疏目标图像进行分割掩膜和热力图

Fig. 8 YOLO v5s-seg model and YOLO-DRR model perform segmentation and heat map detection on sparse target images

对比图 8a、8b 和图 9a、9b 可以看出,YOLO v5s-seg 模型在面对稀疏或密集分布的树冠时,尽管展现了一定的分割能力,但不难发现其存在明显的漏检现象,部分树冠分割结果不完整,直接影响了整体分割的准确性和实用性。相比之下,YOLO-DRR 模型通过集成 DSConv 模块与 RFEM 模块,显著增强了模型对图像细节的捕捉能力和特征表达能力。这些区域精准对应了树冠的实际位置,不仅验证了 YOLO-DRR 模型在特征提取方面的优势,也展示了其在复杂环境下对目标对象的强大分割能力。

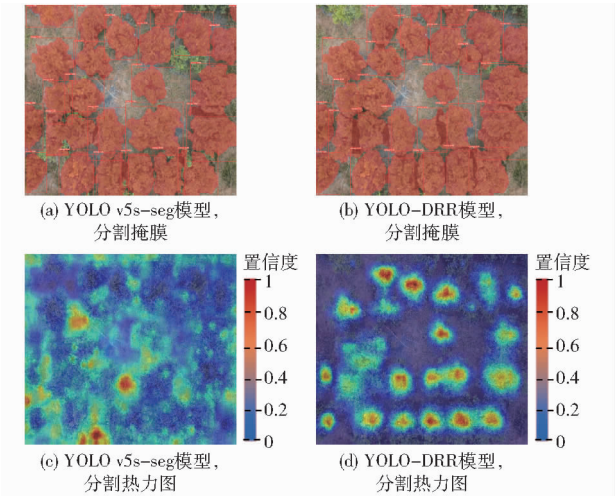

图 9 YOLO v5s-seg 模型和 YOLO-DRR 模型对密集目标图像进行分割掩膜和热力图

Fig. 9 YOLO v5s-seg model and YOLO-DRR model perform segmentation and heat map detection on dense target images

在热力图的呈现中,数据点或区域的颜色值直接映射了它们所代表的数据密集程度或具体数值的高低。具体来说,颜色从冷色调(比如蓝色)渐变至暖色调(比如红色),用以表达从低到高的数据密集性或数值大小,这样能够迅速识别数据分布的热点(高值区域)与冷点(低值区域)。颜色标尺表示模型预测的置信度分数(从 0 到 1),其中 0 表示背景,1 表示对“冠层”类别的最高置信度,中间颜色梯度对应不同程度的预测确信度。进一步分析图 8c、8d 和图 9c、9d,可以观察到 YOLO-DRR 模型在处理不同密度和形态的树冠时,均能保持较高的稳定性和一致性。

3.4 与其他模型的对比分析

将改进后的 YOLO-DRR 模型与其他主流或较新的分割模型进行对比,包括:YOLACT[30]、YOLO v5n-seg、YOLO v8n-seg、YOLO v8s-seg[31]、YOLO v9[32],结果如表 2 所示。

表 2 柑橘果树冠层数据集上其他模型的性能对比

Tab. 2 Performance comparison of other models on dataset of citrus tree canopy structure

模型	mAP@0.5/%	mAP@0.5:0.95/%	模型内存占用量/MB
YOLACT	88.10	51.71	189
YOLO v5n-seg	89.75	60.86	4.1
YOLO v8n-seg	91.23	62.36	6.8
YOLO v8s-seg	90.85	60.24	23.9
YOLO v9	91.25	64.06	55.7
YOLO-DRR	92.86	63.03	26.0

与其他模型相比，YOLO－DRR 模型 mAP@0.5、mAP@0.5:0.95 和模型内存占用量都有显著优势。YOLACT 是经典的单阶段、实时检测、实例分割方法，在 YOLO v5 和 YOLO v8 中的实例分割中，也基于 YOLACT 实现，但对比本研究中提出的 YOLO－DRR 模型，其 mAP 分别低了 4.76 个百分点和 11.32 个百分点，并且模型内存占用量是 YOLO－DRR 的 7.27 倍。YOLO v5n－seg 模型的模型内存占用量虽然约为 YOLO－DRR 模型的 1/6，但是其 mAP 比 YOLO－DRR 模型分别低 3.11 个百分点和 2.18 个百分点。YOLO v8n－seg 的模型内存占用量虽然约为 YOLO－DRR 模型的 1/4，但同样它的 mAP 比 YOLO－DRR 模型分别低 1.63 个百分点和 0.67 个百分点。YOLO v8s－seg 模型虽然模型内存占用量略少于 YOLO－DRR 模型，但是其 mAP 比 YOLO－DRR 模型分别低 2.01 个百分点和 2.79 个百分点。YOLO v9 模型是目前 YOLO 系列最新的分割模型，根据表 2 数据可以看出，它的 mAP@0.5:0.95 高于 YOLO－DRR 模型 1.03 个百分点，但 mAP@0.5 低 1.61 个百分点，且 YOLO v9 的模型内存占用量是 YOLO－DRR 的 2.14 倍，并不利于部署到边缘平台中。

综上所述，本研究中提出的 YOLO－DRR 模型的 mAP@0.5、mAP@0.5:0.95 和模型内存占用量综合表现最好，更适合在自然环境下、复杂背景中进行柑橘果树冠层分割任务。另外，因为模型较小，具有部署到 FPGA 边缘平台的优势。

3.5　基于 RIME 算法的模型超参数优化

以 YOLO－DRR 模型的 28 个超参数组合为优化对象，利用 RIME 控制参数少、寻优速率快等特点，将超参数映射为 RIME 算法的个体可行解，基于群智能寻优迭代机制获取适合柑橘果树冠层分割任务的最优超参数组合。控制参数包括周期为 50，迭代次数为 100，种群数量为 10。优化前后分割的结果对比如表 3 所示。从表 3 可以看出，超参数优化前后，模型的精确度、召回率、mAP@0.5 和 mAP@0.5:0.95 分别提升 0.83、0.50、0.55、0.10 个百分点，说明超参数进行优化可以有效提升柑橘果树冠层分割精度，进一步完善模型的性能。

表 3　超参数优化前后分割结果对比

Tab.3　Comparison of segmentation results before and after hyperparameter optimization　%

处理	P	R	mAP@0.5	mAP@0.5:0.95
优化前	85.51	88.18	92.86	63.03
优化后	86.34	88.68	93.41	63.13

3.6　不同优化算法对 YOLO－DRR 模型的超参数优化分析

选取 PSO 算法、GA 算法和 BWOA 算法作为 RIME 算法的基准对比算法，针对 YOLO－DRR＋PSO、YOLO－DRR＋GA、YOLO－DRR＋BWOA 和 YOLO－DRR＋RIME 4 种优化模型进行系统性对比与评估，分析各种群智能优化算法针对超参数优化任务的优劣性。其中控制参数包括周期为 50，迭代次数为 100，种群数量为 10。

由表 4 可知，3 种传统优化算法的精确率 P、召回率 R、mAP 值都有所降低。实验结果表明，PSO 算法在对于模型超参数优化的任务中，容易使得模型超参数组合优化陷入局部解，且结果缺乏精确性，无法确定一组接近最优解的超参数组合，难以进一步提高模型的性能；GA 算法由于控制变量多，存在与 PSO 算法一样的问题，局部逃逸能力较弱，无法准确获取接近全局最优的超参数组合；BWOA 算法在一定程度上优于 PSO 和 GA 算法，但还是因为存在收敛速度可能较慢、对高维复杂问题的全局搜索能力有限等问题，无法准确获取接近全局最优的超参数组合；RIME 算法能跳出局部最优，通过并行搜索策略有效探索高维空间，可以确定一组接近最优解的超参数组合，有效提升柑橘冠层检测分割精度，进一步完善模型的性能。

表 4　不同优化算法优化前后分割结果对比

Tab.4　Comparison of segmentation results before and after optimization by different optimization algorithms　%

处理	模型	P	R	mAP@0.5	mAP@0.5:0.95
优化前	YOLO－DRR	85.51	88.18	92.86	63.03
优化后	YOLO－DRR＋PSO	84.65	84.65	91.18	57.42
	YOLO－DRR＋GA	82.44	87.51	90.41	60.54
	YOLO－DRR＋BWOA	84.78	86.88	92.88	60.47
	YOLO－DRR＋RIME	86.34	88.68	93.41	63.13

4　基于边缘计算平台的柑橘果树冠层分割

考虑到柑橘果园多位于山地丘陵地区，果园地形复杂，果树之间种植密集，人工无法便捷及时地了解到果树冠层信息，而 FPGA 作为边缘计算平

台,区别于服务器需要部署在数据中心或机房中,需要强大的计算能力和存储容量,FPGA 适合用于需要实时反馈的应用场景,例如地形崎岖的山地果园,同时还能够显著减少数据传输的延迟和成本,实现数据的实时采集和处理[33]。因此,本研究尝试将改进后的模型移植部署到轻便的 FPGA 边缘计算平台,在复杂果园环境中实现实时柑橘果树冠层分割。

本研究采用的 FPGA 边缘计算平台是由 Xilinx 公司生产的 UltraScale + MPSoC ZCU104 计算平台,该平台集成了高性能的多处理器系统,能够同时处理复杂的应用程序和实时控制任务,满足高性能计算和嵌入式系统的双重需求,还包含深度学习处理器单元(Deep learning processor unit, DPU),能够通过可重新配置的硬件架构支持不同的神经网络结构。

4.1 模型移植部署

改进后的模型移植部署流程如图 10 所示,主要步骤包括模型量化编译和程序处理设计,概述如下:

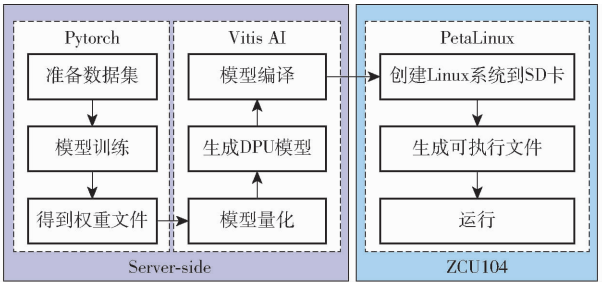

图 10 移植部署流程图

Fig. 10 Transplant deployment flowchart

(1)模型量化编译需要在服务器平台 Linux 系统中,使用 Xilinx Vitis AI 工具来进行。通常,进行神经网络训练时,使用的是 32 位浮点权重和激活值,而通过 Vitis AI 量化后,可以将 32 位浮点权重和激活值转换为 8 位整数格式,使得网络模型的部署所需存储器宽带更少,相比于部署在服务器端的网络模型,速度更快且能效更高。首先,将柑橘果树冠层训练集作为校准数据集,在 Linux 系统下启动 Xilinx Vitis AI Docker 后,使用 ZCU104 运算平台相对应量化器工具对模型进行定点量化,生成编译用的 DPU 模型;其次,选用 ZCU104 运算平台对应 DPU 型号的编译器工具对生成的文件进行编译,得到可以移植进 FPGA 边缘计算平台的 Xmodel 文件;最后使用 SD 卡将该文件移植到 ZCU104 运算平台中。

(2)在程序设计处理部分,使用 PetaLinux 工具构建了 Linux 操作系统。编写 Python 程序,在 ARM 端使用 OpenCV 库、DPU 的库函数和 API 等对柑橘果树冠层进行预处理,并调用 DPU 实现对柑橘果树冠层的实时分割功能。

4.2 模型移植测试结果分析

为测试模型移植效果,设计了基于 ZCU104 边缘计算平台的测试方案:首先,设置模型输入图像尺寸为 640 像素 × 640 像素,实时采集视频流分辨率为 640 像素 × 480 像素;其次,将 ZCU104 边缘计算平台分别通过 USB 接口和 HDMI 接口连接摄像头和显示器,实现实时采集柑橘果树冠层画面,并展示经过计算平台处理后的冠层分割画面;最后,使用功率计量插座来测量边缘计算平台的功耗。所有性能指标均在 FPGA 芯片达到热稳态后,通过连续处理视频流数据测得,以反映真实的部署性能。

如表 5 所示,量化移植后的模型相比移植前的模型小了 16.29 MB,监测分割速度达到 19 f/s,功耗仅为 22 W。因此,基于 ZCU104 边缘计算平台的柑橘果树冠层分割技术方案能够解决柑橘果树冠层信息监测的便携性和实时性问题。

表 5 使用 FPGA 边缘计算平台对柑橘果树冠层分割性能

Tab. 5 Analysis of segmentation performance of citrus tree canopy using FPGA edge computing platform

测试平台	模型内存占用量/MB	分辨率/(像素 × 像素)	平均帧率/($f \cdot s^{-1}$)	功耗/W
服务器	26	640 × 480	26	164
FPGA	9.71	640 × 480	19	22

对比 GPU 平台,该场景的刚性约束是低功耗、高可靠性、低成本,而非追求极致的绝对吞吐量。首先,22 W 的功耗远低于同等算力的 GPU 平台,这对于需要长时间连续运行且对电费敏感的农业现场至关重要。其次,FPGA 可通过定制化减少外围电路,最终硬件成本可控,而 GPU 平台通常是一个完整的计算模块,成本较高。最后,FPGA 的硬件并行流水线架构提供了极低的、确定的延迟,这是 GPU 无法保证的。因此,GPU 平台固然能提供更高的帧率,但这并非该目标场景的首要需求。

综上所述,本研究将经过模型结构改进和霜冰优化算法优化超参数的 YOLO - DRR 模型移植部署到 FPGA 边缘计算平台中,实现了对柑橘果树冠层实时分割监测。本研究中的便携式柑橘果树冠层分割设备设计小巧,便于携带至复杂地形的山地果园,能显著提高现场作业的灵活性和效率,该设备还能实时对柑橘果树冠层进行分割,确保数据收集的即时性和准确性,为果园提供及时反馈,另外低功耗的设计使该设备在长时间果园作业中更加可靠,大大减少了能源消耗。但由于移植过程中需要使用

DPU 进行神经网络加速，DPU 只支持 8 位定点计算，通过 Xilinx Vitis AI 的量化器工具将 32 位浮点运算转换为 8 位定点运算时，不可避免地会产生少量精确度损失。随着农业智能化趋势的加速发展，本研究中的便携式柑橘果树冠层分割设备展现了广阔的应用前景，不仅能显著提升果园管理的智能化水平，还能减少人工干预，降低劳动成本，提高生产效率。未来，随着技术的不断进步和模型优化的深入，后续将进一步减少精确度损失，提升分割精度和速度，同时仍保持轻便、低功耗的优势。

5　结论

（1）本研究针对种植密集复杂背景下、冠层间遮挡重叠严重难以清晰快速得到冠层生长信息等问题，提出了一种自然环境下分割柑橘果树冠层的 YOLO－DRR 模型。该模型计算量较小，能以最小移植损失部署到 FPGA 边缘计算平台中。

（2）通过在 YOLO v5s－seg 模型的骨干网络中 SPFF 模块后添加 RFEM 多尺度特征提取模块，提高了模型对不同尺度目标分割的精确度。将原 YOLO v5s－seg 模型颈部网络中的 C3 模块替换为 DSConv 分布移位卷积模块，降低了卷积核中的内存占用量，提高了运算速度，可更好地移植进边缘计算平台中。通过使用 RIME 算法对 YOLO－DRR 模型超参数进行优化，进一步提升模型的分割精度。优化后的模型分割精确度、召回率、mAP@0.5 和 mAP@0.5：0.95 分别为 86.34%、88.68%、93.41% 和 63.13%。

（3）将改进的模型移植至边缘计算平台中，移植后模型内存占用量为 9.71 MB，检测速度达到 19 f/s，功耗仅为 22 W，实现低能耗、易携带等特点，满足柑橘果园对果树冠层实时分割的要求。

（4）本研究结果表明，所提出的 YOLO－DRR 模型在自然复杂环境下，对柑橘果树冠层进行分割具有良好的性能，且具备在边缘计算平台部署的能力，因此能够优化柑橘果园的精细化管理，为推动智慧农业的实践提供强有力的技术支持。

参　考　文　献

[1] BAO Z, LI W, CHEN J, et al. Predicting and visualizing citrus color transformation using a deep mask-guided generative network[J]. Plant Phenomics, 2023, 5: 0057.

[2] SITU W, WEI B, JI Y, et al. Calculation method of fruit tree canopy parameters based on fusion of LiDAR and Kinect[J]. Automation & Information Engineering, 2023, 44(3): 9－16.

[3] SUMNALL M, CARTER D, ALBAUGH T, et al. Evaluating the influence of row orientation and crown morphology on growth of *pinus taeda* L. with drone-based airborne laser scanning[J]. Plant Phenomics, 2024, 6: 0264.

[4] KRAJEWSKI A, SCHUMANN A, EBERT T, et al. Management of citrus tree canopies for fresh-fruit production[J]. EDIS, 2021(1): 1－8.

[5] 刘杰，付兴兰，李旭，等. 丘陵山地柑橘果园机器人自主导航与精确控制系统设计与试验[J]. 西南大学学报(自然科学版)，2024，46(8)：184－197.
LIU Jie, FU Xinglan, LI Xu, et al. Design and test of autonomous navigation and precise control system for robots used in citrus orchards in hilly and mountainous area[J]. Journal of Southwest University (Natural Science Edition), 2024, 46(8): 184－197. (in Chinese)

[6] LU X, SHEN Y, XIE J, et al. Phenotyping of panicle number and shape in rice breeding materials based on unmanned aerial vehicle imagery[J]. Plant Phenomics, 2024, 6: 0265.

[7] ZHANG P, SUN X, ZHANG D, et al. Lightweight deep learning models for high-precision rice seedling segmentation from UAV-based multispectral images[J]. Plant Phenomics, 2023, 5: 0123.

[8] AMPATIZIDIS Y, PARTEL V. UAV-based high throughput phenotyping in citrus utilizing multispectral imaging and artificial intelligence[J]. Remote Sensing, 2019, 11(4): 410.

[9] OSCO L, NOGUEIRA K, RAMOS A, et al. Semantic segmentation of citrus-orchard using deep neural networks and multispectral UAV-based imagery[J]. Precision Agriculture, 2021, 22(4): 1171－1188.

[10] YANG B, XU Y. Applications of deep-learning approaches in horticultural research: a review[J]. Horticulture Research, 2021, 8(1): 123.

[11] OTSU N. A threshold selection method from gray-level histograms[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1979, 11(285－286): 23－27.

[12] ROERDINK J, MEIJSTER A. The watershed transform: definitions, algorithms and parallelization strategies[J]. IOS Press, 2000, 41(1－2): 187－228.

[13] BOYKOV Y, JOLLY M. Interactive graph cuts for optimal boundary & region segmentation of objects in n－d images[J]. Proceedings Eighth IEEE International Conference on Computer Vision, 2001, 1: 105－112.

[14] 李贞，刘海燕，刘策，等. 基于机器学习的 K-means 聚类优化算法研究[J]. 数据挖掘，2022，12(1)：20－26.
LI Zhen, LIU Haiyan, LIU Ce, et al. Research on K-means clustering optimization algorithm based on machine learning[J]. Hans Journal of Data Mining, 2022, 12(1): 20－26. (in Chinese)

[15] WANG F, HUANG Y, HUANG Z, et al. MRUNet: a two-stage segmentation model for small insect targets in complex environments[J]. Journal of Integrative Agriculture, 2023,22(4):1117 - 1130.
[16] DAI W, ZHU W, ZHOU G, et al. AISOA - SSformer: an effective image segmentation method for rice leaf disease based on the transformer architecture[J]. Plant Phenomics, 2024,6:0218.
[17] MOLNÁR S, KERESZTES B, TAMÁS L. Feature pyramid network based proximal vine canopy segmentation[J]. IFAC - PapersOnLine, 2023,56(2):8920 - 8925.
[18] BARROS T, CONDE P, GONCALVES G, et al. A deep learning comparison study[J]. Computers and Electronics in Agriculture, 2022,195:106782.
[19] SUN Q, ZHANG R, CHEN L, et al. Semantic segmentation and path planning for orchards based on UAV images[J]. Computers and Electronics in Agriculture, 2022,200:107222.
[20] ZOU K, LIAO Q, ZHANG F, et al. A segmentation network for smart weed management in wheat fields[J]. Computers and Electronics in Agriculture, 2022,202:107303.
[21] WU W, YANG T, RUI L, et al. Detection and enumeration of wheat grains based on a deep learning method under various scenarios and scales[J]. Journal of Integrative Agriculture, 2020,19(8):1998 - 2008.
[22] LYU S, ZHOU X, LI Z, et al. YOLO - SCL: a lightweight detection model for citrus psyllid based on spatial channel interaction[J]. Frontiers in Plant Science, 2023,14:1276833.
[23] REBIAI M, BENGAHARBIA B, MAAZOUZ M, et al. FPGA implementation of an improved envelope detection approach for bearing fault diagnosis[J]. Journal of Engineering and Applied Science, 2024,71(1):3.
[24] YACOUB M, ISMAIL S, SAID L, MADIAN A, et al. Support vector machine reconfigurable hardware implementation on FPGA[J]. Franklin Open, 2024,7:100115.
[25] ELSAID K, EL-KHARASHI M, SAFAR M. An optimized FPGA architecture for machine learning applications[J]. AEU-International Journal of Electronics and Communications, 2024,174:155011.
[26] YU Z, HUANG H, CHEN W, et al. YOLO - FaceV2: a scale and occlusion aware face detector[J]. Pattern Recognition, 2024,115:110714.
[27] NASCIMENTO M, FAWCETT R, PRISACARIU V. DSConv: efficient convolution operator[C]// 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019:5147 - 5156.
[28] SU H, ZHAO D, HEIDARI A, et al. RIME: a physics-based optimization[J]. Neurocomputing, 2023,532:183 - 214.
[29] SELVARAJU R, COGSWELL M, DAS A, et al. Grad - CAM: visual explanations from deep networks via gradient-based localization[C]// Proceedings of the IEEE International Conference on Computer Vision, 2017:618 - 626.
[30] BOLYA D, ZHOU C, XIAO F, et al. YOLACT: real-time instance segmentation[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019:9157 - 9166.
[31] VARGHESE R, SAMBATH M. M. YOLOv8: a novel object detection algorithm with enhanced performance and robustness [C]//2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), 2024: 1 - 6.
[32] WANG C, YEH I, LIAO H. YOLOv9: learning what you want to learn using programmable gradient information[C]// European Conference on Computer Vision, 2024:1 - 21.
[33] YACOUB M, ISMAIL S, SAID L, et al. Reconfigurable hardware implementation of K-nearest neighbor algorithm on FPGA [J]. AEU—International Journal of Electronics and Communications, 2024,173:154999.