

基于改进 YOLO v8 的泰山赤鳞鱼轻量化计数模型

全志民¹ 李常浩¹ 李修松^{2,3} 周宇¹ 赵瑶⁴ 孟晓军^{1,3} 荣丽红¹

(1. 青岛农业大学机电工程学院, 青岛 266109; 2. 青岛大牧人机械股份有限公司, 青岛 266000;

3. 农业农村部智能畜禽养殖设施装备重点实验室, 青岛 266000;

4. 青岛农业大学海都学院机电与建筑工程学院, 莱阳 265200)

摘要: 泰山赤鳞鱼是一种泰山区域特有、具有较高经济价值的高端食用鱼种,在渔获季节和日常养殖中仍以人工计数为主。为实现养殖和捕捞过程的精准识别计数,提出了一种基于 YOLO v8 的泰山赤鳞鱼轻量化计数模型 ACSS-YOLO v8。首先,采用 AIFIC 模块加强模型对图像高级特征的处理能力,同时降低模型参数量和计算量。其次,在 AIFIC 模块后加入 CAFM,通过全局特征和局部特征融合,提升模型的检测性能。然后,基于 Star Blocks 结构设计新的颈部,增强模型表达能力、降低计算复杂度。最后,引入 SEAM 提高模型识别遮挡目标的能力,并进一步降低计算冗余和内存消耗。实验结果表明,本文设计的 ACSS-YOLO v8 模型的精确率、召回率和 mAP₅₀ 分别达到了 92.6%、85.8% 和 89.7%,相较于 YOLO v8s 模型,其参数量、浮点运算量和内存占用量分别降低了 20%、25% 和 20%。计数模块和 Jetson Orin Nano Super 部署实验表明,可视化的计数模块能够直观显示目标数量,图像检测速度达到 30 f/s,能够满足复杂现场环境的检测需求。

关键词: 泰山赤鳞鱼; 计数模型; YOLO v8; 轻量化; 边缘计算

中图分类号: TP391.41

文献标识码: A

文章编号: 1000-1298(2026)15-0148-09

OSID:



Lightweight Counting Model for Taishan Red-scaled Fish
Based on Improved YOLO v8

Tong Zhimin¹ Li Changhao¹ Li Xiusong^{2,3} Zhou Yu¹ Zhao Yao⁴ Meng Xiaojun^{1,3} Rong Lihong¹

(1. College of Mechanical and Electrical Engineering, Qingdao Agricultural University, Qingdao 266109, China

2. Qingdao Big Herdsman Machinery Co., Ltd., Qingdao 266000, China

3. Key Laboratory of Intelligent Livestock and Poultry Breeding Facilities and Equipment,
Ministry of Agriculture and Rural Affairs, Qingdao 266000, China

4. School of Mechanical and Electrical Engineering and Construction Engineering,
Haidu College Qingdao Agricultural University, Laiyang 265200, China)

Abstract: The Taishan red-scaled fish (*Varicorhinus macrolepis*) is a kind of high-end edible fish species with high economic value unique to Taishan area. It is still mainly counted manually in the fishing season and daily breeding. In order to realize the accurate identification and counting of the breeding and fishing process, a lightweight identification model ACSS-YOLO v8 was proposed for Taishan red-scaled fish based on YOLO v8. Specifically, multiple structural optimizations were integrated to balance detection accuracy and computational efficiency simultaneously. Firstly, the AIFIC module was used to enhance the model's ability to process high-level image features, while reducing parameter volume and calculations. Secondly, CAFM was added after the AIFIC module. By fusing global and local features, the detection performance of the model was enhanced. Then a neck network was designed based on the star blocks structure. This design enhanced the feature expression ability of the model and further reduced the computational complexity. Finally, SEAM was introduced to improve the model's ability to recognize occluded objects, and further reduce computational redundancy and memory consumption. The experimental results showed that the P, R and mAP₅₀ of the ACSS-YOLO v8 model designed reached 92.6%, 85.8% and 89.7%, respectively. Compared with the YOLO v8s model, the number of

收稿日期: 2026-03-17 修回日期: 2026-03-25

基金项目: 泰山产业领军人才工程专项经费资助项目(TSCX 202507039)

作者简介: 全志民(1977—),男,副教授,博士,主要从事农业信息感知与智慧畜牧养殖装备研究,E-mail: leicahit@qau.edu.cn

通信作者: 李修松(1978—),男,教授级高级工程师,博士,主要从事畜禽养殖装备研究,E-mail: lixs@bigherdsman.com

parameters, GFLOPs and memory usage were reduced by 20%, 25% and 20%, respectively. The counting module was developed to provide intuitive visual feedback for practical application scenarios. The experiments on the counting module and Jetson Orin Nano Super deployment showed that the visualized counting module can intuitively display the number of objects, and the image inference speed reached 30 f/s, which can meet the detection requirements of complex on-site environments.

Key words: Taishan red-scaled fish; counting model; YOLO v8; lightweight; edge computing

0 引言

中国水产产量持续稳居世界首位^[1]。泰山赤鳞鱼 (*Varicorhinus macrolepis*) 属鲤科鱼巴亚科突吻鱼属动物, 分布于泰山海拔 270 ~ 800 m 的泰山溪流中, 其繁殖能力低、生长速度缓慢。其蛋白质含量、矿物质含量、氨基酸种类等都很高, 具有较高的营养价值和药用价值。

泰山赤鳞鱼渔获季节依赖人工计数。近年来, 深度学习技术广泛应用于渔业养殖中, 具有高效完成数量监测与统计的优势^[2]。因此, 研发一种基于深度学习的泰山赤鳞鱼高效识别模型十分必要。

在识别和计数研究中, 精度和轻量化是衡量模型的两个重要的指标。当前主流高精度模型普遍存在计算复杂度高、参数规模大、内存占用多等问题, 例如双阶段的 R - CNN^[3] 系列 (包含 Fast R - CNN^[4]、Faster R - CNN^[5] 和 Mask R - CNN^[6]), 以及单阶段的 SSD^[7]。在轻量化的相关研究进程中, 衍生出了很多的方法, 包括网络剪枝^[8]、知识蒸馏^[9]、模型压缩^[10] 和研究设计轻量化的结构^[11] 等。基于此, 很多轻量化网络涌现出来, 例如 StarNet^[12]、GhostNet^[13]、MobileNetV4^[14] 和 ShuffleNet^[15] 等, 虽然参数量、计算复杂度与内存占用上具有优势, 但在多数实际任务中仍存在精度不足的局限。为此, 研究者围绕精度提升与模型优化展开了深入探索。Zhang 等^[16] 通过将 YOLO v5 - Nano 和改进 SORT 算法等方法结合, 提出了一种动态鱼苗计数方法, 可以准确计数大量鱼苗, 并降低了鱼苗遮挡的影响。尹莉娜等^[17] 提出了一种基于密度图回归的 U - Fish 鱼群计数方法, 实现了对鱼群的精准计数。张璐等^[18] 提出一种基于多尺度融合与无锚点 YOLO v3 的鱼群计数方法, 该方法在面对复杂背景及多尺度鱼群图像时, 展现出更优的特征提取性能。Wang 等^[19] 提出了一种端到端神经网络方法, 在高密度养殖条件下, 可以实现对鱼类个体精准高速的跟踪。郑荣才等^[20] 提出了基于 YOLO v7 模型改进的轻量化鲑鱼检测模型 YOLO v7 - CSMRep, 提高了网络特征提取能力, 解决了深海网箱养殖中鲑鱼相互遮挡导致的识别计数难题。刘晓阳等^[21] 利用无人船、YOLO v10 和 ByteTrack 结合的方法, 实现了对海参

的准确计数和定位。

尽管现有目标识别与密度图回归方法已取得进展, 但在兼顾计算效率与模型轻量化方面仍存在不足, 难以满足实际精准识别的需求。为此, 本文基于 YOLO v8s 模型, 提出一种面向泰山赤鳞鱼的改进轻量化计数模型 ACSS - YOLO v8, 兼顾检测精度与效率, 以期为泰山赤鳞鱼的日常养殖与高效渔获作业提供技术支持。

1 数据采集与数据集制作

1.1 数据来源

由于泰山赤鳞鱼的养殖条件苛刻, 养殖户渔场大多位于海拔较高的山区, 所以本实验所使用的数据集取自山东省泰安市泰山区大津口乡 (36. 30°N, 117. 13°E) 的某养殖户渔场。数据集于 2024 年 4 月和 5 月间采集, 在随机天气条件下, 分别于 09:00、12:00 和 15:00 进行拍摄。拍摄工具选用佳能 R10 型微单相机, 搭配变焦镜头 RF24 - 105mm f4 - f7. 1。拍摄期间严格遵循养殖人员的指导, 首先将泰山赤鳞鱼捕捞到水产养殖盆中, 然后在水产养殖盆正上方 1 m 处以俯视角度进行拍摄, 获取数据集图像格式为 jpg, 数据集图像分辨率为 6 000 像素 × 4 000 像素。

1.2 数据集制作

本研究共采集到约 1 300 幅图像, 然后人工筛选和处理, 最终得到 831 幅原始图像, 并按照 8:2 的比例将其随机划分为训练集和验证集。为了增强模型的泛化能力, 本文对采集到的数据集进行数据增强处理, 包括镜像、裁剪、翻转、缩放、颜色、亮度变换和高斯噪声等混合增强, 最后获得了 1 824 幅增强后的训练集图像, 数据增强后的部分图像效果如图 1 所示。利用 Labellmg 标注软件对数据集进行标注, 标注的标签为“fish”, 并生成标注文件。图 1 是数据增强后的图像样例, 图 1a 为原图, 图 1b ~ 1f 为混合增强的图像。

2 泰山赤鳞鱼识别方法

2.1 YOLO v8 检测模型

YOLO v8 继承了 YOLO 系列的优点和改进, 其主要结构为主干网络 (backbone)、颈部网络 (neck)

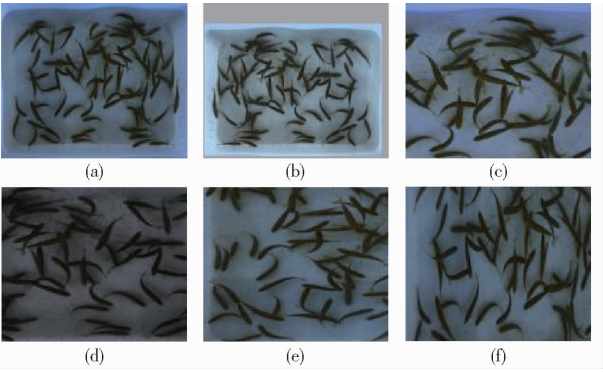


图1 数据增强图像

Fig.1 Data augmentation images

和检测头(head)。其创新主要在主干网络和头部网络,引入了新的功能和改进,通过对特征网络的改进、多尺度特征融合技术的引入和无锚分裂头的使用等方法,进一步提升了模型性能和灵活性。YOLO v8有5个版本,分别为YOLO v8n、YOLO v8s、YOLO v8m、YOLO v8l和YOLO v8x,可以适用于不同的检测需求以及不同应用环境下的部署。由于YOLO v8n精度较低,其模型参数量难以匹配本研究的任务复杂度,因此选择了精度较高且相对轻量化的YOLO v8s,无论在检测精度、检测速度,还是模型规模方面,其综合性能都优于YOLO v8系列中其他模型。因此本文选择YOLO v8s作为基础模型进行改进。

2.2 ACSS-YOLO v8 模型建立

本文提出了一种改进的轻量化计数模型ACSS-YOLO v8。首先,在模型的主干网络中引入AIFI^[22],并在其前面加入一个卷积层,形成AIFIC组合模块,以下简称AIFIC模块。AIFIC模块替换了YOLO v8s原先的SPPF层,增强了模型处理高级图

像特征的能力,同时降低了模型的参数量和计算复杂度;其次,在AIFIC模块后面引入CAFM^[23],利用全局特征与局部特征融合,提升了检测性能;然后,将颈部网络中C2f的Bottleneck替换为StarNet^[12]中的Star Blocks,在保持模型精度的同时降低了计算复杂度;最后,将YOLO v8s检测头结构中的第2个卷积层替换为SEAM^[24]模块,增强了模型识别遮挡目标的能力,并进一步降低了计算冗余和内存消耗。ACSS-YOLO v8模型如图2所示。

2.2.1 改进的AIFIC模块

结合注意力机制的尺度特征交互机制(Attention based intra-scale feature interaction,AIFI),由百度公司在RT-DETR(Real-time detection transformer)算法里提出。在AIFI中,多头自注意力机制会对输入进行注意力权重计算,从而更好地捕获多尺度信息,在添加操作(Add)和归一化(Norm)中,添加操作(Add)融合两个输入特征图,归一化(Norm)在添加操作后,利用层归一化对特征进行归一化处理,FFN增强特征表示,并利用非线性变化提取复杂特征。AIFI能高效处理高级图像特征,提升模型的检测性能,在AIFI之前引入一个卷积层,压缩其通道数量,能够有效减小空间位置编码的复杂度和计算量。在面对具有丰富语义信息高级特征的泰山赤鳞鱼图像时,AIFIC模块可以利用其自我注意力进行处理,并在高级特征内进行尺度交互,使其更加专注于高级特征,增强了模型的检测能力。图3是AIFI结构图,图4为AIFIC结构图。

2.2.2 加入卷积与注意力融合模块CAFM

本实验受光照条件变化和水质条件变化等因素影响,导致目标在图像中呈现的颜色变化多样、模

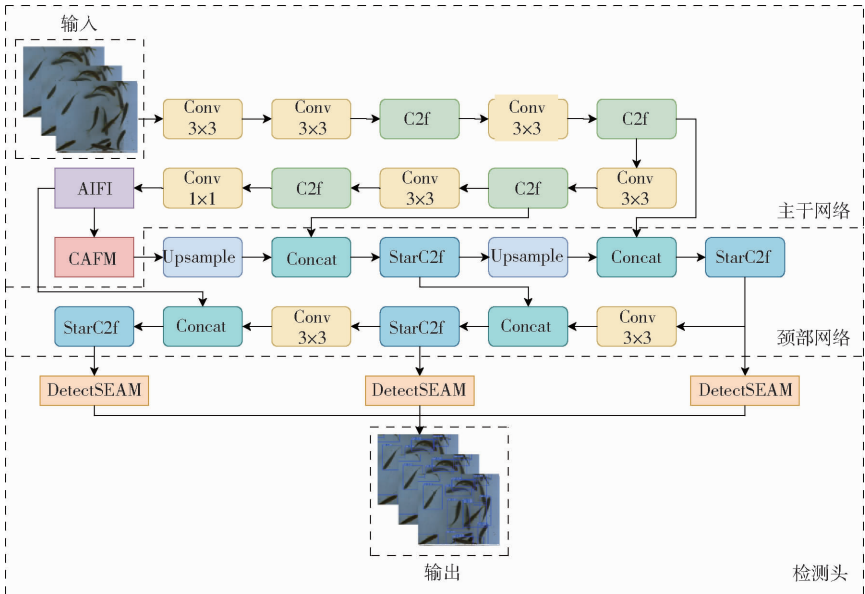


图2 ACSS-YOLO v8 模型结构图

Fig.2 Structure diagram of ACSS-YOLO v8 model

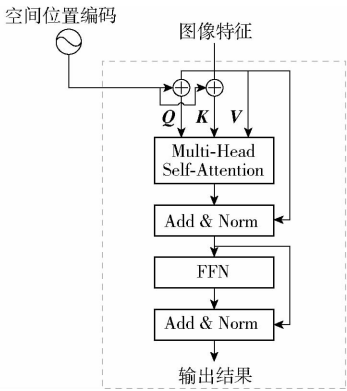


图 3 AIFI 模块结构图

Fig. 3 Structure diagram of AIFI module



图 4 AIFIC 模块结构图

Fig. 4 Structure diagram of AIFIC module

糊。同时,泰山赤鳞鱼的大小不同、姿态各异也使得其特征表示不足。因此本文引入了卷积和注意力融合模块 CAFM (Convolution and attention fusion module)。CAFM 由全局分支和局部分支构成,全局分支依靠注意力机制模拟长距离特征间的依赖关系,局部分支利用卷积和通道重组提取局部特征,从而全面提升模型检测性能^[23]。图 5 是 CAFM 的结构图。

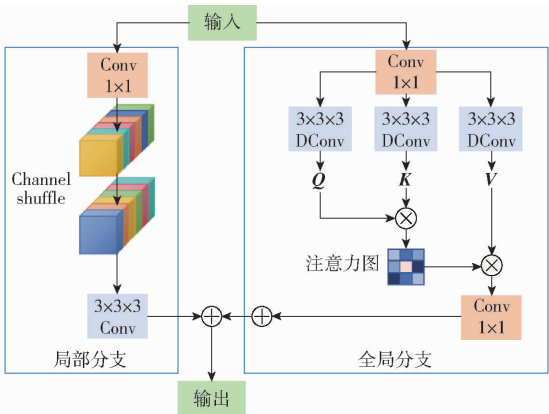


图 5 CAFM 模块结构图

Fig. 5 Structure diagram of CAFM module

在局部分支中,先利用 1×1 卷积来调整通道维度,经通道重组后用 $3 \times 3 \times 3$ 卷积提取特征。局部分支的公式为

$$F_{conv} = W_{3 \times 3 \times 3} (CS(W_{1 \times 1}(Y))) \quad (1)$$

式中 F_{conv} ——局部输出

CS——通道混洗操作运算符

Y ——输入特征图

在全局分支中, 1×1 卷积和 3×3 深度卷积生成 Q 、 K 和 V , Q 表示查询, K 表示键, V 表示值,得到 $H \times W \times C$ 的 3 个张量,然后将 Q 和 K 重塑, Q 和

K 通过相互作用计算注意力图。

全局分支的输出公式为

$$F_{att} = W_{1 \times 1} \text{Attention}(\hat{Q}, \hat{K}, \hat{V}) + Y \quad (2)$$

$$\text{其中 } \text{Attention}(\hat{Q}, \hat{K}, \hat{V}) = \hat{V} \text{Softmax}(\hat{K}\hat{Q}/\alpha) \quad (3)$$

式中 α ——可学习的缩放函数

α 在 Softmax 函数的前面控制 $\hat{K}$ 和 $\hat{Q}$ 的矩阵乘法幅度。

最后,CAFM 的计算输出公式为

$$F_{out} = F_{att} + F_{conv} \quad (4)$$

2.2.3 引入 StarC2f 模块

YOLO v8 中的 C2f 模块采用多瓶颈结构,使梯度消失和爆炸问题得到解决,进而提升了模型性能,但由于 C2f 内部瓶颈结构的堆叠,导致特征图出现冗余并增加计算困难。因此引入 Star Blocks 改进 C2f 模块,该模块以高效的特征图乘法运算为主导,促进特征向更深层次发展,进而提高模型参数效率。本文将 YOLO v8s 中颈部 C2f 的 Bottleneck 结构换为 StarNet 中的 Star Blocks,形成了新的 StarC2f。在 Star Blocks 中,深度可分离卷积操作将标准卷积分解为深度卷积和逐点卷积,提升了模型的计算速度;跳跃连接和 ReLU6 减少了梯度消失和爆炸带来的影响,增强了梯度流动,使模型得到快速收敛,同时跳跃连接稳定了模型训练;为获取更多层次的信息,将不同尺度的卷积核加入到特征提取单元中,同时特征融合单元还能结合不同尺度的特征。在引入 StarC2f 对颈部网络改进后,大大降低了模型的计算复杂度,有效提升了模型对泰山赤鳞鱼的识别速度。图 6 是 StarC2f 结构图。

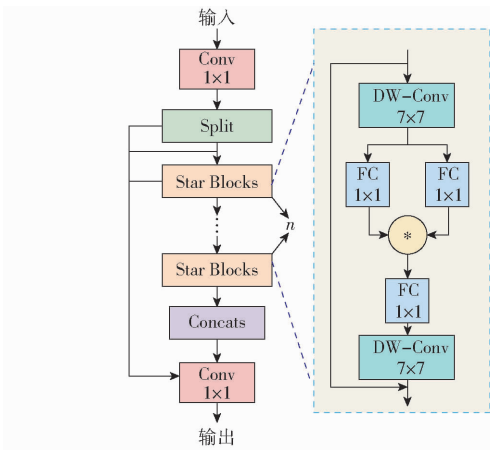


图 6 StarC2f 模块结构图

Fig. 6 Structure diagram of StarC2f module

2.2.4 引入 SEAM 模块

在本研究的大部分数据集中,目标数量较多、大小不一,存在目标之间相互遮挡和重叠的现象,部分目标只露出鱼头或者鱼尾,甚至被完全覆盖。这些目标只有一小部分像素,在常规模型进行检测时可

能会被遗漏。基于此,本研究将 YOLO v8s 检测头中的第 2 层卷积层替换为 SEAM 模块。在 SEAM 模块中,初始信息会进入通道和空间混合模块(CSMM),CSMM 会学习来自通道的信息,在 CSMM 的内部,深度可分离卷积(Depthwise convolution)逐层进行深度处理,然后进行 GELU 激活、批量归一化(Batch norm)、逐点卷积(Pointwise convolution)和批量归一化(Batch norm),上述操作可以增强空间和通道维度的特征交互;在经过 CSMM 内部的一系列操作后,平均池化层压缩特征图的空间维度;然后,通道信息通过两层全连接网络融合,加强通道间的连接性,全连接层的输出经过对数变化后交给指数函数处理,其值从[0,1]扩展到[1,e]。该模块的添加,使模型在满足轻量化的同时,提高了模型对遮挡目标的检测能力。图 7 是 SEAM 结构图,图 8 是 DetectSEAM 结构图。

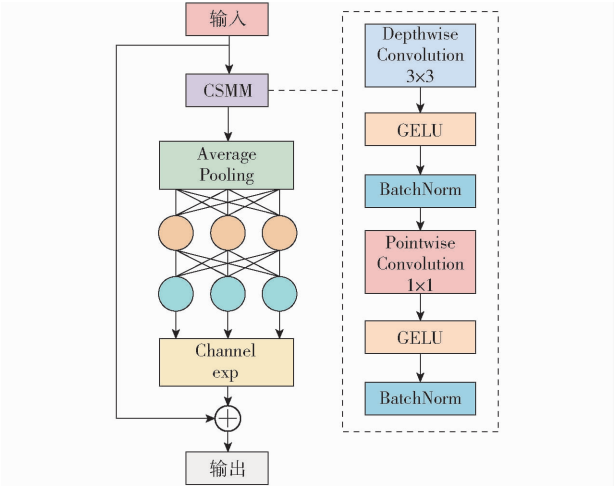


图 7 SEAM 模块结构图

Fig. 7 Structure diagram of SEAM module

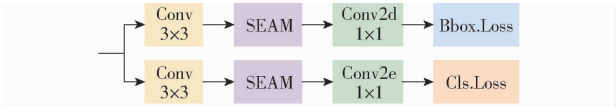


图 8 DetectSEAM 结构图

Fig. 8 Structure diagram of DetectSEAM

2.3 实验条件与评价标准

2.3.1 实验环境与参数设置

本实验所使用的操作系统为 Linux (Ubuntu 20.04),CPU 为 16 vCPU Intel (R) Xeon (R) Gold 6430,GPU 为 RTX 4090 (24 GB),运行内存为 24 GB。软件环境为 Ubuntu 20.04,编程语言为 Python 3.8,深度学习框架为 PyTorch 1.11.0,加速库为 CUDA 11.3,使用 SGD 优化器对模型训练进行优化。在训练时,输入图像尺寸为 640 × 640,迭代次数为 200 次,图像批处理量设为 16,工作线程设为 32。

2.3.2 评价指标

本研究基于实际的应用需求,采用精确率、召回率、平均精度均值 (mAP₅₀)、参数量、浮点运算量和内存占用量作为模型性能评价指标,以此来兼顾精准度和轻量化。

3 实验

3.1 消融实验

为了验证该研究所提出的各个改进模块的性能,本研究以 YOLO v8s 模型为基准模型,基于相同的数据集和相同的实验条件进行了消融实验。最后依据评价指标对不同的消融实验模型进行评估,所得到的实验结果如表 1 所示。

表 1 消融实验结果

Tab. 1 Results of ablation experiment

模型 编号	AIFIC	CAFM	StarC2f	SEAM	精确率/ %	召回率/ %	mAP ₅₀ / %	参数量	浮点运 算量	内存占用 量/MB
1	×	×	×	×	91.9	83.2	86.7	1.113 × 10 ⁷	2.84 × 10 ¹⁰	21.5
2	√	×	×	×	93.4	85.1	90.4	1.057 × 10 ⁷	2.77 × 10 ¹⁰	20.4
3	×	√	×	×	91.5	86.2	88.5	1.234 × 10 ⁷	2.94 × 10 ¹⁰	23.8
4	×	×	√	×	92.4	84.9	88.6	9.850 × 10 ⁶	2.46 × 10 ¹⁰	19.0
5	×	×	×	√	92.7	85.6	89.0	1.065 × 10 ⁷	2.57 × 10 ¹⁰	20.6
6	√	√	×	×	94.3	83.9	90.9	1.068 × 10 ⁷	2.78 × 10 ¹⁰	20.6
7	√	×	√	×	92.3	85.0	89.0	9.300 × 10 ⁶	2.39 × 10 ¹⁰	17.9
8	√	√	√	×	92.2	84.4	89.4	9.410 × 10 ⁶	2.40 × 10 ¹⁰	18.2
9	√	√	√	√	92.6	85.8	89.7	8.930 × 10 ⁶	2.13 × 10 ¹⁰	17.3

注: × 表示不运行此模块;√ 表示运行此模块。

模型 1 为 YOLO v8s,其精确率、召回率、mAP₅₀、参数量、浮点运算量和内存占用量分别为 91.9%、83.2%、86.7%、1.113 × 10⁷、2.84 × 10¹⁰和 21.5 MB,

将其作为后续改进的评估基准。模型 2 引入 AIFIC 模块,AIFIC 能够在高级特征内尺度交互,使精确率、召回率和 mAP₅₀ 指标分别提高了 1.5、1.9、3.7

个百分点,同时降低了计算复杂度,可以从图 9 的可视化特征图验证。模型 3 引入 CAFM 模块,通过全局特征和局部特征融合,从而使召回率与 mAP_{50} 分别提升 3.0、1.8 个百分点,但精确率降低 0.4 个百分点,且计算资源消耗增加。当模型 6 将 CAFM 模块与 AIFIC 模块结合使用时,模型的特征表达和泛化能力得到增强,精确率、召回率和 mAP_{50} 分别提升 2.4、0.7、4.2 个百分点。模型 4 对 StarC2f 的改进大幅降低了计算复杂度,同时精确率、召回率和 mAP_{50} 分别提升 0.5、1.7、1.9 个百分点。模型 7 结合 StarC2f 模块与 AIFIC 模块,二者结合使模型在保持高精度的同时,降低了模型复杂度,精确率、召回率和 mAP_{50} 提升了 0.4、1.8、2.3 个百分点。模型 5 使用了 DetectSEAM 结构,在实现轻量化的同时,精确率、召回率和 mAP_{50} 相较于原模型提升了 0.8、2.4、2.3 个百分点,该模块有效提高了模型抗遮挡能力。模型 8 整合 AIFIC 模块、CAFM 模块及 StarC2f 模块,其精确率、召回率和 mAP_{50} 分别提升 0.3、1.2、2.7 个百分点。模型 9,即 ACSS-YOLO v8 模型,集成了全部模块。在微调后其精确率、召回率和 mAP_{50} 均高于模型 8,且计算资源消耗进一步降低。相较于基准模型,ACSS-YOLO v8 的精确率、召回率和 mAP_{50} 分别提升 0.7、2.6、3.0 个百分点,达到

92.6%、85.8% 和 89.7%,参数量、浮点运算量和内存占用量分别降低 20%、25% 和 20%。实验结果表明,针对因环境影响、目标规模庞大、分布密集和分层导致的遮挡及特征表现不明显等问题,本文提出的 ACSS-YOLO v8 模型展现出最优且稳定的综合性能。

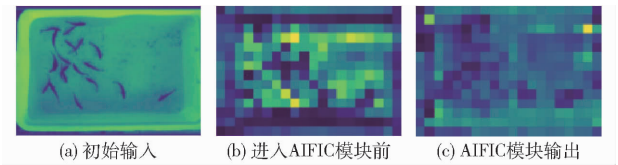


图 9 特征图对比

Fig. 9 Comparison of feature maps

3.2 对比实验

为了进一步验证本文 ACSS-YOLO v8 算法在泰山赤鳞鱼识别计数方面的有效性,本文选取 YOLO v5s、YOLO v7-tiny、YOLO v8s、YOLO v10s 和 YOLO 11s。同时按照轻量化改进策略,选取 GhostNet、MobileNetV4 和 StarNet 的主干对 YOLO v8s 的主干进行替换,形成了 GhostNet-YOLO v8、MobileNetV4-YOLO v8 和 StarNet-YOLO v8。最后,选取轻量化网络 NanoDet 和改进 NanoDet 主干的 NanoDet-RepVGG。整个训练过程均在同样的数据集和同样的实验环境下开展,不同模型的综合对比结果如表 2 所示。

表 2 对比实验结果
Tab. 2 Result of comparative experiment

模型算法	精确率/%	召回率/%	mAP_{50} /%	参数量	浮点运算量	内存占用量/MB
YOLO v5s	91.8	85.2	88.4	9.110×10^6	2.380×10^{10}	17.6
YOLO v7-tiny	92.0	84.1	86.8	6.010×10^6	1.160×10^{10}	12.3
YOLO v8s	91.9	83.2	86.7	1.113×10^7	2.840×10^{10}	21.5
YOLO v10s	91.8	84.5	89.1	7.220×10^6	2.140×10^{10}	15.7
YOLO 11s	91.6	85.4	88.3	9.410×10^6	2.130×10^{10}	18.2
GhostNet-YOLO v8	90.8	81.0	85.4	5.920×10^6	1.610×10^{10}	11.6
MobileNetV4-YOLO v8	77.9	71.5	76.7	1.063×10^7	3.390×10^{10}	20.6
StarNet-YOLO v8	83.0	75.9	80.5	6.540×10^6	1.730×10^{10}	12.7
ACSS-YOLO v8	92.6	85.8	89.7	8.930×10^6	2.130×10^{10}	17.3
NanoDet	-	-	70.1	2.800×10^5	3.900×10^8	1.2
NanoDet-RepVGG	-	-	79.9	7.400×10^6	1.139×10^{10}	28.3

与 YOLO v5s、YOLO v7-tiny、YOLO v8s、YOLO v10s、YOLO 11s、GhostNet-YOLO v8、MobileNetV4-YOLO v8、StarNet-YOLO v8、NanoDet 和 NanoDet-RepVGG 模型相比,本文提出的模型算法在精确率、召回率及 mAP_{50} 指标上优于所有对比模型。具体而言,该模型的精确率较上述模型分别提升了 0.8、0.6、0.7、0.8、1.0、1.8、14.7、9.6 个百分点;召回率分别提升了 0.6、1.7、2.6、1.3、0.4、4.8、14.3、9.9 个百分点; mAP_{50} 则分别提升了 1.3、2.9、3.0、0.6、

1.4、4.3、13.0、9.2、19.6、9.8 百分点。在参数量上,ACSS-YOLO v8 较 YOLO v5s、YOLO v8s、YOLO 11s 和 MobileNetV4-YOLO v8 均有所减少,但与剩余模型相比,其参数量仍较大。在浮点运算量上,改进模型略低于 YOLO v10s,未能优于 YOLO v7-tiny、GhostNet-YOLO v8、StarNet-YOLO v8、NanoDet 和 NanoDet-RepVGG,且与 YOLO 11s 持平。最后,在内存占用量上,ACSS-YOLO v8 小于 YOLO v5s、YOLO v8s、YOLO 11s 和 MobileNetV4-

YOLO v8,但大于其他模型。YOLO v7 - tiny、YOLO v10 和 YOLO 11 是 YOLO 系列的轻量化版本,而 GhostNet 和 StarNet 是近年来表现突出的轻量化网络,因此基于其构建的改进网络也属于轻量化模型。NanoDet 和 NanoDet - RepVGG 虽然在参数量、浮点运算量和内存占用量上远小于改进模型,但其精度过低。尽管本文提出的 ACSS - YOLO v8 模型在部分轻量化指标上不如一些对比模型,存在劣势,但其精确率、召回率和 mAP_{50} 指标均实现了超越。综合考量各维度性能,本文提出的改进模型 ACSS - YOLO v8 在自建泰山赤鳞鱼数据集上相比其他模型展现出更优越的综合性能。

3.3 检测结果对比分析

为了展示改进模型与原模型、部分对比模型的改进效果,选取了 3 幅原始未标记数据集图像,并选取了对应的 YOLO v5s、YOLO v8s、StarNet - YOLO v8、MobileNetV4 - YOLO v8 和 ACSS - YOLO v8 检测的数据集图像。图 10 为不同模型检测结果对比图。由对比模型的检测结果可知,上述模型在目标遮挡和水质干扰等条件下特征提取能力明显不足,从而导致目标漏检(红色边界框标注)和目标被重复检测(黄色边界框标注)等情况的发生,在 YOLO v5s 的第 1 幅检测图中也出现了目标整体未被完全检测的情况(绿色边界框标注)。在 ACSS - YOLO v8 检测图中,改进模型 ACSS - YOLO v8 成功识别了在水质模糊和遮挡等因素影响下,上述模型漏检的目标。由此可得,其 AIFIC 模块实现了对高级特征的交互,CAFM 模块提升了模型在复杂环境中的泛化能力,而 SEAM 模块则提高了模型检测遮挡目标的能力,ACSS - YOLO v8 模型在复杂场景下的鲁棒性得到提升,体现出 ACSS - YOLO v8 较 YOLO v8s 强大的检测能力。

3.4 泰山赤鳞鱼图像识别热力图分析

本文采用 HiResCAM 技术进行视觉评估。如图 11 所示,图 11a ~ 11c 是原始图像,图 11d ~ 11f 是 YOLO v8s 的热力图,图 11g ~ 11i 是 ACSS - YOLO v8 的热力图。根据图 11d ~ 11i 的对比可以看出,YOLO v8s 在泰山赤鳞鱼的识别任务中存在明显的注意力分散问题。具体表现为部分目标个体漏检,以及对非目标区域的错误关注。在图 11f 中,基准模型将地面与养殖盆接缝处误判为有效特征,说明其抑制干扰信息的能力与全局识别能力均有不足。相比之下,改进后的模型视觉注意力分布更为集中且有序。在目标密集区域,注意力主要聚焦于鱼的头部;在目标稀疏区域,注意力则覆盖鱼体整体。这表明,本研究引入的 AIFIC 模块通过多尺度

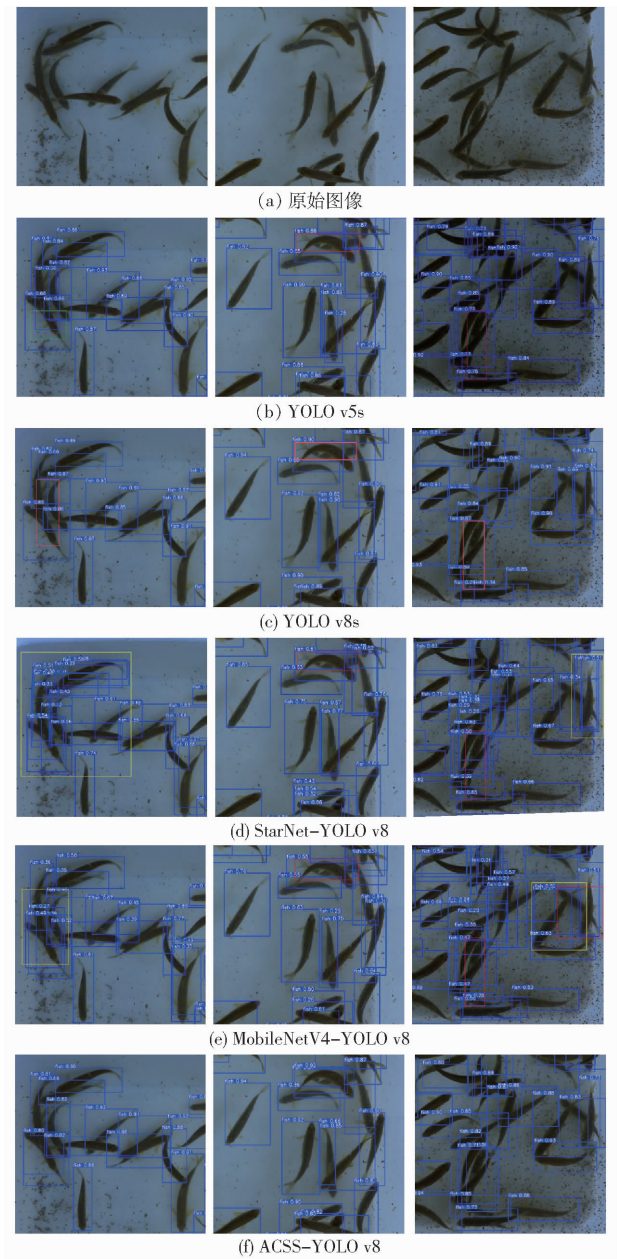


图 10 不同模型检测结果对比图

Fig. 10 Comparison image of detection results from different models

融合与深层特征交互,CAFM 通过全局与局部特征融合,以及 SEAM 模块对遮挡目标的有效增强,共同提升了对目标关键部位与整体的识别能力,加强了对遮挡部位的适应能力,并有效抑制了干扰。证明了本文所提出的改进模型在优化策略的有效性。

3.5 计数模块和模型部署分析

为实现 ACSS - YOLO v8 模型的计数功能,本研究首先设计了相应的计数模块。该模块在检测目标锚框数量后,可在图像左上角输出计数结果。如图 12 所示,计数模块识别并显示该图像中共包含 51 条泰山赤鳞鱼。

为验证模型的实际部署性能,本研究分别在主机端与 Jetson Nano 边缘计算设备上进行测试。改

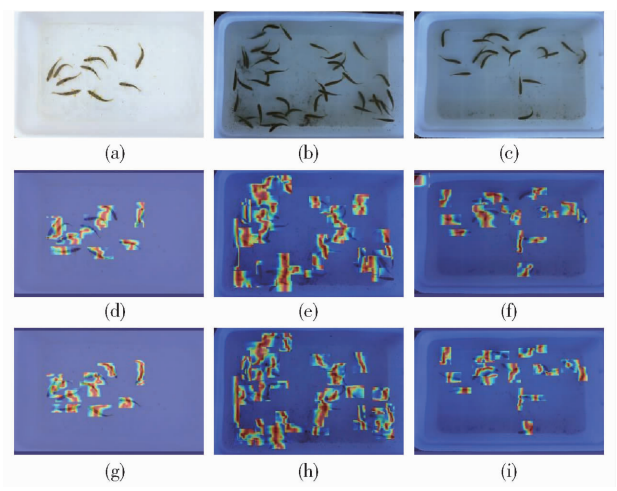


图 11 不同模型的 HiResCAM 可视化分析图

Fig. 11 Visualization analysis images of HiResCAM for different models

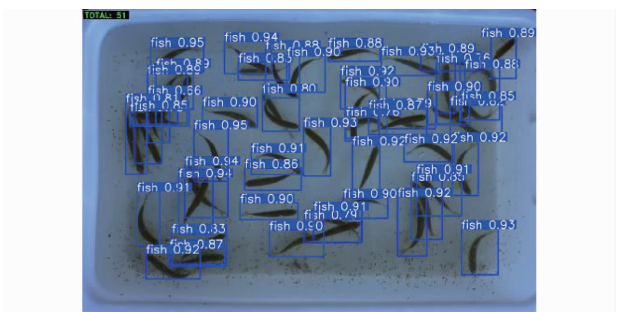


图 12 计数模块

Fig. 12 Counting module

进的 ACSS-YOLO v8 模型在主机端的推理速度达到 76 f/s,在 Jetson Nano 边缘设备上可实现 30 f/s 的检测速度。尽管边缘设备上的速度相对较低,但其性能已满足现场测试要求,表明该模型具备较高的实际部署价值与实用性。运行状态如图 13 所示。

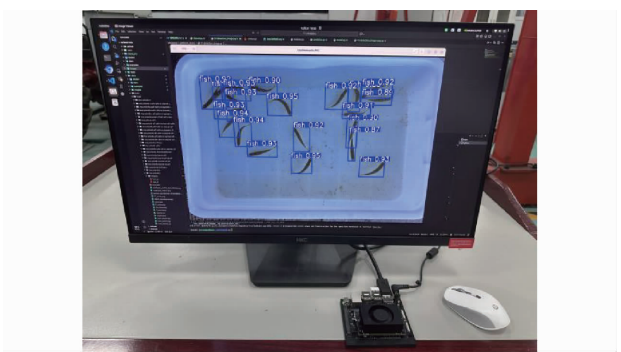


图 13 Jetson Nano 运行图

Fig. 13 Jetson Nano operation diagram

4 结论

(1) 建立了 ACSS-YOLO v8 模型。首先将 SPPF 层替换为 AIFIC 模块,增强了对高级特征的处理能力;在 AIFIC 模块后引入 CAFM 注意力机制,提高了模型的检测性能;然后在颈部网络中引入 Star Blocks,大大降低了计算复杂度;最后在检测头中引入 SEAM 模块,有效解决了目标间的遮挡问题。

(2)改进后的 ACSS-YOLO v8 模型在精确率、召回率、mAP₅₀、参数量、浮点运算量和内存占用量分别达到了 92.6%、85.8%、89.7%、 8.93×10^6 、 2.130×10^{10} 和 17.3 MB,精确率、召回率、mAP₅₀相较于 YOLO v8s 分别提升了 0.7、2.6、3.0 个百分点,参数量、浮点运算量和内存占用量比 YOLO v8s 分别降低了 20%、25% 和 20%。

(3)增加了直观的计数模块和 Jetson Nano 实验。在 Jetson Nano 部署中,检测速度达到 30 f/s,满足现场复杂的检测需要,保持了稳定的检测性能。

参 考 文 献

[1] 国务院新闻办公室发布《中国的远洋渔业发展》白皮书[J]. 中国水产,2023(11):2-4.

[2] 李道亮,李万超,杜壮壮. 基于音视频信息融合与 Self-Attention-DSC-CNN6 网络的鲈鱼摄食强度分类方法[J]. 农业机械学报,2025,56(1):16-24.

Li Daoliang, Li Wanchao, Du Zhuangzhuang. Classification method of feeding intensity of sea bass based on Self-Attention-DSC-CNN6 and multi-modal fusion[J]. Transactions of the Chinese Society for Agricultural Machinery,2025,56(1):16-24. (in Chinese)

[3] 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.

[4] Girshick R. Fast R-CNN[C]//Proceedings of the IEEE International Conference On Computer Vision, 2015: 1440-1448.

[5] Ren S, He K, 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, 2016, 39(6): 1137-1149.

[6] He K, Gkioxari G, Dollár P, et al. Mask R-CNN[C]// Proceedings of the IEEE International Conference on Computer Vision, 2017: 2961-2969.

[7] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector[C]//Computer Vision-ECCV 2016: 14th European Conference. Amsterdam: Springer International Publishing, 2016: 21-37.

[8] Yang C, Liu H. Channel pruning based on convolutional neural network sensitivity[J]. Neurocomputing, 2022, 507: 97-106.

[9] Wang J, Chen Y, Zheng Z, et al. CrossKD: cross-head knowledge distillation for object detection[C]// Proceedings of the

IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024; 16520 – 16530.

[10] Zhu X, Li J, Liu Y, et al. A survey on model compression for large language models[J]. Transactions of the Association for Computational Linguistics, 2024, 12: 1556 – 1577.

[11] Xie F, Lu P, Liu X. Multi-scale convolutional attention network for lightweight image super-resolution[J]. Journal of Visual Communication and Image Representation, 2023, 95: 103889.

[12] Ma X, Dai X, Bai Y, et al. Rewrite the stars[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024; 5694 – 5703.

[13] Han K, Wang Y, Tian Q, et al. Ghostnet: more features from cheap operations[C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020; 1580 – 1589.

[14] Qin D, Leichner C, Delakis M, et al. MobileNetV4: universal models for the mobile ecosystem[C]//European Conference on Computer Vision. Cham: Springer Nature Switzerland, 2024; 78 – 96.

[15] 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.

[16] Zhang H, Li W, Qi Y, et al. Dynamic fry counting based on multi-object tracking and one-stage detection[J]. Computers and Electronics in Agriculture, 2023, 209: 107871.

[17] 尹莉娜,陈晓燕,黄芳,等. U – Fish:一种基于密度图回归的鱼群计数方法[J]. 南京农业大学学报,2026,49(1):214 – 224.

Yin Lina,Chen Xiaoyan,Huang Fang,et al. U – Fish: a fish school counting method based on density map regression[J]. Journal of Nanjing Agricultural University,2026,49(1): 214 – 224. (in Chinese)

[18] 张璐,黄琳,李具备,等. 基于多尺度融合与无锚点 YOLO v3 的鱼群计数方法[J]. 农业机械学报,2021,52(增刊1): 237 – 244.

Zhang Lu, Huang Lin, Li Beibei, et al. Fish school counting method based on multi-scale fusion and no anchor YOLO v3[J]. Transactions of the Chinese Society for Agricultural Machinery,2021,52(S1):237 – 244. (in Chinese)

[19] Wang H, Zhang S, Zhao S, et al. Real-time detection and tracking of fish abnormal behavior based on improved YOLOV5 and SiamRPN + + [J]. Computers and Electronics in Agriculture, 2022, 192: 106512.

[20] 郑荣才,谭鼎文,徐青,等. 基于改进 YOLO v7 的鲑鱼检测模型轻量化研究[J]. 农业机械学报,2024,55(11):132 – 139.

Zheng Rongcai, Tan Dingwen, Xu Qing, et al. Lightweight salmon detection model based on improved YOLO v7 [J]. Transactions of the Chinese Society for Agricultural Machinery,2024,55(11):132 – 139. (in Chinese)

[21] 刘晓阳,王文良,郑欣宇,等. 面向池塘养殖的水下海参原位计数无人船系统设计与试验[J]. 农业机械学报,2025, 56(10):82 – 93.

Liu Xiaoyang, Wang Wenliang, Zheng Xinyu, et al. Design and testing of unmanned boat system for in-situ counting of sea cucumbers in pond aquaculture[J]. Transactions of the Chinese Society for Agricultural Machinery, 2025,56(10):82 – 93. (in Chinese)

[22] Zhao Y, Lv W, Xu S, et al. Detrs beat yolos on real-time object detection[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024; 16965 – 16974.

[23] Hu S, Gao F, Zhou X, et al. Hybrid convolutional and attention network for hyperspectral image denoising[J]. IEEE Geoscience and Remote Sensing Letters, 2024, 21;5.

[24] Yu Z, Huang H, Chen W, et al. YOLO – facev2: a scale and occlusion aware face detector[J]. Pattern Recognition, 2024, 155: 110714.