

DOI:10.11784/tdxbz202502004

## 基于深度学习及规则推理的堆石坝 卸料时间实时感知模型研究

俞 澎, 王晓玲, 张 君, 曾拓程, 王佳俊

(天津大学水利工程智能建设与运维全国重点实验室, 天津 300350)

**摘要:** 堆石坝卸料时间的实时感知对于坝面施工进度仿真分析和施工机械管理调度具有重要意义。基于深度学习的计算机视觉技术由于能够实现施工机械的实时检测与跟踪,为卸料时间感知提供了有效途径,然而现有方法在大场景视频监控下进行自卸汽车卸料活动识别时存在速度慢、精度低的问题,且难以分析自卸汽车在不同填筑分区上卸料作业规律的差异性。针对上述问题,提出基于深度学习及规则推理的堆石坝卸料时间实时感知模型。基于显式视觉中心(EVC)机制改进 YOLOv8 目标识别算法的特征金字塔结构,以弥补 YOLOv8 在特征提取过程中难以完全捕获小尺度目标全局信息和易丢失局部特征的不足,提高 YOLOv8 对于小尺度目标的识别精度,并将改进的 YOLOv8 目标识别算法(EVC-YOLOv8)作为 ByteTrack 跟踪算法的目标检测器,以提高 ByteTrack 跟踪算法对小尺度目标的跟踪精度。此外,引入 YOLOv8-seg 算法分割提取目标填筑分区范围,实现自卸汽车目标检测与跟踪的分区统计,进一步基于规则推理提取堆石坝不同分区的卸料时间。案例分析表明:EVC-YOLOv8 目标识别算法较基准模型识别精度提高了 2.3%;改进的 ByteTrack 模型在 25 帧/s 时的多目标跟踪精度为 90.6%,目标识别准确率为 93.2%,满足仿真参数获取的实时性和精度的需求;检测获取的卸料时间平均误差为 4.2%,验证了本模型的准确性与实时性。

**关键词:** 堆石坝; YOLOv8-seg 模型; EVC-YOLOv8 模型; 改进的 ByteTrack 模型; 规则推理; 卸料时间参数

**中图分类号:** TV52 **文献标志码:** A **文章编号:** 0493-2137(2025)12-1288-14

## Research on Real-Time Perception Model for Unloading Time on Rockfill Dams Based on Deep Learning and Rule Reasoning

Yu Peng, Wang Xiaoling, Zhang Jun, Zeng Tuocheng, Wang Jiajun

(State Key Laboratory of Hydraulic Engineering Intelligent Construction and Operation, Tianjin University,  
Tianjin 300350, China)

**Abstract:** The real-time perception of unloading time on rock-fill dams is of significance for the simulation analysis of dam surface construction progress, as well as the management and scheduling of construction machinery. The computer vision technology based on deep learning provides an effective way to sense the unloading time, as it can achieve real-time detection and tracking of construction machinery. However, the existing methods are slow, inaccurate and unable to analyze the differences in unloading operations of dump trucks in different filling zones when recognizing unloading activities in large-scale video surveillance. To address these issues, a real-time perception model for the unloading time on rockfill dams based on deep learning and rule reasoning is proposed. This model improves the feature pyramid structure of the YOLOv8 object detection algorithm by incorporating the explicit visual center(EVC)mechanism to compensate for the limitations of YOLOv8 in fully capturing the global information for small-scale objects and its tendency to lose local features during the feature extraction. This improvement enhances YOLOv8's detection accuracy for small-scale objects, and the improved YOLOv8 (EVC-YOLOv8)is used as the

收稿日期: 2025-02-09; 修回日期: 2025-05-07.

作者简介: 俞 澎(1997—),男,博士研究生,yupeng\_0929@tju.edu.cn.

通信作者: 张 君,zhangdajun@tju.edu.cn.

基金项目: 国家自然科学基金青年科学基金资助项目(52309165).

Supported by the Youth Science Fund Project of the National Natural Science Foundation of China(No. 52309165).

object detector for the ByteTrack tracking algorithm, which improves the ByteTrack's tracking accuracy for small-scale objects. Additionally, the YOLOv8-seg algorithm is introduced to segment and extract the target filling zones, enabling partitioned statistics for dump truck detection and tracking. The unloading time in different zones of the rock-fill dam is then extracted based on rule reasoning. Case analysis shows that the EVC-YOLOv8 object detection algorithm improves the detection accuracy by 2.3% compared with that of the baseline model. The improved ByteTrack model achieves a multiple object tracking accuracy of 90.6% and an identification  $F_1$  score of 93.2% at 25 frame/s, meeting the real-time and accuracy requirements for simulation parameter acquisition. The average error in detected unloading time is 4.2%, verifying the accuracy and real-time performance of this model.

**Keywords:** rockfill dam; YOLOv8-seg model; EVC-YOLOv8 model; improved ByteTrack model; rule reasoning; unloading time parameter

堆石坝施工由于其规模大、周期长、填筑方量大、过程复杂等特点,受到复杂动态不确定性因素的影响和诸多施工条件的约束,给工程施工资源的合理组织与施工过程管控指令的高效决策带来了挑战。堆石坝施工仿真作为一种辅助施工组织设计的有效手段,为堆石坝建设过程的模拟、分析、优化和控制提供了有效途径,在堆石坝施工组织优化与进度管控中得到了广泛的应用<sup>[1-2]</sup>。由于施工参数具有随着施工推进而动态变化的特征,因此,堆石坝施工仿真参数的实时准确建模是确保堆石坝施工仿真准确性和实时性的关键。随着计算机水平的提高和人工智能的高速发展,计算机视觉、云计算、大数据、物联网等新一代技术与大坝建设深度融合,大坝建设逐渐由数字化施工向智能化施工推进,基于新一代信息技术进行堆石坝施工仿真参数实时准确建模将进一步提高施工仿真的实时性和有效性。堆石坝施工是一个由坝料开挖、坝料运输、坝料填筑共3部分构成的循环往复的过程,其中卸料工序是制约运输和填筑进程的关键工序,卸料时间的实时准确建模对堆石坝施工过程的精确仿真和优化运输配置具有重要意义<sup>[3]</sup>,因此,如何基于新一代信息技术实时准确获取自卸汽车(truck)卸料时间是保证堆石坝施工仿真实时性及提高施工仿真准确性的关键。

早期土石方工程施工仿真参数主要使用静态输入数据,通过专家经验或同类项目经验人工统计获取仿真工序参数,不具备实时感知处理观测数据的能力<sup>[4]</sup>,且无法反映实际施工过程的不确定性。随着定位技术的快速发展,在21世纪初期以全球定位系统(global positioning system, GPS)和实时定位系统(real-time location systems, RTLSS)为主要代表的空间定位技术被应用于堆石坝施工监控, GPS和RTLSS也成为最早被引入堆石坝施工仿真以提高仿真模型准确性的信息感知技术<sup>[5]</sup>。Pradhananga等<sup>[6-7]</sup>基于GPS定位标签记录自卸汽车和挖机进入现场与

离开现场的时刻,进而统计仿真活动时间参数并用于基于元胞的仿真建模;同时,创建了GPS数据可视化界面实现施工设备的自动时空分析,并能够通过手动或自动的方法分析机械活动时间<sup>[8]</sup>。Hildreth等<sup>[9]</sup>开发了1种基于GPS数据自动提取仿真活动起止时间的程序,并将其应用于离散事件仿真中。Vahdatikhaki等<sup>[10-11]</sup>基于实时定位技术提出了一个近实时仿真的框架,从机械位置数据中挖掘卡车和挖机的运动状态,进而生成仿真工序时长参数,以驱动仿真模型近实时更新。钟登华等<sup>[12]</sup>通过碾压实时监控获取的定位数据生成碾压机速度、偏转角度以及错距时间等碾压仿真工序参数。钟登华等<sup>[13]</sup>将基于堆石坝施工实时监控系统中获取的机械活动时间、仓面单元面积、碾压机速度等信息作为仿真输入。刘宁<sup>[14]</sup>根据交通运输GPS实时监控采集到的瞬时车速更新堆石坝运输仿真模型。Song等<sup>[15]</sup>基于GPS传感器开发了1种可以通过知识规则提取活动时间参数的方法,进而实现随着施工环境不断变化的自适应仿真。上述研究表明,实时定位技术是获取仿真参数的重要途径,然而大型堆石坝施工过程中自卸汽车数量通常多达几百辆且分属不同车队,在施工过程中车辆极有可能频繁更换<sup>[14]</sup>,为自卸汽车GPS设备的安装和维护造成了较大的困难,对此需要投入大量的人力、物力和时间成本。

近年来,随着人工智能的不断发展,基于深度学习的计算机视觉技术成为一种分析施工活动的有效手段<sup>[16]</sup>。许多学者基于计算机视觉技术对施工机械活动状态的识别<sup>[17-19]</sup>和相关施工参数的提取进行了研究。Gong等<sup>[20]</sup>提出了1种可以从施工作业视频监测内容中提取混凝土施工循环活动时间的推理模型,提供了1种基于视频的施工活动时间参数感知方法。Kim等<sup>[21]</sup>提出了1种基于区域的全卷积网络的施工设备目标检测模型,结合上、下文推理提取隧道土方施工仿真所需的装载活动时间参数和运输回路周期

时间参数,并输入至 WebCYCLONE 仿真模型中以提高仿真的准确性;此后,其又相继开发了 1 种基于视频识别施工现场出入口处自卸汽车车牌检测与识别算法,生成自动访问日志,进而计算运输循环时间参数<sup>[22]</sup>. Zheng 等<sup>[23]</sup>通过监控视频来识别跟踪预制件模块安装活动,从而获取预制件在摄像机视野内的持续时间. Wang 等<sup>[24]</sup>提出了 1 种基于增强现实(augmented reality, AR)技术和卷积神经网络视觉检测的缆机运输循环时间检测方法,为拱坝施工参数建模提供了 1 种新的途径. Alsakka 等<sup>[25]</sup>提出了 1 种基于计算机视觉的过程时间数据采集(time data acquisition, TiDA)系统,在板材施工过程中采集生产时间、周期时间和流程启动时间. Wang 等<sup>[26]</sup>将 YOLOv5 和 CC-SORT 应用于大坝施工中缆索起重机运输的监测,通过跟踪吊斗的周期性活动轨迹,获取缆索起重机的运输时间参数并用于进行生产力分析. Kim 等<sup>[27]</sup>提出 1 种通用视觉框架用于计算建筑资源的总投入时间和机械运行状态的持续时间,检测建筑资源在监控中的首次出现时间(开始时间)和消失时间(结束时间),然后通过从到达时间中减去出发时间(结束时间减去开始时间)来计算投入时间. Wang 等<sup>[28]</sup>采用级联 R-CNN 的单摄像头跟踪进行土方运输车检测,基于改进的 TransReID 进行外观特征提取,结合 WDA-Tracker 算法实现跨不连续摄像头视角的目标关联,从而获取对应土方运输车在指定路段上的运输时间并用于运输效率分析. 曾拓程等<sup>[3]</sup>提出融合 ByteTrack、HRNet 与 DCL 的大场景监控视频运输车卸料识别方法,采用机器视觉技术进行运输车卸料识别研究. 虽然现有研究在施工仿真参数的获取方面取得了一定进展,但对于自卸汽车卸料环节的关键参数如卸料时间的自动提取尚未形成成熟、可靠的获取机制. 在大场景视频监控环境下,自卸汽车通常以小尺度目标形式存在,受限于目标尺寸小、特征不显著、背景复杂等因素,精确识别与跟踪小尺度目标也成为当前施工仿真视觉识别中的 1 个技术难题.

小尺度目标检测作为计算机视觉研究中的 1 个重要方向,在复杂背景和资源有限的实际应用场景中展现出广阔的应用前景. Zhang 等<sup>[29]</sup>提出了 1 种基于 DsPAN 的小物体检测网络,改进 YOLOv8 框架并通过优化 PAN 路径和引入注意力机制,有效提高了多尺度缺陷检测精度,其小物体检测中表现优越. Zeng 等<sup>[30]</sup>提出了 1 种基于混合注意力机制的 SCA-YOLO 算法,通过设计坐标注意力模块和改进的 SEB 模块,结合多层特征融合结构,有效提升了无人

机图像中小物体的检测精度. Jiang 等<sup>[31]</sup>针对无人机图像中小目标比例高、尺度差异大等问题,提出多尺度特征融合小目标检测网络 MFFSODNet,通过引入小目标预测头、MSFEM 模块和双向密集特征金字塔结构,有效提升了小目标检测精度及模型的泛化能力. Dang 等<sup>[32]</sup>针对遥感图像中小目标检测精度低的问题,提出改进 YOLOv7 的 YOLO-FNC 模型,引入 C3-Faster 模块、NWD-GIoU 损失函数与坐标注意力机制,有效提升了复杂背景下小目标的检测性能. Liu 等<sup>[33]</sup>提出了 1 种新的小目标检测框架 DNTR,通过引入去噪特征金字塔网络(denoising feature pyramid network, DN-FPN)和基于 Transformer 的 Trans-R-CNN 检测器,有效抑制了多尺度特征融合中的噪声,并提升了小目标检测的精度. Wang 等<sup>[34]</sup>提出了基于 YOLOv8 的 YOLOv8-QSD 网络,结合双向特征金字塔网络和查询式模型,提升了自驾场景中小目标的检测精度与速度. Ma 等<sup>[35]</sup>通过融合 GhostNet 骨干和多尺度注意力机制,提出 1 种适用于嵌入式设备的小目标检测算法,兼顾模型压缩与检测性能. Yue 等<sup>[36]</sup>提出 LE-YOLO 算法,引入轻量主干网络 LHGNet 和 LGSCSP 融合模块,在保证精度的前提下降低了模型参数,具备良好的边缘部署适应性. Peng 等<sup>[37]</sup>提出了 1 种改进的 YOLOv7 模型,通过引入可变形卷积特征提取器(deformable convolutional feature extractor, DCFE)和注意力特征提取器(attention feature extractor, AttFE),提高了机场场景中小物体和遮挡物体的检测精度. 现有小尺度目标检测研究主要集中于遥感、无人机和自动驾驶等领域,尚缺乏面向特定施工场景下如自卸汽车卸料过程等具有时序性与结构性特点的检测机制. 同时,现有方法多集中于提升单帧图像中的小目标检测性能,对于大场景视频监控中自卸汽车卸料活动的连续识别仍存在识别速度慢、检测精度不高等问题.

上述研究为堆石坝卸料作业规律的分析提供了有效途径,但现有方法在大场景视频监控下进行自卸汽车卸料活动识别时存在速度慢、精度低的问题. 此外,在堆石坝不同的填筑分区仓面上(如心墙、过渡、反滤等),由于坝料性质、自卸汽车型号不同,卸料时间通常呈现不同的规律特征,而现有研究无法自动匹配识别对象与目标填筑分区,难以分析自卸汽车在不同填筑分区上卸料作业规律的差异性. 因此,如何提高大场景监控视频下小尺度自卸汽车的识别跟踪精度,自动匹配自卸汽车位置与填筑分区位置,并且基于自卸汽车卸料活动时长统计的逻辑规则推理获取

对应填筑分区的卸料时间参数是视频感知在高心墙堆石坝施工活动智能分析中应用的难点之一。

针对上述难题,本研究提出 1 种基于深度学习及规则推理的堆石坝卸料时间实时感知模型,其能够在提高大场景监控视频下小尺度自卸汽车识别跟踪精度的基础上,自动匹配自卸汽车位置与填筑分区位置,从而获取对应填筑分区的卸料时间参数,为堆石坝施工仿真提供可靠的仿真参数输入。

### 1 研究框架

本文的研究框架如图 1 所示,基于深度学习及规则推理的堆石坝卸料时间实时感知模型的构建与实施主要由以下 4 部分组成。

(1) 构建基于 EVC-YOLOv8 的自卸汽车识别与分区统计模型。首先,基于显式视觉中心(explicit visual center, EVC)机制改进 YOLOv8 模型进行小尺度自卸汽车的识别训练,以弥补 YOLOv8 在特征提取过程中难以完全捕获小尺度自卸汽车全局信息和易丢失局部特征的不足。其次,采用 YOLOv8-seg 算

法对大场景坝面监控画面自动进行目标填筑分区的分割提取,以实现自卸汽车位置与填筑分区位置的自动匹配。

(2) 基于改进 ByteTrack 的多目标跟踪。将上述 EVC-YOLOv8 模型作为 ByteTrack 的目标检测器以提高其对自卸汽车的识别跟踪精度,在提取的目标填筑分区上通过改进的 ByteTrack 进行自卸汽车的多目标识别跟踪,以实现目标填筑分区中自卸汽车的实时跟踪效果。

(3) 基于规则推理的卸料时间统计。基于工程实际情况提出自卸汽车卸料时间的定义,并制定卸料时间统计规则。在目标检测与跟踪结果的基础上,基于提出的规则推理方法统计每辆自卸汽车在目标填筑分区停留的时间作为卸料时间,进而完成堆石坝卸料时间的实时感知。

(4) 工程案例分析。将所建立的堆石坝卸料时间实时感知模型运用于实际大坝填筑工程中,并将获取的自卸汽车卸料时间与现场统计的实际卸料时间进行对比,采用平均误差与对角误差图来表征模型的感知效果,验证所提模型的准确性与实时性。

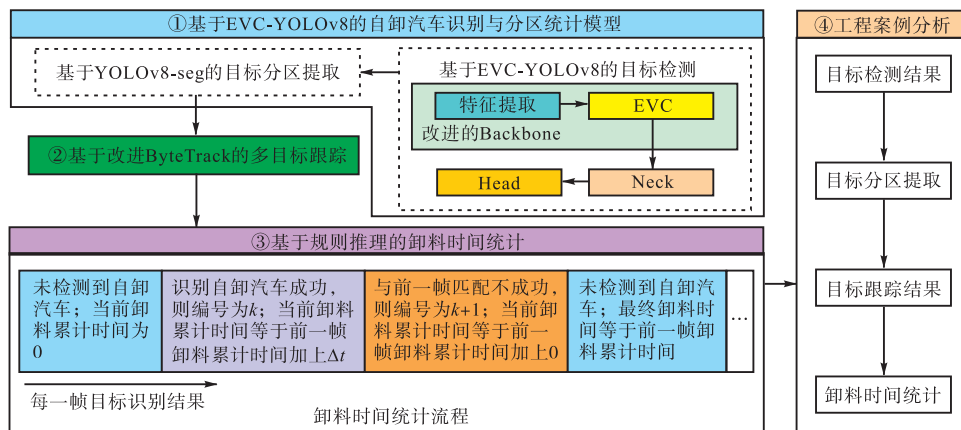


图 1 研究框架

Fig.1 Research framework

## 2 研究方法

### 2.1 基于 EVC-YOLOv8 的自卸汽车识别与分区统计模型

现有研究在大场景视频监控下对小尺度自卸汽车的识别过程中存在全局特征与局部特征丢失等问题,导致识别效果不佳,且存在识别对象无法与目标填筑分区自动匹配的问题。针对上述问题,提出基于 EVC-YOLOv8 的自卸汽车识别与分区统计模型,其中,基于 EVC 机制改进 YOLOv8 目标识别算法的特

征金字塔结构,以弥补 YOLOv8 在特征提取过程中难以完全捕获小尺度目标全局信息和易丢失局部特征的不足,提高 YOLOv8 对于小尺度目标的识别精度。此外,通过 YOLOv8-seg 算法分割提取目标填筑分区范围,实现自卸汽车识别的分区统计。

#### 2.1.1 基于 EVC-YOLOv8 的大场景监控视频自卸汽车目标识别

YOLOv8 作为 YOLO (You Only Look Once) 系列目标检测算法的最新版本,以其实时检测能力和较高的准确性而被广泛应用<sup>[38-40]</sup>,它包括 3 个模块: Backbone、Neck 和 Head。与 YOLOv5<sup>[41]</sup>模型相比,在

YOLOv8 模型的 Backbone 中第 1 个卷积模块中的卷积核从  $6 \times 6$  改为  $3 \times 3$ , C3 模块被基于 CSP 思想的 C2f 模块取代, 其中 C2f 模块借鉴了 YOLOv7 中的 ELAN 思想, 将 C3 和 ELAN 相结合, 形成了 C2f 模块<sup>[42]</sup>, C2f 模块使用 3 个卷积模块和  $n$  个 bottleneck, 这使得模型进一步轻量化并获得更丰富的梯度流信息. 在 backbone 的尾部, 使用了流行的 SPPF 模块, 通过串联 3 个  $5 \times 5$  的 Maxpool 层来处理不同尺寸的特征图, 以确保对不同尺度目标的准确检测. 在 Neck 部分, YOLOv8 使用的特征融合方法仍然是 PAN-FPN, 使用了两个上采样和多个 C2f 模块来组成 Neck 模块, 加强了不同尺度特征层信息的融合和利用. 在 Head 部分, YOLOv8 借鉴了 YOLOX 中的 Decouple Head 结构, 将分类和检测头分开, 同时也将 Anchor-Based 换成了 Anchor-Free, 以获得一个新的准确性水平.

虽然 YOLOv8 以其实时检测速度和较高的检测精度而被广泛应用, 但其在特征提取过程中难以完全捕获小尺度目标全局信息, 且易丢失局部特征, 导致 YOLOv8 对大场景下小尺度目标的检测存在一定的局限性. 基于全局显式集中调节方案的目标检测算法——中心化特征金字塔(centralized feature pyramid, CFP)<sup>[43]</sup>, 不仅关注不同层之间的特征交互, 还考虑了同一层内的特征调节. CFP 提出的显式视觉中心不仅能够捕获全局信息, 还能聚合局部信息, 从而高效地获得全方位的、具有判别力的特征表示. EVC 主要由两个并行连接的模块组成, 如图 2 所示, 其中轻量级多层感知机(multilayer perceptron, MLP)用于

捕获顶级特征的全局远程依赖关系(即全局信息). 同时, 为了保留局部角区域(即局部信息), 采用可学习视觉中心(learnable visual center, LVC)机制来聚合层内局部区域特征. 这两个模块的结果特征图沿着通道维度连接在一起, 作为下游识别的 EVC 的输出. 相比于直接映射原始特征, EVC 添加 Stem Block 以进行特征平滑. Stem Block 由输出通道大小为 256 的  $7 \times 7$  卷积组成, 后面是批量归一化层和激活函数层, 上述过程可以表述如下.

$$X = \text{Cat}(\text{MLP}(X_m); \text{LVC}(X_m)) \quad (1)$$

式中:  $X$  为 EVC 的输出;  $\text{Cat}(\cdot)$  为沿通道维度串联的特征图;  $\text{MLP}(X_m)$  和  $\text{LVC}(X_m)$  分别为所使用的轻量级 MLP 和 LVC 机制的输出特征;  $X_m$  是 Stem Block 的输出, 其可表示为

$$X_m = \sigma(\text{BN}(\text{Conv}_{7 \times 7}(X^*))) \quad (2)$$

式中:  $X^*$  为 Stem Block 的输入, 即顶级特征;  $\text{Conv}_{7 \times 7}(\cdot)$  为步长为 1 的  $7 \times 7$  卷积, 通道大小设置为 256;  $\text{BN}(\cdot)$  为批量归一化层;  $\sigma(\cdot)$  为 ReLU 激活函数.

图 2 中轻量级 MLP 主要由基于深度卷积的模块和基于通道 MLP 的模块组成, 其中基于 MLP 模块的输入是基于深度卷积模块的输出. 这两个模块之后都是通道缩放操作和正则化操作, 以提高特征泛化和鲁棒性能力. 可学习视觉中心模块通过卷积层增减通道数, 以此减少计算量并且允许模型更好地收敛. 然后, 将经过 CBR 模块处理后的特征输入至码本中, 其中平滑因子  $S$  将输入特征和固有码本  $B =$

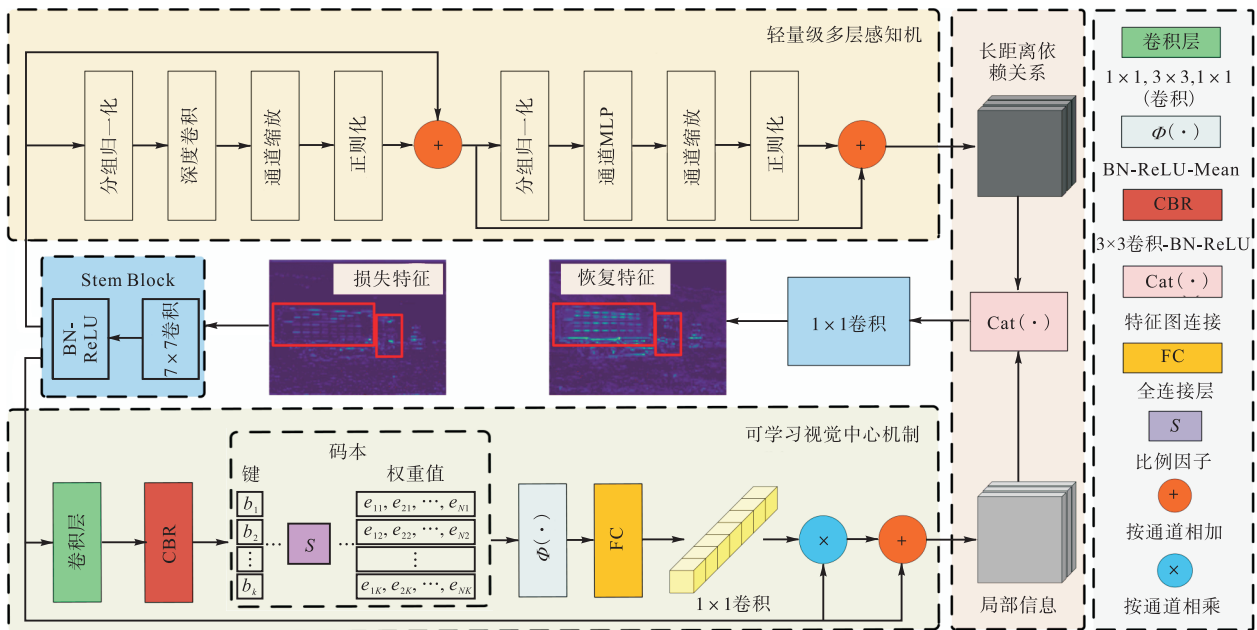


图 2 EVC 网络结构

Fig.2 Network structure of EVC

$(b_1, b_2, \dots, b_k)$  逐一匹配, 并基于两者的差生成权重, 然后将权重与差相乘来求和, 以输出结果特征。

图 3 展示了所提的 EVC-YOLOv8 模型的网络结构, 该网络由输入、Backbone、Neck、Head 和输出组成, 其中 EVC 模块位于 Backbone 最后 1 层 SPPF 模块之后, 用于捕获顶级特征的全局远程依赖关系 (即全局信息), 以及聚合层内局部区域特征。然后, 再将 EVC 的输出特征进行上采样而从与浅层特征进行特征融合, 以提高小尺度目标的识别精度。

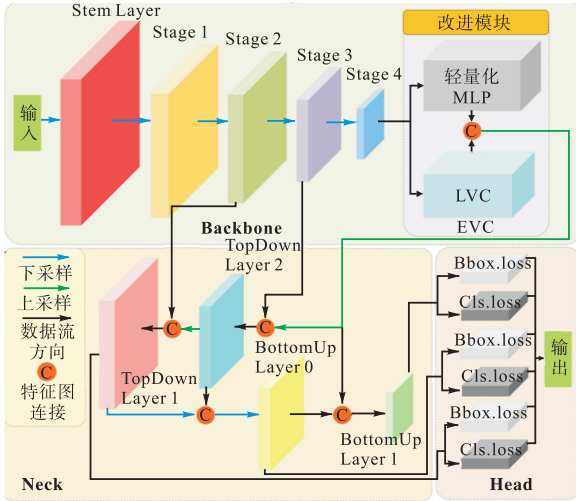


图 3 EVC-YOLOv8 模型网络结构

Fig.3 Network structure of EVC-YOLOv8 model

### 2.1.2 基于 YOLOv8-seg 的堆石坝目标填筑分区提取

为获取与堆石坝坝面分区匹配的自卸汽车卸料时间及其参数分布, 可采用图像分割算法进行监控视频图像中目标分区的提取, 然后再分别针对不同分区进行目标检测与跟踪。

YOLOv8-seg 是 YOLOv8 的实例分割模型, 与对象检测模型相比, 区别在于实例分割模型在头部结构中具有原型掩码分支和掩码系数, 用于生成实例掩码。YOLOv8-seg 模型的 Head 模块采用 YOLACT<sup>[44]</sup> 的思想, 分为两个分支: Detect head 和 proto. Detect head 是一种解耦的无锚结构, 用于实现目标检测和分类。检测头的检测函数中使用的损失函数是完全交叉并集 (complete intersection over union, CIoU) 损失和分布焦点损失 (distribution focal loss, DFL), 分类函数中使用的损失函数是二元交叉熵 (binary cross-entropy, BCE) 损失函数, 3 个损失函数的表达式为

$$\begin{cases} l_{\text{CIoU}} = 1 - l_{\text{IoU}} + \frac{\rho^2(b, b^{\text{gt}})}{c^2} + \alpha v \\ \alpha = v / (1 - l_{\text{IoU}}) + v \\ v = \frac{4}{\pi^2} \left[ \arctan(\omega^{\text{gt}} / h^{\text{gt}}) - \arctan(\omega / h) \right]^2 \end{cases} \quad (3)$$

式中:  $b$  和  $b^{\text{gt}}$  分别为预测和真实边界框的中心点;  $\rho$  为两个中心点的欧几里得距离;  $c$  为可以同时包含这两个框的最小闭合区域的对角线距离;  $w$  和  $h$  分别为预测边界框的宽和高;  $w^{\text{gt}}$  和  $h^{\text{gt}}$  分别为真实边界框的宽和高;  $\alpha$  与  $v$  为权重系数, 用于计算两个框长宽比的相似性;  $l_{\text{IoU}}$  为交叉并集损失。

$$l_{\text{DFL}} = - \left[ (y_{i_0+1} - y) \ln s_{i_0} + (y - y_{i_0}) \ln s_{i_0+1} \right] \quad (4)$$

式中:  $s$  为网络 sigmoid 激活函数的输出;  $y$  为边界框的坐标;  $i_0$  为框的变化范围。

$$l_{\text{BCE}} = - \frac{1}{N} \sum_{i_1=1}^N \left[ \text{target}[i_1] \cdot \ln(\phi[i_1]) + (1 - \text{target}[i_1]) \cdot \ln(1 - \phi[i_1]) \right] \quad (5)$$

式中:  $\text{target}[i_1]$  为目标的标签;  $\phi[i_1]$  为网络预测的标签;  $N$  为目标的总数;  $i_1$  为目标的编号。

另一个分支 proto 用于获取具有 BCEWithLogits-loss 损失函数的原型掩码, 该函数本质上是 BCE 损失加上 sigmoid 函数, 通过对两个分支的结果进行加权和阈值化来获得最终检测结果。

## 2.2 基于改进 ByteTrack 的大场景监控视频自卸汽车多目标实时跟踪

ByteTrack 是由 Zhang 等<sup>[45]</sup>提出的 1 种多目标跟踪模型, 采用 YOLOX 作为检测器, 相比于以往多目标跟踪算法直接丢弃置信度较低的目标, ByteTrack 保留了全部检测结果, 并将它们分为高分值组与低分值组。首先, 将高分值组与跟踪对象相关联。然后, 将低分值检测结果与未匹配的跟踪对象相关联来保留低分值检测结果并过滤背景。

Byte 策略包含 1 个物体检测器、1 个卡尔曼滤波器以及 3 个阈值 (检测阈值  $T_{\text{high}}$ 、 $T_{\text{low}}$  与跟踪阈值  $E$ ), 其输入为视频中的每一帧图像, 输出为物体的坐标以及唯一性编号。在实际运行过程中, 对于每一帧图像将检测器输出的物体按照阈值分组, 大于  $T_{\text{high}}$  的归入高分值组, 介于  $T_{\text{high}}$  与  $T_{\text{low}}$  之间的归入低分值组。同时, 使用卡尔曼滤波器预测已跟踪物体的下一个位置。

第 1 次数据匹配涉及检测结果的高分值组与全部跟踪对象。基于 IoU 计算相似度, 再使用匈牙利算法完成匹配。此时, 未匹配的检测结果与跟踪结果依然予以保留。

第 2 次数据匹配涉及检测结果的低分值组与上一步剩余的跟踪对象。完成匹配后, 保留未匹配的跟踪对象并丢弃仍未匹配的检测结果。这一步需要优先使用 IoU 作为匹配依据, 因为低分值对象的外观容易发生劣化。

在匹配完成后,第 2 次数据匹配也失败的跟踪对象会被暂存 30 帧,若超过 30 帧仍未获得匹配,则丢弃该跟踪对象.

最后 1 步则是跟踪新出现的物体,这些物体来自未被匹配的高分值检测结果.对于每一个检测结果,如果其分值高于跟踪阈值  $E$  且在连续两帧中出现,则将其纳入跟踪对象集合中.

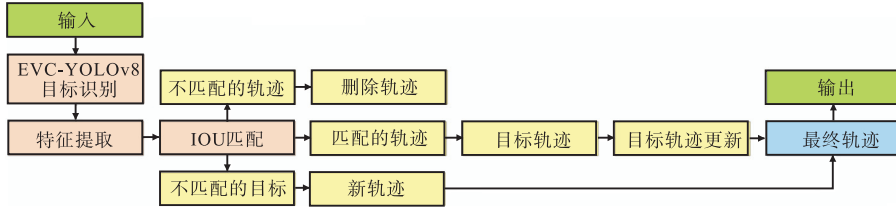


图 4 改进的 ByteTrack 目标跟踪流程

Fig.4 Flow chart of improved ByteTrack object tracking process

经过第 2.1.2 节处理后的坝面视频监控视频中只保留目标分区,通过改进 ByteTrack 模型检测并跟踪目标分区中的卸料汽车,以实现大场景坝面监控画面下跟踪目标与对应填筑分区的自动匹配.

### 2.3 基于规则推理的堆石坝卸料时间实时感知

自卸汽车卸料时间指每台自卸汽车在对应填筑分区中进行卸料活动的时长,单台自卸汽车通常将大坝填筑分区对应的土石料从料场运输至卸料点,并空载返回至料场.由于多目标跟踪算法无法自动统计跟踪目标的活动时长,且可能存在错误跟踪轨迹对时长统计的影响,因此在第 2.1 节和第 2.2 节目标分区坝面自卸汽车目标检测与跟踪的基础上,基于规则推理方法计算自卸汽车单次卸料时间.规则推理是一种基于规则系统进行逻辑推理的方法,由于其生成的决策过程是透明和可解释的,因此采用该方法进行卸料时间的统计可以排除错误跟踪轨迹对卸料时间的影响,同时易于理解和验证.考虑到自卸汽车进入填筑分区后将影响其他工序的正常进行,所以将一辆自

将 ByteTrack 原有的 YOLOX 检测器替换为 EVC-YOLOv8 进行改进,以此用于快速实现坝面监控视频图像的高效特征提取及目标识别跟踪,并记录目标车辆的行驶轨迹,为卸料时间的获取提供车辆型号、编号和轨迹等信息.改进的 ByteTrack 目标跟踪流程如图 4 所示.

卸汽车进入分区到离开分区的时长识别为单辆自卸汽车的卸料时间.根据目标分区的自卸汽车目标检测与跟踪结果,基于规则推理的卸料时间参数分析流程如图 5 所示.首先,填筑分区开始施工后调取坝面视频监控,采用 YOLOv8-seg 模型进行目标填筑分区的提取.其次,通过改进的 ByteTrack 多目标跟踪算法获取编号  $k$  的自卸汽车是否在对分区内的目标活动参数  $B_k(t')$ .进一步,以  $\Delta t'$  的时间间隔提取监控关键帧,持续进行目标检测与跟踪直至该分区完成施工,结合规则推理获取单台自卸汽车在目标分区中的卸料时间,其计算式为

$$T_k^d = \sum_{t'=t_0}^{t_{end}} \Delta t' B_k(t') \tag{6}$$

式中:  $T_k^d$  为识别为编号  $k$  的自卸汽车的卸料时间,  $s$ ;  $B_k(t')$  为在  $t'$  时刻自卸汽车  $k$  是否在对分区内的目标活动参数,其为 1 个二元变量,在该分区内取值为 1,否则取值为 0;  $\Delta t'$  为提取视频流关键帧的时间间隔,在本文中取值为 1 s;  $t_0$  为填筑分区开始施工时

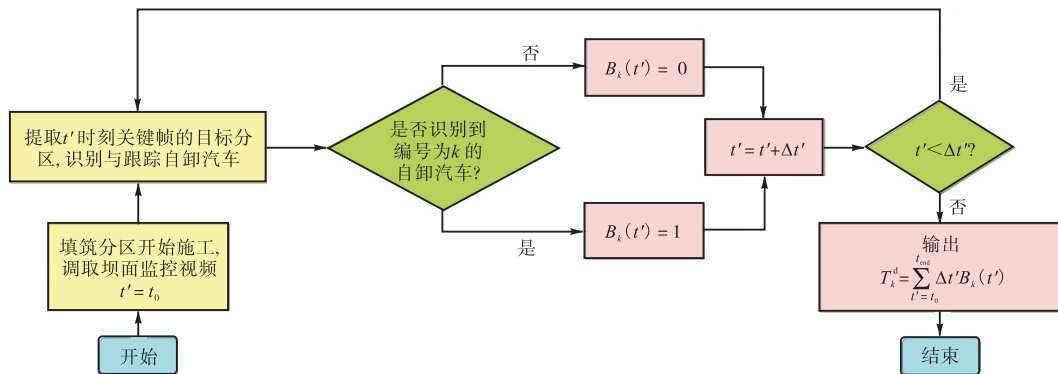


图 5 基于规则推理的卸料时间参数分析流程

Fig.5 Flow chart of unloading time parameter analysis based on rule reasoning

刻； $t_{end}$  为填筑分区完成施工时刻。

### 3 工程实例

以中国西南地区某水电站坝面填筑施工的监控视频为研究对象，通过左坝肩监控摄像头采集了多个不同日期的坝面施工监控视频，并通过人工拆帧和筛选制作成填筑分区提取和自卸汽车多目标识别的数据集。拆帧时每 30 帧保留一张图片，筛选共计 3 117 张图片作为坝面数据集。由于坝面施工监控获取的视频图像属于超大场景识别，难以识别自卸汽车的局部特征，采用摄像机获取现场施工照片作为坝面施工监控视频照片作为增强数据集，用于提高训练效果。增强数据集共采用摄像机获取 3 200 张现场施工照片以及 210 min 坝面施工监控视频，筛选共计 883 张图片作为增强数据集。因此，共计 4 000 张图片作为完整数据集进行训练，利用其中 80% 的图片构建训练集，10% 作为测试集，10% 作为验证集，只针对自卸汽车进行识别跟踪。所有实验在配置为 Intel Xeon(R) Gold 6132 CPU @ 2.60 GHz, 128 G 内存, NVIDIA Quadro RTX 8000 显卡, Ubuntu16.04 系统, 使用 Python 语言和 PyTorch 深度学习架构完成计算。

#### 3.1 基于 EVC-YOLOv8 的自卸汽车识别与分区提取结果

使用 Darklabel 2.4 工具对数据集进行标注。本案例中模型训练参数选取如下：采用  $960 \times 544$  图像大小作为网络输入；训练迭代 200 个 epoch (即完整数据集训练的遍数)；batch 大小设置为 32；梯度下降采用 SGD 算法；初始学习率为 0.01；采用余弦学习率调度器；使用 Mosaic 数据增强，部分训练样本集

如图 6 所示。单类别目标平均识别精度 (mean average precision, mAP) 的计算式为

$$I_{mAP} = \frac{1}{N'} \sum_{i_2} I_{AP_{i_2}} \quad (7)$$

式中： $N'$  为类别数； $I_{AP_{i_2}}$  为第  $i_2$  个类别的识别精度。



图 6 部分训练样本集

Fig.6 Part of training sample dataset

结果表明，训练损失函数及评估指标趋向稳定，说明模型的训练有效。同时，为证明所提 EVC-YOLOv8 模型对于小尺度目标的检测性能，在相同硬件平台中采用相同的数据集和相同训练参数进行模型性能对比实验，结果如表 1 所示。

表 1 不同目标检测模型性能对比

Tab.1 Comparison of performance among different object detection models

| 模型          | 模型预测为正例且实际为正例的样本数 $n_{TP}$ | 模型预测为正例但实际为负例的样本数 $n_{FP}$ | 模型预测为负例但实际为正例的样本数 $n_{FN}$ | 精度    | 召回率   | $F_1$ 值 | $I_{mAP}/\%$ | 训练时间/h | 单帧处理时间/ms |
|-------------|----------------------------|----------------------------|----------------------------|-------|-------|---------|--------------|--------|-----------|
| YOLOv8      | 4 672                      | 337                        | 511                        | 0.930 | 0.901 | 0.925   | 94.6         | 19.6   | 18.63     |
| CBAM-YOLOv8 | 4 751                      | 301                        | 432                        | 0.940 | 0.913 | 0.926   | 95.3         | 20.8   | 23.26     |
| EMA-YOLOv8  | 4 837                      | 246                        | 346                        | 0.952 | 0.927 | 0.939   | 95.9         | 27.3   | 34.48     |
| EVC-YOLOv8  | 4 963                      | 103                        | 220                        | 0.980 | 0.958 | 0.969   | 96.9         | 24.2   | 37.04     |

EVC-YOLOv8 模型在关键性能指标上显著优于其他模型，EVC-YOLOv8 召回率较基线模型提升了 5.7%，且明显高于经典注意力机制改进的 YOLOv8 模型，展现出了更强的小目标检测能力；尤其在大场景监控中，其能够有效减少漏检。同时， $I_{mAP}$  也表现出色，较基线模型提升了 2.3%，较 CBAM-YOLOv8 提升了 1.6%，较 EMA-YOLOv8 提升了 1%，显示出

EVC-YOLOv8 在不同 IoU 阈值下的整体检测精度有了显著提升。由于 EVC 模块和注意力机制模块的加入，导致改进的 YOLOv8 模型的训练时间均高于基线模型，但是 EVC 的轻量化设计使其训练时间低于 EMA-YOLOv8 模型，在召回率和精度上的显著提升使得这一计算开销是可以接受的，且训练时间不影响实际部署后的使用情况。由结果可知，EVC-YOLOv8

的单帧处理时间约为 37.04 ms, 即每秒可以处理 27 帧. 虽然较基线模型有所降低, 但是由于坝面超清视频监控帧数为 15 ~ 25 帧/s, 因此 EVC-YOLOv8 每秒

的处理时间能够满足仿真参数获取的实时性和精度的需求. EVC-YOLOv8 与基线 YOLOv8 部分目标识别对比结果如图 7 所示.

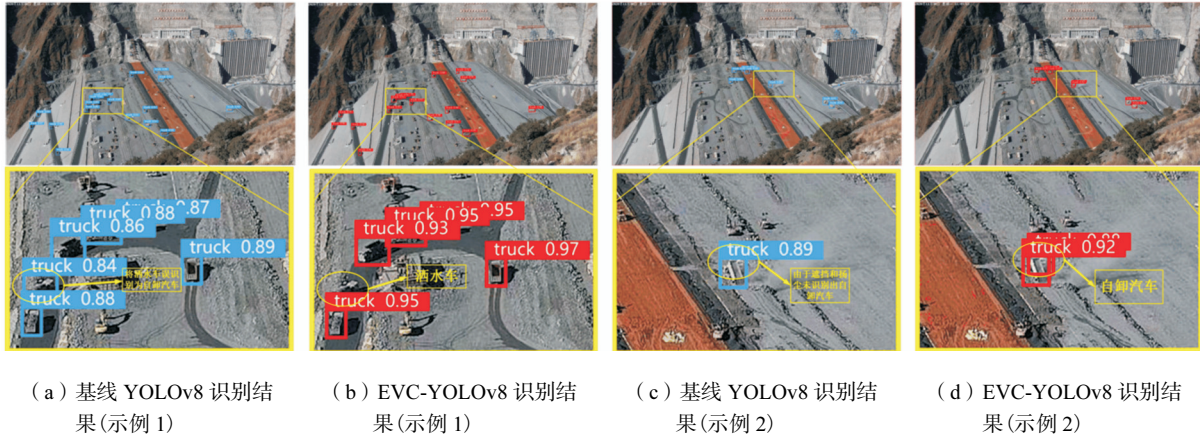


图 7 基线 YOLOv8 与 EVC-YOLOv8 部分目标识别对比结果

Fig.7 Comparison of recognition result between baseline YOLOv8 and EVC-YOLOv8 for partial objects

为验证所提方法目标识别性能提升的显著性, 在相同硬件平台、相同数据集和相同训练参数的条件下, 进行多次交叉验证, 并通过  $t$  检验来评估 EVC-YOLOv8 与基线 YOLOv8、CABM-YOLOv8 和 EMA-YOLOv8 之间在  $I_{mAP}$  上的差异是否显著. 本文使用  $K = 10$  交叉验证来确保模型评估的稳定性, 并计算了每次验证后的  $I_{mAP}$ , 如图 8 所示. 随后, 针对 EVC-YOLOv8 与基线 YOLOv8、CABM-YOLOv8 和 EMA-YOLOv8 的  $I_{mAP}$  数据, 进行样本  $t$  检验, 如表 2 所示, EVC-YOLOv8 在  $I_{mAP}$  上的提升具有显著性 ( $p$  值  $< 0.05$ ), 即 EVC-YOLOv8 的  $I_{mAP}$  相比于基线 YOLOv8 提升在统计上显著的, 说明模型改进带来的性能提升并非偶然, 而是由于 EVC 模块引入的全局和局部特征融合机制有效提升了小目标检测能力.

表 2 YOLOv8 及其 3 种改进模型  $I_{mAP}$  结果的显著性检验结果

Tab.2 Significance test results of  $I_{mAP}$  results for YOLOv8 and its three improved models

| 模型          | $t$ 值      | $p$ 值     |
|-------------|------------|-----------|
| YOLOv8      | —          | —         |
| CBAM-YOLOv8 | -1.727 744 | 0.041 152 |
| EMA-YOLOv8  | -2.887 512 | 0.009 806 |
| EVC-YOLOv8  | -4.907 009 | 0.000 114 |

采用 YOLOv8-seg 模型对图像和视频做预处理, 本案例主要提取高心墙堆石坝的心墙区作为目标识别区域, 区域提取流程示意如图 9 所示. 对于 YOLOv8-seg 模型的训练, 在坝面数据集中随机选取 1 000 张图片作为填筑分区提取的数据集, 其中 80% 的图片构建训练集, 10% 作为测试集, 10% 作为验证集, 采用 Labelimg 注释目标分区制作数据集, 进行实例分割训练.

### 3.2 基于改进 ByteTrack 的自卸汽车多目标跟踪结果

将 BytrTrack 原有的 YOLOX 检测器替换为 EVC-YOLOv8, 然后将训练好的 EVC-YOLOv8 模型权重嵌入 ByteTrack 框架中进行自卸汽车的跟踪. 为了证明基于改进的 ByteTrack 进行自卸汽车多目标跟踪模型训练的有效性, 分别统计了所采用的 ByteTrack 框架在同类数据集与在本文自卸汽车跟踪应用中多目标跟踪精度 (multiple object tracking accuracy, MOTA) 与目标识别准确度 (identification  $F_1$  score,  $IDF_1$ ) 两个常用指标.

MOTA 综合考虑了跟踪目标检测的准确度和目标编号的切换次数, 侧重于表征多目标跟踪过程中检测器的性能, 其计算式为

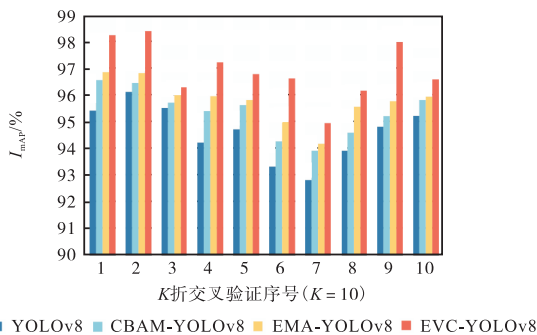


图 8 基于  $K$  折交叉验证的 EVC-YOLOv8 及对比模型在  $I_{mAP}$  上的性能比较

Fig.8 Comparison of performance between EVC-YOLOv8 and contrast models on  $I_{mAP}$  based on  $K$ -fold cross validation

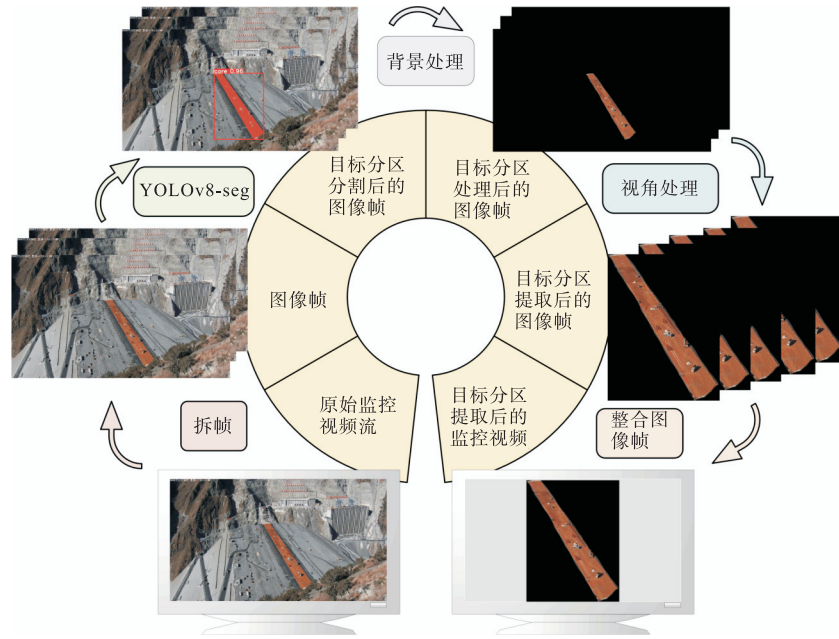


图 9 监控视频分区提取流程

Fig.9 Flow chart of monitoring video partition extraction process

$$I_{MOTA} = 1 - \frac{\sum_{i_3} (n_{FN_{i_3}} + n_{FP_{i_3}} + n_{IDSW_{i_3}})}{\sum_{i_3} n_{gt_{i_3}}} \quad (8)$$

式中： $i_3$  为视频帧索引； $n_{FN_{i_3}}$  为第  $i_3$  帧图像中自卸汽车目标漏检的数量； $n_{FP_{i_3}}$  为第  $i_3$  帧图像中自卸汽车目标误检的数量； $n_{IDSW_{i_3}}$  为第  $i_3$  帧图像中自卸汽车目标检测编号的切换次数； $n_{gt_{i_3}}$  为第  $i_3$  帧图像中真实自卸汽车目标的数量。

$IDF_1$  引入了编号信息，能够重点表征预测轨迹与真实轨迹之间的匹配程度，其计算式为

$$I_{IDF_1} = \frac{2n_{IDTP}}{2n_{IDTP} + n_{IDFP} + n_{IDFN}} \quad (9)$$

式中： $n_{IDTP}$  为正确分配编号的目标框数量； $n_{IDFP}$  为错误分配编号的目标框数量； $n_{IDFN}$  为真实标签中未被分配编号的目标框数量。

在 210 min 的坝面施工监控视频中选取 6 个视频片段共计 15 min 32 s 作为多目标跟踪算法的测试集。图 10 为测试视频片段的  $I_{MOTA}$  结果和  $I_{IDF_1}$  结果。表 3 给出了统计的改进的 ByteTrack 框架在其他多目标跟踪案例与在本文案例中的  $I_{MOTA}$  与  $I_{IDF_1}$  指标。可以看出，在本文应用场景中，由于采用了 YOLOv8-seg 图像分割模型去除了图像中背景部分的干扰信息，同时通过 EVC 模块捕获全局和局部信息，保留了小尺度目标的相关特征，所以多目标跟踪的性能结果更优。

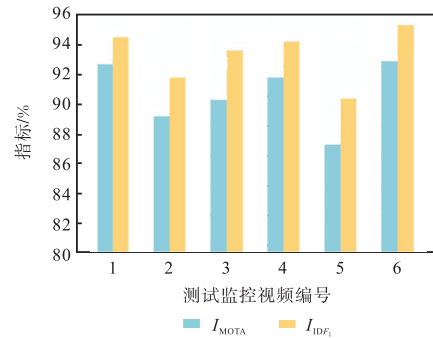


图 10 改进 ByteTrack 多目标测试集  $I_{MOTA}$  和  $I_{IDF_1}$  结果

Fig.10  $I_{MOTA}$  and  $I_{IDF_1}$  results of improved ByteTrack on multi-object test dataset

表 3 ByteTrack 多目标跟踪性能对比

Tab.3 Comparison of multi-object tracking performance using ByteTrack

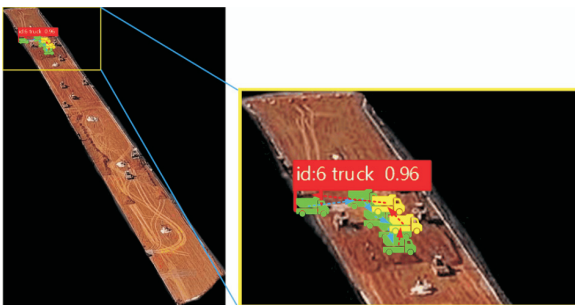
| 案例出处   | 多目标跟踪内容    | $I_{MOTA}/\%$ | $I_{IDF_1}/\%$ |
|--------|------------|---------------|----------------|
| 文献[46] | 行人/汽车多目标跟踪 | 51.1          | 53.0           |
| 文献[47] | 行人/汽车多目标跟踪 | 42.6          | 58.0           |
| 文献[48] | 汽车多目标跟踪    | 45.3          | 43.1           |
| 文献[3]  | 自卸汽车       | 87.3          | 91.9           |
| 本文     | 自卸汽车       | 90.6          | 93.2           |

跟踪器每秒处理的帧数是衡量多目标跟踪速度的评价指标<sup>[46]</sup>，改进的 ByteTrack 在测试集上的平均指标值为 25 帧/s，基线模型的平均指标值为 35 帧/s。因为添加 EVC 模块改进了 YOLOv8，导致检测器识别速度较基线有所降低。由于坝面超清视频监控帧数

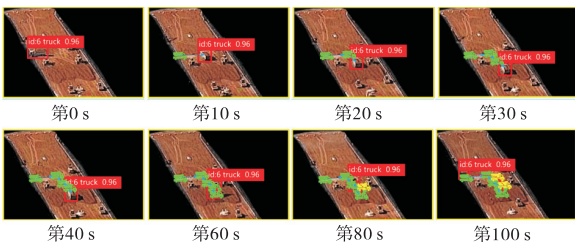
为 15 ~ 25 帧/s, 因此改进的 ByteTrack 的平均指标值能够满足仿真参数获取的实时性和精度的需求.

### 3.3 基于规则推理的卸料时间统计结果

在目标检测与跟踪结果的基础上, 基于提出的规则推理方法统计每辆自卸汽车在心墙区停留的时间作为卸料时间. 图 11 给出了某台自卸汽车从进入心墙区到离开心墙区的目标检测与跟踪结果. 图 11(a) 中左图为该自卸汽车卸料全过程轨迹示意, 右图为示意放大图, 图中绿色小车表示自卸汽车在进行卸料工序前进入填筑分区的相应时刻位置, 蓝色箭头表示自卸汽车在进行卸料工序前进入填筑分区的时刻顺序. 黄色小车表示自卸汽车在进行卸料工序后离开填筑分区的相应时刻位置, 红色箭头表示自卸汽车在进行卸料工序后离开填筑分区的时刻顺序. 图 11(b) 以放大图的形式展示该自卸汽车进行卸料活动的轨迹, 将该自卸汽车进入心墙区的时刻标记为 0 s, 以 10 s 为时间间隔进行自卸汽车轨迹时刻位置的标注, 第 40 ~ 60 s 为该自卸汽车在同一位置进行卸料的时间, 其离开心墙区的时刻为第 100 s, 则统计的 100 s 即为该自卸汽车的卸料时间.



(a) 自卸汽车跟踪轨迹示意及放大图



(b) 自卸汽车跟踪轨迹流程

图 11 某台自卸汽车跟踪轨迹

Fig.11 Tracking trajectory of one dump truck

为了进一步说明所提方法的有效性, 将第 3.2 节中 15 min 32 s 的多目标跟踪算法测试集作为卸料时间感知的测试集. 基于所提方法获取自卸汽车的卸料时间, 测试集中共计 23 辆自卸汽车进行卸料活动, 将检测到的卸料时间作为纵坐标, 自卸汽车实际的卸料时间作为横坐标, 绘制对角误差图如图 12 所示.

由于多目标检测的漏检误检可能造成部分数值较小的卸料时间检测值, 根据实际施工统计和行业经验, 自卸汽车的卸料时间几乎不可能小于 40 s.

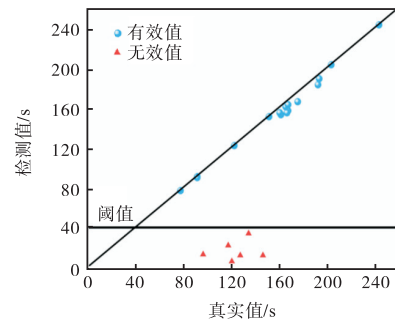


图 12 心墙区卸料时间检测值与真实值对角误差图

Fig.12 Diagonal error plot of unloading time detection values versus actual values in core zone

卸料时间受到多种因素的影响, 如作业环境、卸料距离、不同施工机械交汇、机械性能等. 通过对中国西南地区某水电站施工现场的实际数据进行分析, 发现自卸汽车的卸料时间通常在 40 s 以上, 且极少低于此阈值. 故将小于 40 s 的检测值作为无效值, 直接剔除, 如图 12 中的红色坐标点. 图 12 中蓝色坐标点为检测的卸料时间有效值, 坐标点越接近对角线, 说明检测方法的精度越高, 统计检测值与真实值的平均误差为 4.2%. 此外, 检测值与真实值统计的直方图与概率分布曲线基本一致, 说明了所提方法的有效性.

## 4 结论

针对现有基于定位技术的堆石坝施工卸料时间获取方法存在易损坏、难维护和传感器布设困难等问题, 提出采用基于监控视频的目标检测技术进行卸料时间的提取. 针对现有基于视觉的堆石坝施工卸料时间获取方法存在无法自动匹配自卸汽车位置与所属填筑分区位置, 以及大场景监控视频下小尺度自卸汽车识别精度低等问题, 提出基于深度学习及规则推理的堆石坝卸料时间实时感知模型. 本文取得的主要成果如下.

- (1) 提出卸料时间实时感知研究框架, 结合 YOLOv8-seg、EVC-YOLOv8 和 ByteTrack, 克服了 YOLOv8 在特征提取过程中难以完全捕获小尺度目标全局信息和易丢失局部特征的不足, 实现了目标分区自卸汽车的识别与跟踪, 从而结合规则推理提取了不同目标分区的卸料时间, 为堆石坝施工卸料时间的获取提供了新思路.

(2) 以中国西南地区某水电站施工现场测试视频数据进行验证, 案例分析结果表明: 所提的 EVC-YOLOv8 模型的自卸汽车单类别  $I_{mAP}$  为 96.9%, 较基准模型识别精度提高了 2.3%; 基于 YOLOv8-seg 模型的目标分区提取方法能够在 31 帧/s 的情况下实现目标分区的实时提取; 提出的改进 ByteTrack 模型在 25 帧/s 时的  $I_{MOTA}$  为 90.6%,  $I_{IDF_1}$  为 93.2%, 满足卸料时间获取的实时性和精度的需求。检测获取的卸料时间平均误差为 4.2%, 且获取的卸料时间与真实值概率分布基本一致, 验证了本模型的可行性和准确性。

本研究提出的卸料时间实时感知框架适用于固定俯视视角的监控场景(建议监控视角能够包含整个大坝坝面), 并且摄像头分辨率应大于等于 1920 像素  $\times$  1080 像素, 以保证目标的清晰度和追踪的精度。在夜间或极端天气条件下, 模型的表现可能会受到一定影响。因此, 后续研究可进一步探索如何在这些特殊条件下提高模型的适应性, 例如可考虑结合使用红外摄像头等技术以提高监控环境下的识别与跟踪性能。同时, 为了提高模型的工程应用价值和实际部署可行性, 未来研究将关注模型的轻量化和高效部署, 尤其是在边缘设备上的应用。笔者课题组计划进一步探索如何将本研究中的目标检测和跟踪模型部署在低成本嵌入式设备上, 并优化模型的延迟、计算资源占用及 GPU 显存使用, 以确保模型在资源受限的环境下也能保持较高的性能。针对目标识别误差, 计划在未来研究中引入轨迹平滑滤波技术(例如  $\alpha$ - $\beta$  滤波器), 以减少跟踪中断带来的误差并优化目标检测的精度。本模型的应用不仅为堆石坝施工现场卸料时间的获取提供了新途径, 也为其他同类工程的施工参数获取提供了新思路。

#### 参考文献:

- [1] 钟登华, 常昊天, 刘 宁, 等. 高堆石坝施工过程的仿真与优化[J]. 水利学报, 2013, 44(7): 863-872.  
Zhong Denghua, Chang Haotian, Liu Ning, et al. Simulation and optimization of high rock-filled dam construction process[J]. Journal of Hydraulic Engineering, 2013, 44(7): 863-872 (in Chinese).
- [2] 张 君, 王金国, 余 佳, 等. 多模态感知驱动下高堆石坝施工仿真参数集成深度学习模型[J]. 水利学报, 2022, 53(9): 1049-1063, 1072.  
Zhang Jun, Wang Jinguo, Yu Jia, et al. Multimodal perception-driven high rockfill dam construction simulation input modeling using an ensemble deep learning model[J]. Journal of Hydraulic Engineering, 2022, 53(9): 1049-1063, 1072 (in Chinese).
- [3] 曾拓程, 王佳俊, 王晓玲, 等. 大场景视频监控下大坝运输车改进多目标多视觉卸料识别模型研究[J]. 水利学报, 2023, 54(5): 519-529, 540.  
Zeng Tuocheng, Wang Jiajun, Wang Xiaoling, et al. Research on improved multi-target multi-vision unloading identification model of dam transport vehicle under large scene video surveillance[J]. Journal of Hydraulic Engineering, 2023, 54(5): 519-529, 540 (in Chinese).
- [4] Xie H, Fernando S, AbouRizk S. Integrating realtime project progress input into a construction simulation model[C]//Proceedings of the 2011 Winter Simulation Conference (WSC). Phoenix, USA: IEEE, 2011.
- [5] 崔 博, 胡连兴, 刘东海. 高心墙堆石坝填筑施工过程实时监控系統研发与应用[J]. 中国工程科学, 2011, 13(12): 91-96.  
Cui Bo, Hu Lianxing, Liu Donghai. Development and application of real-time monitoring system for filling construction process of high core rock-fill dam[J]. Strategic Study of CAE, 2011, 13(12): 91-96 (in Chinese).
- [6] Pradhananga N, Teizer J. Development of a cell-based simulation model for earthmoving operation using real-time location data[C]//Construction Research Congress. Atlanta, USA: ASCE, 2014.
- [7] Pradhananga N, Teizer J. Cell-based construction site simulation model for earthmoving operations using real-time equipment location data[J]. Visualization in Engineering, 2015, 3: 12.
- [8] Pradhananga N, Teizer J. Automatic spatio-temporal analysis of construction site equipment operations using GPS data[J]. Automation in Construction, 2013, 29: 107-122.
- [9] Hildreth J, Vorster M, Martinez J. Reduction of short-interval GPS data for construction operations analysis[J]. Journal of Construction Engineering and Management, 2005, 131(8): 920-927.
- [10] Vahdatikhaki F, Hammad A. Framework for near real-time simulation of earthmoving projects using location tracking technologies[J]. Automation in Construction, 2014, 42: 50-67.
- [11] Vahdatikhaki F, Setayeshgar S, Hammad A. Location-aware real-time simulation framework for earthmoving

- projects using automated machine guidance[C]//2013 Winter Simulation Conference. Washington D.C., USA: IEEE, 2013.
- [12] 钟登华, 翟海峰, 崔博, 等. 基于实时监控的沥青混凝土心墙坝坝面碾压施工仿真研究[J]. 水力发电学报, 2015, 34(7): 118-126.  
Zhong Denghua, Zhai Haifeng, Cui Bo, et al. Simulation on construction for roller compaction of dam with asphalt concrete core based on real-time monitoring[J]. Journal of Hydroelectric Engineering, 2015, 34(7): 118-126(in Chinese).
- [13] 钟登华, 刘东海, 崔博. 高心墙堆石坝碾压质量实时监控技术及应用[J]. 中国科学: 技术科学, 2011, 41(8): 1027-1034.  
Zhong Denghua, Liu Donghai, Cui Bo. Real-time compaction quality monitoring of high core rockfill dam[J]. Scientia Sinica: Technologica, 2011, 41(8): 1027-1034(in Chinese).
- [14] 刘宁. 高心墙堆石坝施工场内交通仿真与实时控制研究[D]. 天津: 天津大学建筑工程学院, 2013.  
Liu Ning. Research on Traffic Simulation and Real-Time Control for Construction of High Core Rock-Fill Dam[D]. Tianjin: School of Civil Engineering, Tianjin University, 2013(in Chinese).
- [15] Song L G, Eldin N N. Adaptive real-time tracking and simulation of heavy construction operations for look-ahead scheduling[J]. Automation in Construction, 2012, 27: 32-39.
- [16] 张兆省, 马山玉, 皇甫英杰, 等. 基于三维实景模型的水库无线视频监控系统[J]. 水力发电学报, 2019, 38(3): 179-185.  
Zhang Zhaosheng, Ma Shanyu, Huangfu Yingjie, et al. Wireless video reservoir monitoring system based on 3D real scene modeling[J]. Journal of Hydroelectric Engineering, 2019, 38(3): 179-185(in Chinese).
- [17] Zeng T C, Wang J J, Cui B, et al. The equipment detection and localization of large-scale construction jobsite by far-field construction surveillance video based on improving YOLOv3 and grey wolf optimizer improving extreme learning machine[J]. Construction and Building Materials, 2021, 291: 123268.
- [18] 康栋, 张君, 王佳俊, 等. 高心墙堆石坝大场景视频监控网络覆盖 NCHHO 优化[J]. 水力发电学报, 2022, 41(7): 47-60.  
Kang Dong, Zhang Jun, Wang Jiajun, et al. Nonlinear chaotic Harris hawks optimization model for large-scene video monitoring network coverage of high core rockfill dams[J]. Journal of Hydroelectric Engineering, 2022, 41(7): 47-60(in Chinese).
- [19] Wang D, Ren B Y, Cui B, et al. Real-time monitoring for vibration quality of fresh concrete using convolutional neural networks and IoT technology[J]. Automation in Construction, 2021, 123: 103510.
- [20] Gong J, Caldas C H. Computer vision-based video interpretation model for automated productivity analysis of construction operations[J]. Journal of Computing in Civil Engineering, 2010, 24(3): 252-263.
- [21] Kim H, Bang S, Jeong H, et al. Analyzing context and productivity of tunnel earthmoving processes using imaging and simulation[J]. Automation in Construction, 2018, 92: 188-198.
- [22] Kim H, Ham Y, Kim W, et al. Vision-based nonintrusive context documentation for earthmoving productivity simulation[J]. Automation in Construction, 2019, 102: 135-147.
- [23] Zheng Z J, Zhang Z Q, Pan W. Virtual prototyping- and transfer learning-enabled module detection for modular integrated construction[J]. Automation in Construction, 2020, 120: 103387.
- [24] Wang D, Wang X L, Ren B Y, et al. Vision-based productivity analysis of cable crane transportation using augmented reality-based synthetic image[J]. Journal of Computing in Civil Engineering, 2022, 36(1): 04021030.
- [25] Alsakka F, El-Chami I, Yu H T, et al. Computer vision-based process time data acquisition for offsite construction[J]. Automation in Construction, 2023, 149: 104803.
- [26] Wang H, Yang Q G, Liu Q, et al. Automated productivity analysis of cable crane transportation using deep learning-based multi-object tracking[J]. Automation in Construction, 2024, 166: 105644.
- [27] Kim J, Hwang J, Jeong I, et al. Generalized vision-based framework for construction productivity analysis using a standard classification system[J]. Automation in Construction, 2024, 165: 105504.
- [28] Wang X L, Li D Z, Wang J J, et al. Continuous multi-target tracking across disjoint camera views for field transport productivity analysis[J]. Automation in Construction, 2025, 171: 105984.
- [29] Zhang Y, Zhang H F, Huang Q Q, et al. DsP-YOLO: An anchor-free network with DsPAN for small

- object detection of multiscale defects[J]. *Expert Systems with Applications*, 2024, 241: 122669.
- [30] Zeng S, Yang W Z, Jiao Y Y, et al. SCA-YOLO: A new small object detection model for UAV images[J]. *The Visual Computer*, 2024, 40(3): 1787-1803.
- [31] Jiang L J, Yuan B X, Du J W, et al. MFFSODNet: Multiscale feature fusion small object detection network for UAV aerial images[J]. *IEEE Transactions on Instrumentation and Measurement*, 2024, 73: 5015214.
- [32] Dang L X, Liu G, Hou E Y, et al. YOLO-FNC: An improved method for small object detection in remote sensing images based on YOLOv7[J]. *IAENG International Journal of Computer Science*, 2024, 51(9): 1281-1290.
- [33] Liu H I, Tseng Y W, Chang K C, et al. A DeNoising FPN with transformer R-CNN for tiny object detection[J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2024, 62: 4704415.
- [34] Wang H, Liu C Y, Cai Y F, et al. YOLOv8-QSD: An improved small object detection algorithm for autonomous vehicles based on YOLOv8[J]. *IEEE Transactions on Instrumentation and Measurement*, 2024, 73: 2513916.
- [35] Ma S Z, Lu H M, Liu J, et al. LAYN: Lightweight multi-scale attention YOLOv8 network for small object detection[J]. *IEEE Access*, 2024, 12: 29294-29307.
- [36] Yue M, Zhang L Q, Huang J, et al. Lightweight and efficient tiny-object detection based on improved YOLOv8n for UAV aerial images[J]. *Drones*, 2024, 8(7): 276.
- [37] Peng R J, Liao C L, Pan W J, et al. Improved YOLOv7 for small object detection in airports: Task-oriented feature learning with Gaussian Wasserstein loss and attention mechanisms[J]. *Neurocomputing*, 2025, 634: 129844.
- [38] Dumitriu A, Tatui F, Miron F, et al. Rip current segmentation: A novel benchmark and YOLOv8 baseline results[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Vancouver, Canada: IEEE, 2023.
- [39] Lou H T, Duan X H, Guo J M, et al. DC-YOLOv8: Small-size object detection algorithm based on camera sensor[J]. *Electronics*, 2023, 12(10): 2323.
- [40] Talaat F M, ZainEldin H. An improved fire detection approach based on YOLO-v8 for smart cities[J]. *Neural Computing and Applications*, 2023, 35: 20939-20954.
- [41] Xiang X Z, Wang Z Y, Qiao Y L. An improved YOLOv5 crack detection method combined with transformer[J]. *IEEE Sensors Journal*, 2022, 22(14): 14328-14335.
- [42] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[C]//*IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Vancouver, Canada: IEEE, 2023.
- [43] Quan Y, Zhang D, Zhang L Y, et al. Centralized feature pyramid for object detection[J]. *IEEE Transactions on Image Processing*, 2023, 32: 4341-4354.
- [44] Meng T F, Zhang W Q. Fast video object segmentation via dynamic YOLACT[C]//*2022 IEEE International Conference on Acoustics, Speech and Signal Processing*. Singapore: IEEE, 2022.
- [45] Zhang Y F, Sun P Z, Jiang Y, et al. ByteTrack: Multi-object tracking by associating every detection box [C]//*European Conference on Computer Vision*. Cham, Switzerland: Springer Nature Switzerland, 2022.
- [46] Zhang Q. Multi-object trajectory extraction based on YOLOv3-DeepSort for pedestrian-vehicle interaction behavior analysis at non-signalized intersections[J]. *Multimedia Tools and Applications*, 2023, 82: 15223-15245.
- [47] Kapania S, Saini D, Goyal S, et al. Multi object tracking with UAVs using deep SORT and YOLOv3 RetinaNet detection framework[C]//*Proceedings of the 1st ACM Workshop on Autonomous and Intelligent Mobile Systems*. New York, USA: Association for Computing Machinery, 2020.
- [48] 黄凯文, 凌六一, 王成军, 等. 基于改进 YOLO 和 DeepSORT 的实时多目标跟踪算法[J]. *电子测量技术*, 2022, 45(6): 7-13.  
Huang Kaiwen, Ling Liuyi, Wang Chengjun, et al. Real-time multiple object tracking algorithm based on improved YOLO and DeepSORT[J]. *Electronic Measurement Technology*, 2022, 45(6): 7-13 (in Chinese).
- (责任编辑: 武立有)