

DOI:10.19651/j.cnki.emt.2416889

# 基于多目标跟踪优化的路口车辆轨迹提取方法<sup>\*</sup>

唐继杰<sup>1</sup> 刘金广<sup>2</sup> 欧晓放<sup>1</sup>

(1. 中国人民公安大学交通管理学院 北京 100038; 2. 公安部道路交通安全研究中心 北京 100062)

**摘要:** 为应对传统方法研究车辆轨迹存在精度和效率局限的问题,加快推进道路数字化治理模式,本文提出了一种基于多目标跟踪优化的路口车辆轨迹提取方法。首先,基于YOLOv8s算法框架,引入多分支卷积模块并设计了一种结合标准卷积与深度可分离卷积的图像处理方法,以提高模型对不同场景的鲁棒性并保持帧率稳定。然后,通过精确量化角度差异和距离损失,改进了DeepSORT算法的损失函数,以提高模型的收敛速度和处理不规则物体的准确度。最后,通过推导出像素坐标系与真实世界坐标系的转换关系,确保了车辆轨迹的准确提取。实验结果表明,改进模型较原模型mAP、召回率和MOTA分别提升了2.9%、5.6%和0.7%,编码变换次数下降64%,在检测的同时能够保持帧率稳定,能够准确提取车辆在监控录像中的轨迹信息。这对于深入研究车辆特性和道路交通风险提供了方法支撑,具有较高实战应用价值。

**关键词:** 车辆轨迹提取;实时检测;YOLOv8;DeepSORT;交通安全

**中图分类号:** TN98 **文献标识码:** A **国家标准学科分类代码:** 520.6099

## Vehicle trajectory extraction method at intersection based on multi-target tracking optimization

Tang Jijie<sup>1</sup> Liu Jinguang<sup>2</sup> Ou Xiaofang<sup>1</sup>

(1. School of Traffic Management, People's Public Security University of China, Beijing 100038, China;

2. Road Traffic Safety Research Center, Ministry of Public Security, Beijing 100062, China)

**Abstract:** To address the problems of accuracy and efficiency limitations in traditional methods of studying vehicle trajectories and accelerate the promotion of digital road traffic management, this paper proposes a vehicle trajectory extraction method at intersections based on multi-target tracking optimization. First, based on the YOLOv8s algorithm framework, a multi-branch convolution strategy was introduced and an image processing method combining standard convolution and depthwise separable convolution was designed to improve the robustness of the model to different scenes and maintain a stable frame rate. Then, the loss function of the DeepSORT algorithm is improved by accurately quantifying the angle difference and distance loss to increase the convergence speed of the model and the accuracy of handling irregular objects. Finally, the accurate extraction of vehicle trajectories is ensured by deriving the conversion relationship between the pixel coordinate system and the real-world coordinate system. The experimental results show that the improved model has improved mAP, recall rate and MOTA by 2.9%, 5.6% and 0.7% respectively compared with the original model, and the number of encoding transformations (IDS) has decreased by 64%. The frame rate can be kept stable during detection. And by deriving the conversion relationship between the pixel coordinate system and the real-world coordinate system, the vehicle's trajectory information in the surveillance video can be accurately extracted. This provides methodological support for in-depth research on vehicle characteristics and road traffic risks, and has high practical application value.

**Keywords:** vehicle trajectory extraction; real-time detection; YOLOv8; DeepSORT; traffic safety

## 0 引言

近些年来,公安部强化部署道路交通事故预防“减量控

大”工作,“努力减少事故死亡总量,全力防控较大事故”,成为道路交通安全研究的重要目标<sup>[1]</sup>。在这一背景下,为了更好地预防和减少交通事故,对交通冲突的观测<sup>[2]</sup>和分析

收稿日期:2024-09-14

\* 基金项目:国家重点研发计划(2023YFB4302701)、公安部技术研究计划项目(2022JSZ15)资助

变得尤为重要。

数据是研判道路风险和分析事故致因的基础。交通冲突数据提取经历了从人工方法逐渐向智能化技术转变的过程,王正武等<sup>[3]</sup>通过摄像机录像并使用 Tracker 软件提取车辆运动轨迹数据,取得的数据精度高且包含丰富的微观信息。任丽丽等<sup>[4]</sup>通过无人机拍摄高精度视频,结合 Kinovea 软件进行视频处理,提取出的车辆轨迹数据覆盖范围广且数据精度能够符合要求。Krajewski 等<sup>[5]</sup>通过使用无人机拍摄视频并结合 U-Net 语义分割算法和 Rauch-Tung-Striebel 平滑技术,提取车辆轨迹数据,获得的数据丰富且自然行为未受干扰。上述文献对轨迹数据的提取都取得了一定的成果。然而,随着人工智能技术的进一步发展,计算机视觉和深度学习算法凭借其在成本、覆盖范围、数据丰富性和算法性能方面的优势,逐渐取代了传统的传感器数据采集方式,成为交通冲突数据提取的主流方法。

深度学习算法应用于监控录像中车辆行驶轨迹提取主要依赖两大部分:目标检测和目标跟踪算法<sup>[6]</sup>。目标检测分为两类:单阶段检测和两阶段检测<sup>[7]</sup>。在两阶段检测中, Ren 等<sup>[8-11]</sup>提出了区域卷积神经网络(R-CNN)、快速区域卷积神经网络(Fast R-CNN)和超快区域卷积神经网络(Faster R-CNN)等检测器,这些算法的优点在于通过先生成高质量的候选区域,再进行精细分类和回归,提高了目标检测的准确性和精度。相比之下,以 YOLO 系列<sup>[12]</sup>为代表的单阶段检测器虽然精度较低,但其速度快、结构简单、便于端到端训练,非常适合实时应用和计算资源受限的场景。Wu 等<sup>[13]</sup>通过在 YOLOv3 中集成 DenseNet 并使用 K-means 聚类重新确定锚框,实现了在检测速度不变的前提下提高检测目标的精度。Jocher 等<sup>[14]</sup>通过移除 YOLOv5 中的一些卷积操作,在大多数改进模型上提升了 mAP 并且减少了模型的参数量,使模型部署更加便利和灵活。YOLOv8s 模型融合了 YOLOv7-Wang 的设计思路和 YOLOv5 的优势模块,采用解耦头结构和 Anchor-Free 策略,不仅速度快、实时性好且兼顾较高的准确率<sup>[15-17]</sup>。综合评估各算法之后,YOLOv8s 因其轻量级结构和较快的处理速度,较适合实时监控应用,能够满足实际需求。因此,本文选择 YOLOv8s 作为目标检测基础模型。根据跟踪策略的不同,目标跟踪可以分为基于检测的跟踪和基于初始框的跟踪<sup>[18]</sup>。基于初始框的跟踪(detection free tracking, DFT)不需要预训练的目标检测器,能够处理任意类型的目标,避免因检测器性能不佳导致的跟踪错误<sup>[19]</sup>,不足之处在于需要手动初始化目标的位置和特征,难以应用于实际。与此同时,Wojke 等<sup>[20]</sup>提出了基于检测的跟踪(detection-based tracking, DBT),该方法依赖目标检测器和跟踪算法的好坏,通过自动检测、处理动态目标变化和结合多种信息,提高了目标跟踪的精度和鲁棒性,在实际应用中更具优势。在目前基于检测的跟踪算法中,公平检测与重识别系统(fairness of detection and re-identification

in multiple object tracking, FairMOT)<sup>[21]</sup>复杂且训练需求高,在某些快速运动或遮挡场景下易出现跟踪不准。Bewley 等<sup>[22]</sup>提出的简单在线实时跟踪算法(simple online and realtime tracking, SORT)在高密度人群、目标遮挡频繁的场景中表现不佳,可能导致目标丢失或 ID Switch 频繁。而深度在线实时跟踪算法(deep simple online and realtime tracking, DeepSORT)在 SORT 算法的基础上进行了改进,加入了深度学习特征提取模块,能够满足快速运动目标以及多目标的跟踪需要。因此,本文选择 DeepSORT 作为目标跟踪基础算法。

现有的研究基本上都是面向车辆检测技术的改进,缺乏针对车辆轨迹的应用研究。同时,传统方法虽然能够实现轨迹提取,但需要人工标定初始模板、处理过程复杂且耗时较长<sup>[23]</sup>。此外,这些方法所依赖的设备昂贵、受环境影响大且精度有限。本文针对上述存在的问题,改进最新的检测跟踪算法,提出 YOLOv8-SN-DeepSORT(S-IoU)检测系统并将其应用于监控摄像头车辆的轨迹提取,通过融合不同坐标系间的转换关系,计算出真实路车辆的轨迹信息。实验结果表明,本文提出的方法不仅改善了车辆检测和多目标追踪的效果,还推导出了像素坐标系与真实世界坐标系的转换关系,并成功地提取了每辆车在特定时刻的位置坐标和速度信息,为深入研究车辆特性和道路交通风险提供了一种实时便捷、低成本且易于集成的轨迹数据提取方法。

## 1 研究方法

针对现有方法缺乏对车辆轨迹精确提取等问题,本文设计了一个智能识别、跟踪和轨迹转换系统。整体框架图如图 1 所示。首先,将实时或预录制的路口监控图像输入到改进的 YOLOv8 算法中进行车辆的识别和边界框标定。然后将筛选出需要跟踪车辆的运动特征输入到 DeepSORT 算法中,卡尔曼滤波器会根据目标运动特征进行目标轨迹的预测,然后利用匈牙利算法将图像每一帧中的检测框和预测轨迹进行最优匹配。在下一阶段中,错误匹配的轨迹会在卡尔曼滤波器更新阶段被过滤掉,仅保留修正后的轨迹。最后,这些轨迹将保存到文本 txt 然后进行真实坐标转换,实现视频轨迹数据的提取。

### 1.1 YOLOv8s 算法改进

为了更进一步提高 YOLOv8s 算法的精确度和小目标检测能力,本文通过改进 YOLOv8s 的结构网络,提出一种新的检测模型 YOLOv8-SN。首先,引入多元分支模块(diverse branch block, DBB)替换 YOLOv8s 的骨干网络第 8 层的跨特征融合(cross stage partial network fusion, C2f)模块。如图 2 所示, DBB 模块添加了多个分支结构,每个分支可以使用不同的卷积操作和卷积核(即滤波器)。这些分支的特征在经过深度可分离卷积和归一化处理后被融合在一起,以增强模型的特征提取能力。同时, DBB 模块在

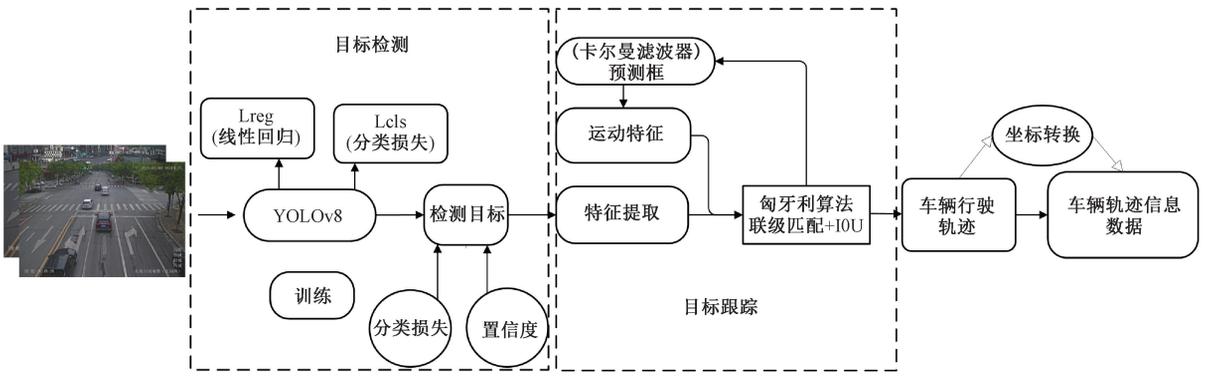


图 1 流程示意图

Fig. 1 Process diagram

推理阶段将这些复杂结构等效地转换为一个或多个标准的 2D 卷积层,以提高实际应用中的推理效率。其次,本文在 YOLOv8n 的颈部网络中采用了体素网络-高斯空间约束自池化结构 (volume of voxel networks-gaussian spatially constrained self-pooling, VoV-GSCSP) 进行特征融合,并设计了细长的颈部结构,以应对替换 DBB 模块后可能出现的帧率下降问题,如图 3 所示,通过结合上下文信息和拓展输出渠道,在降低检测器的计算复杂性和推理时间的同时,仍能保持原模型的精度。

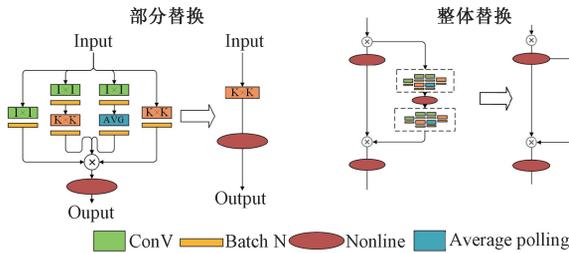


图 2 DBB 替换 C2f 模块示意图

Fig. 2 Schematic diagram of replacing C2f module with DBB module

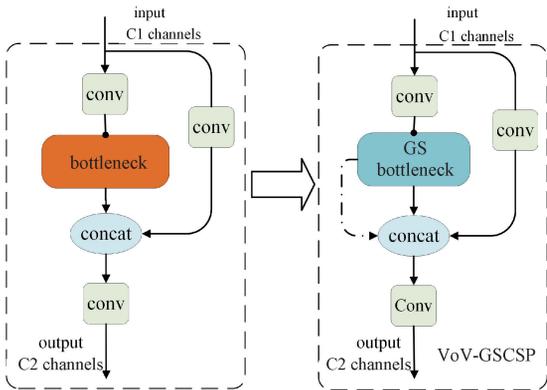


图 3 VoV-GSCSP 网络架构优化示意图

Fig. 3 VoV-GSCSP network architecture optimization diagram

如图 4 所示,在 VoV-GSCSP 中,首先对输入图像进行标准卷积的下采样。接着,应用深度可分离卷积

(depthwise convolution, DW Conv) 处理下采样后的图像。最后,将这两个卷积步骤的结果合并在一起。这种结构可以大大降低计算成本和参数数量,同时保持高效的数据处理和特征提取能力。具体地,深度可分离卷积通过分开处理空间关系和通道关系来减少计算量,而通道混洗则确保了网络各部分之间的充分信息交流,增强了特征学习能力。最终,分组卷积进一步降低了参数数量,这种结构的设计使得它较为适合硬件资源受限的环境中部署。

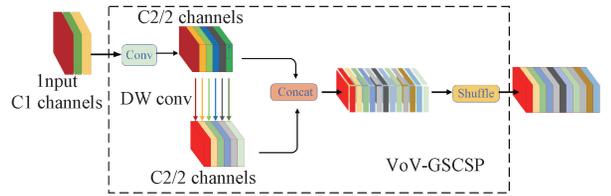


图 4 VoV-GSCSP 模块结构图

Fig. 4 VoV-GSCSP module structure diagram

### 1.2 改进 DeepSORT 对车辆进行跟踪

DeepSort 算法在目标匹配阶段采用交并比 (intersection over union, IoU) 作为关联指标。然而 IoU 依赖于边界框回归的度量集合,如预测框和地面实际框的距离、重叠面积和宽高比等,会导致收敛速度和效率难以应对监控视频中激增的目标检测数量。因此,本文提出 S-IoU 代替 IoU,以解决上述问题。S-IoU 损失函数由 4 个损失函数组成:

1) 角损失

引入角损失的初衷是为了量化描述与距离相关的变量数。如图 5 所示,  $C_n$  为实际框和预测框中心点的高度差,  $\sigma$  为真实框和预测框中心点的距离。算法通过将预测定义在 X 轴与 Y 轴上,并沿着坐标轴展开研究。当  $\alpha \leq \frac{\pi}{4}$ , 收敛过程会首先最小化  $\alpha$ , 否则,会最小化  $\beta (\beta = \frac{\pi}{2} - \alpha)$ 。当中心点在 X 轴或 Y 轴上对齐时,  $\Delta = 0$ ; 当中心点连接到 X 轴且  $45^\circ$  时,  $\Delta = 1$ 。

$$\Delta = 1 - 2 \times \sin^2 \left( \arcsin(x) - \frac{\pi}{4} \right) \quad (1)$$

式中:  $x = \frac{C_h}{\sigma} = \sin(a)$ ,  $\arcsin(x)$  等于角度。

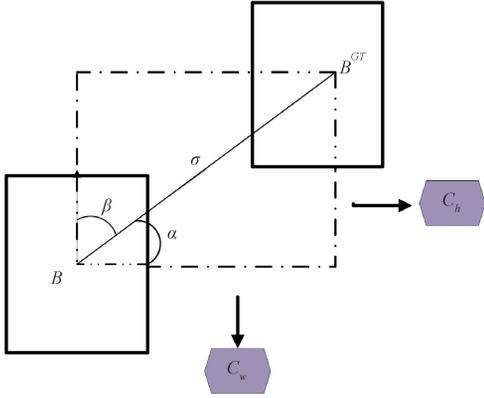


图5 中心框收敛示意图

Fig. 5 Center frame convergence diagram

## 2) 距离损失

距离公式是根据角损失来定义的:

$$\Delta = \sum_{t=x,y} (1 - e^{-\gamma \rho_t}) \quad (2)$$

$$\rho_x = \left( \frac{b_{C_x}^{gt} - b_{C_x}}{C_w} \right)^2 \quad (3)$$

$$\rho_y = \left( \frac{b_{C_y}^{gt} - b_{C_y}}{C_w} \right)^2 \quad (4)$$

$$\gamma = 2 - \Delta \quad (5)$$

式中:  $\rho$  表示两框之间中心点宽之差和二者之间最大宽之比,  $\gamma$  表示随着角度增加而被赋予时间优先的距离值。  $b_{C_x}^{gt}$ 、 $b_{C_x}$  分别表示预测框和实际框中点在坐标系中  $x$  的值,  $b_{C_y}^{gt}$ 、 $b_{C_y}$  分别表示预测框和实际框中点在坐标系中  $y$  的值,  $C_w$ 、 $C_h$  分别表示两框中点的最大长度和最大宽度。

## 3) 双边收敛

通常情况下, IoU 损失主要考虑两框整体形状的收敛, 而 S-IoU 是以长、宽两个边收敛来实现整体形状的收敛。双边收敛的定义如下:

$$\Omega = \sum_{t=w,h} (1 - e^{-W_t})^\rho \quad (6)$$

式中:  $W_w$ 、 $W_h$  分别表示预测框与实际框的最大长度和最大宽度。

这个值定义了真实框和预测框的形状损失, 并且它的值对于每个数据集都是唯一的, 如图6所示。

## 4) IoU 损失

最后根据上面3个函数, 可以定义最终的损失函数, 即:

$$L_{Box} = 1 - IoU + \frac{\Delta + \Omega}{2} \quad (7)$$

## 1.3 图像坐标转换

### 1) 监控摄像头标定

监控摄像头标定用于确定摄像头捕获的图像和实际

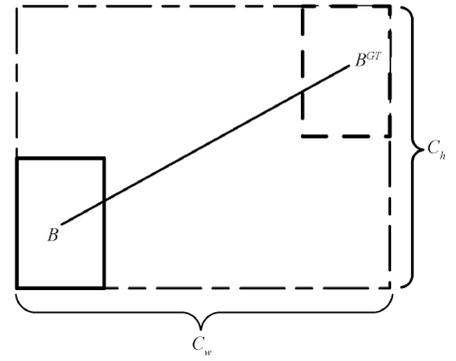


图6 边界框定位与形态损失示意图

Fig. 6 Bounding box positioning and morphological loss diagram

三维世界之间的关系。通过摄像头标定能够确定不同坐标系之间转换的具体参数, 包括监控摄像机的内参和外参。由于不同型号的监控摄像头在几何和光学特性上存在差异, 其内参数也会有所不同。针对内参数的标定方法已经广泛存在, 因此未提供具体参数。本文主要介绍摄像机外参数的标定方法, 并利用数学公式推导出图像像素坐标系与世界坐标系之间的转换关系。根据这种换算关系可以获得车辆实际运动特征, 图7展示了从不同坐标系的转换过程, 直观描述了如何通过不同坐标系将一个三维空间点映射到二维图像平面上。

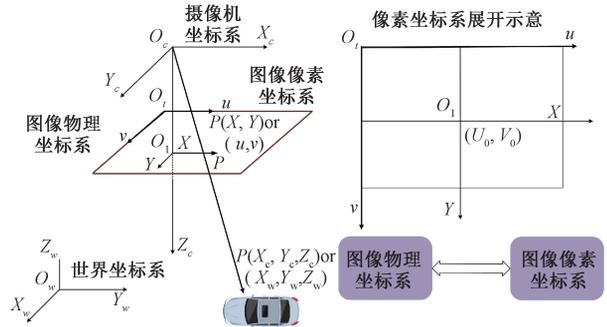


图7 坐标转换原理示意图

Fig. 7 Schematic diagram of coordinate transformation principle

### 2) 图像坐标向地面坐标转换

如图8所示, 在实际的监控摄像头标定过程中, 城市交叉口的地面范围相对有限且水平高度差可以忽略不计, 所以把它近似看作一个平面的误差相对较小。因此, 监控摄像头标定主要涉及的是图像像素坐标系与交叉口地面坐标系之间的二维坐标转换。

在监控摄像头标定中, 由于投影变换的存在, 转换矩阵非奇异, 这意味着该矩阵具有唯一的逆矩阵, 从而确保了图像平面与三维世界坐标系之间转换的唯一性。如图9所示, 本文通过梳理4个坐标系之间的转换关系, 并使用数学公式详细描述了它们的转换过程, 最终推导出图像像素坐标系与世界坐标系之间的转换关系。转换公式如下:

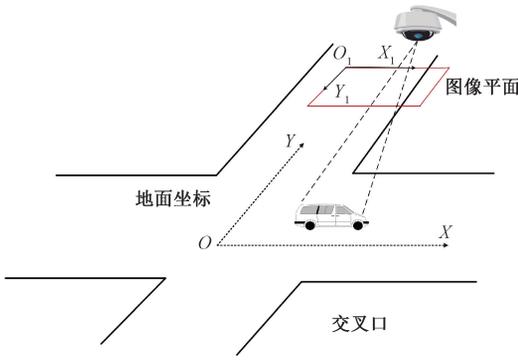


图 8 图像坐标与地面坐标

Fig. 8 Image coordinates and ground coordinates

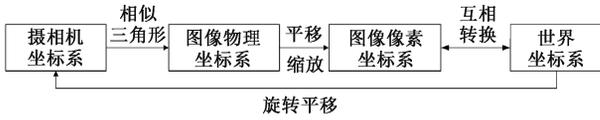


图 9 坐标系之间的关系

Fig. 9 Relationships between coordinate systems

(1) 图像物理坐标系与图像像素坐标系之间的平移关系

$$u = \frac{X}{dx} + u_0 \quad (8)$$

$$v = \frac{Y}{dy} + v_0 \quad (9)$$

式中:  $(u_0, v_0)$  表示图像像素坐标系原点,  $dx, dy$  分别表示像素平面中沿  $x, y$  轴方向每个像素大小,  $X, Y$  表示点在图像物理坐标系上值。

(2) 图像物理坐标系、摄像机坐标系和世界坐标系之间的关系

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f}{dx} & 0 & u_0 \\ 0 & \frac{f}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = K \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \quad (10)$$

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \mathbf{R} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + \mathbf{t} \quad (11)$$

式中: 摄像机坐标系某点  $P$  坐标为  $(X_c, Y_c, Z_c)^T$ , 在图像物理坐标系  $(X, Y)^T$ , 焦距为  $f$ 。点  $(X_w, Y_w, Z_w)^T$  为世界坐标系的中点,  $\mathbf{R}, \mathbf{t}$  分别为二者间的旋转矩阵、平移向量,  $K$  值为常数。

(3) 图像像素坐标系与世界坐标系的转换关系

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (12)$$

该式由式(7)~(10)综合得出。

## 2 实验结果与分析

### 2.1 实验准备

1) 数据集处理

本文采用了长沙市部分路口的监控录像以及山西某县部分路口的监控录像作为研究数据, 从 12 个路 33 段视频中截取了共计 26 889 张图像, 使用 LabelIMG 工具进行标注。将所收集的数据集按特定比例划分为训练组和测试组如表 1 所示。其中, 为了更好的提升模型对非机动车的检测精度, 在训练过程中将非机动车分为电动自行车和电动三轮车两类, 单独标注分开训练。

表 1 训练组、测试组比例

Table 1 Ratio of training group and test group

| 目标类型 | 机动车   | 非机动车  |       | 行人    |
|------|-------|-------|-------|-------|
|      |       | 电动自行车 | 电动三轮车 |       |
| 训练组  | 9 844 | 4 789 | 4 256 | 6 422 |
| 测试组  | 455   | 300   | 300   | 323   |

2) 实验环境

本文实验环境配置如表 2 所示, YOLOv8-SN 模型训练过程中参数的设置情况如下: 迭代次数设置为 300 epochs, 每轮 batch 为 32 张, 输入图像的大小为  $640 \times 640$ 。

表 2 实验环境配置

Table 2 Experimental environment configuration

|           |                                      |
|-----------|--------------------------------------|
| 操作系统      | Windows 11                           |
| 深度学习框架    | Pytorch: (version=2.2.2)             |
| CPU 型号    | 13th Intel Core i7-13650HX(2.60 GHz) |
| 显卡(GPU)型号 | NVIDIA GeForce RTX 4060 Laptop GPU   |
| CUDA 版本   | 12.1                                 |

3) 评估指标

(1) 目标检测评价指标

本文采用准确率 (precision, P)、召回率 (recall, R)、平均精度 (average precision, AP)、平均精确度 (mean average precision, mAP)、每秒处理帧数 (frames per second, FPS) 以及参数数量作为模型检测性能的评估标准。计算公式如下:

$$Precision = \frac{TP}{TP + FP} \quad (13)$$

$$Recall = \frac{TP}{TP + FN} \quad (14)$$

$$AP = \int_0^1 P(R) dR \quad (15)$$

$$mAP = \frac{1}{N} \sum_{k=1}^N R_n P_n \quad (16)$$

$$FPS = \frac{F_i}{T_n} \quad (17)$$

式中:  $P$  表示所有被预测为正的样本中, 实际为正的比例;  $R$  表示实际为正的样本中, 被正确预测为正的比例;  $AP$  指的是对于单个检测类别的准确性;  $mAP$  则代表所有类别中  $AP$  的平均准确性;  $FPS$  表示在道路目标测试集上的平均检测速度;  $F_i$  代表视频总帧数;  $T_n$  则表示检测时间。

### (2) 多目标跟踪算法评价指标

多目标跟踪效果的评估重点与其实际应用领域密切相关, 在道路交通安全的应用中更注重跟踪的准确率和实时性。编码变换次数(identity switches, IDS)表示跟踪过程中目标身份错误切换的次数。通常来说 IDS 数值越小, 代表跟踪系统性能越好。另一个评估指标是多目标跟踪准确率(multiple object tracking accuracy, MOTA), 它通过考虑漏检、误检和身份切换错误, 来测量跟踪系统的整体准确性, 公式如下:

$$MOTA = 1 - \frac{\sum_i F_n + F_p + IDS}{\sum_i GT_i} \in (-\infty, 1) \quad (18)$$

式中:  $F_n$  代表漏检率,  $F_p$  代表误检率,  $IDS$  表示跟踪过程中目标身份错误地被切换的次数,  $GT_i$  代表时间点  $t$  的实际目标数量。

MOTA 提供了一种直观的方式来评估跟踪器在检测目标和保持目标轨迹方面的性能, 而与目标检测的准确性无关。

### (3) 轨迹评价指标

在车辆轨迹提取应用中, 评估算法性能的关键在于其准确性。平均绝对误差(mean absolute error, MAE)是一种衡量预测值与真实值之间差异的指标, 能够有效量化轨迹提取的精度, 公式如下:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (19)$$

式中:  $n$  表示样本的数量,  $\hat{y}_i$  表示第  $i$  个预测值,  $y_i$  表示第  $i$  个真实值,  $|y_i - \hat{y}_i|$  表示第  $i$  个预测值和真实值之间的绝对误差。

MAE 作为衡量误差的标准, 能够帮助验证所提出的方法在真实环境中的适用性和可靠性, 确保其结果具有实际价值。

## 2.2 目标检测结果分析

### 1) 不同模型结果对比分析

本文将 Faster-CNN、YOLOv3\_wu、YOLOv5s、YOLOv7、YOLOv8s 与本文所提 YOLOv8-SN 进行对比实验,  $P$ 、 $R$ 、 $mAP$ 、 $FPS$  及模型参数量如表 3 所示。

表 3 训练组、测试组比例

Table 3 Ratio of training group and test group

| 模型         | Accuracy/%  |             |             | mAP@0.5     | Recall/%    | FPS       | Params/M    |
|------------|-------------|-------------|-------------|-------------|-------------|-----------|-------------|
|            | 机动车         | 行人          | 非机动车        |             |             |           |             |
| Faster-CNN | 67.7        | 60.7        | 67.7        | 66.2        | 81.3        | 26        | 27.0        |
| YOLOv3_wu  | 82.7        | 67.4        | 77.3        | 77.6        | 75.9        | 66        | 51.5        |
| YOLOv5s    | 88.6        | 65.9        | 70.2        | 80.3        | 77.8        | 62        | 22.5        |
| YOLOv7     | 90.8        | 67.5        | 76.2        | 82.5        | 82.1        | 77        | 36.5        |
| YOLOv8s    | 92.4        | 80.3        | 84.9        | 86.7        | 86.3        | <b>88</b> | 13.6        |
| 本文         | <b>95.1</b> | <b>82.6</b> | <b>88.6</b> | <b>91.6</b> | <b>88.9</b> | 85        | <b>10.3</b> |

通过表 3 可知, 本文介绍的 YOLOv8-SN 在路口监控视频中识别机动车、非机动车和行人方面表现良好, 平均精确度 ( $mAP$ ) 高于 YOLOv7、YOLOv3\_wu、YOLOv5s、YOLOv8s 和 Faster-CNN, 分别提高了 9.1%、14%、11.3%、4.9% 和 25.4%。它在预测框的准确性方面也优于其他模型, 有效减少漏检和误检。尽管采用 C2f-DBB 模块延长了训练时间, 其帧率虽低于 YOLOv8s, 但仍比 Faster R-CNN 快 3.2 倍, YOLOv3\_wu 快 1.2 倍, 满足实时监控需求。此外, YOLOv8-SN 的参数量极低, 只有 YOLOv3\_wu 的 1/5、不到 YOLOv7 的 1/3, 适合部署在路口摄像头等小型设备上。

为了验证模型的整体性能, 从数据集中选取了 1 378 张图像, 并基于得到的准确率和召回率数据, 绘制了不同模

型在不同类别上的准确率-召回率 ( $P$ - $R$ ) 曲线, 如图 10 所示。模型  $P$ - $R$  曲线越向上突出, 表明其性能越好。图中 Faster-CNN 在高召回率的环境中准确率大幅下降, 这意味着实际的检测中更容易存在误检和漏检, 而本文提出的模型在维持高召回率的同时, 能保持相对较高的精确率, 这意味着模型对正类样本的识别能力较强, 误检率相比其他模型更低。

此外, 图 11 分别展示了本文模型与 YOLOv3\_wu、Faster-CNN 的准确率曲线和召回率曲线的比较。从图 11 中可以看到, 改进模型在准确率在较短的训练周期内提升迅速, 并在 200 轮训练后仍有提升。从召回率曲线来看, VoV-GSCSP 的特征优化使得模型能够更快地学习到有关目标的有效信息, 使得模型在训练的初期阶段, 召回率的

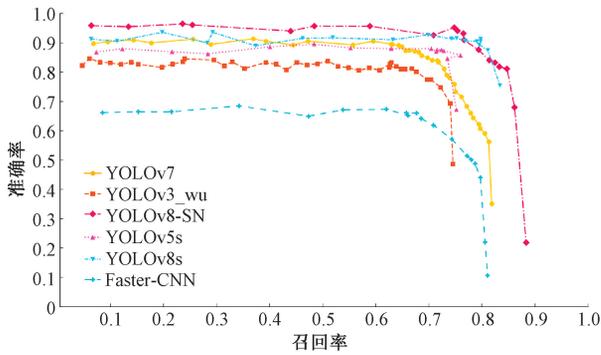
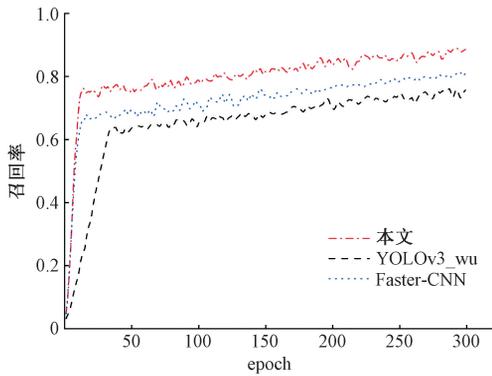


图 10 不同模型准确率-召回率曲线

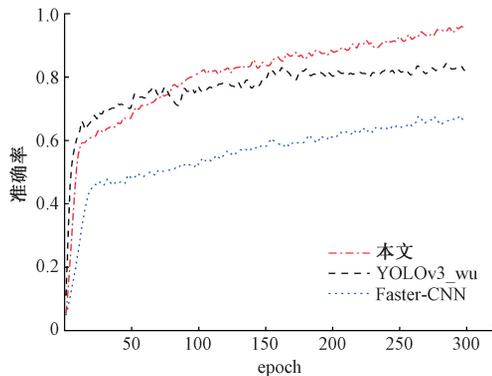
Fig. 10 Curves of precision-recall for different models

提升非常迅速。这表明模型对于各类目标有着较好的识别能力，特别是在处理复杂或变化的场景时，能够有效地减少漏检数量。



(a) 模型召回率对比曲线

(a) Recall rate comparison curve of the model



(b) 模型准确率对比曲线

(b) Precision rate comparison curve of the model

图 11 部分模型召回率和准确率对比曲线

Fig. 11 Partial model recall and accuracy comparison curves

### 2) 实际检测目标可视化验证与对比

通过可视化实际图像检测结果，能够展示模型在实际应用中的性能。如图 12 所示，本文提出的模型在处理目标遮挡和小目标检测方面展现出明显优势。在图 12 中可以看到，尽管车辆部分被树叶遮挡，模型仍能准确识别并

紧密绘制边界框，显示其优秀的定位能力。图 12 中，从楼顶高空探头捕获的小目标也被模型精确识别和锁定，证明了其在小目标检测方面的高效能力。



(a) 遮挡检测效果  
(a) Occlusion detection effect



(b) 小目标检测效果  
(b) Small target detection effect

图 12 检测效果展示图

Fig. 12 Detection effect display diagram

在图 13 的可视化分析中，本文比较了不同模型在不同天气和光照条件下的检测效果。在光照良好的情况下，由于目标类型和数量较多，大多数算法都出现了漏检和误检的问题。然而，YOLOv8-SN 算法展现了其稳定性，能够精确检测到所有目标并显著降低漏检率。在黄昏和傍晚的路口，在光照条件较差以及两边人行道的无关目标干扰下，其他算法的性能大幅下降，而 YOLOv8-SN 依靠多尺度融合特征学习技术，仍能准确识别道路上的目标。

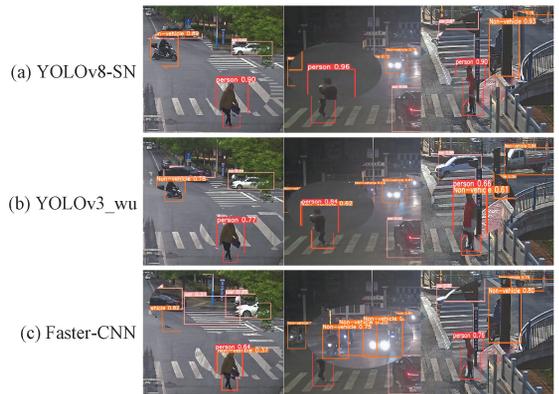


图 13 不同模型路口检测效果

Fig. 13 Detection effects of different model intersections

### 3) 消融实验

为了验证新增 C2f-DBB 模块和 VoV-GSCSP 特征融

合的改进效果,同时与已有改进 YOLOv8 算法的模块比较,本文设计了消融实验以分析并验证每个改进模块如何

影响检测准确度、mAP 和每秒帧数值。实验结果如表 4 所示,其中“√”表示对应模块的应用。

表 4 消融实验

Table 4 Ablation experiments

| YOLOv8s | C2f-DBB | VoV-GSCSP | efficientViT | C2f-Faster-EMA | Precision/% | mAP@0.5     | FPS       |
|---------|---------|-----------|--------------|----------------|-------------|-------------|-----------|
| √       |         |           |              |                | 92.4        | 86.7        | 88        |
| √       |         |           | √            |                | 94.2        | 90.3        | 86        |
| √       |         |           |              | √              | 92.7        | 88.1        | 85        |
| √       | √       |           |              |                | <b>95.6</b> | <b>94.3</b> | 59        |
| √       |         | √         |              |                | 91.4        | 87.8        | <b>89</b> |
| √       | √       | √         |              |                | 95.1        | 91.6        | 85        |

通过对实验结果分析可知,C2f-DBB 模块和 VoV-GSCSP 特征融合能够有效的提升原模型的性能。如表 4 所示,当替换 C2f-DBB 模块后,准确率和 mAP 较原模型提升了 3.4%和 8.7%,但 FPS 值却下降到 59 帧。这说明,替换 C2f-DBB 模块虽然显著增强了对物体的检测能力,但是复杂化了训练过程从而导致帧率下降。而融合 VoV-GSCSP 特征结构后降低了原来模型推算和网络结构的复杂性,同时保持原有的准确率和 mAP 不受太大的影响。从表可以看出,在同时替换 C2f-DBB 模块和融合 VoV-GSCSP 特征之后,改进模型较原模型准确率和召回率提升了 2.9%和 5.6%,同时帧率保持稳定。与其他改进模块相比,C2f-

Faster-EMA 模块<sup>[24]</sup>虽然可以减小训练过程中的噪声以及提高模型的稳定性,但是相比于原模型精度的提升并不明显,efficientViT 结构<sup>[25]</sup>的改进模型准确率和 mAP 于相比于原模型提高了 4.1%,但提升效果与本文模型相比低 52.8%。

### 2.3 目标跟踪结果分析与应用

#### 1) 目标跟踪结果分析

为了验证本文提出的改进 DeepSort 算法的性能,对该算法的不同改进模块进行了消融实验。通过使用身份切换次数(ID switch)和多目标跟踪准确率(MOTA)作为评价标准,分析这些模块对算法跟踪能力和检测效果的影响。实验结果如表 5 所示。

表 5 多目标跟踪对比结果

Table 5 Multi-target tracking comparison results

| 模型      |         |           | 模块       |       | ID switch | MOTA/%      |
|---------|---------|-----------|----------|-------|-----------|-------------|
| YOLOv5s | YOLOv8s | YOLOv8-SN | DeepSort | S-IoU |           |             |
| √       |         |           | √        |       | 77        | 73.2        |
| √       |         |           | √        | √     | 30        | 73.9        |
|         | √       |           | √        |       | 28        | 78.7        |
|         | √       |           | √        | √     | 16        | 79.1        |
|         |         | √         | √        |       | 14        | 80.9        |
|         |         | √         | √        | √     | <b>6</b>  | <b>81.6</b> |

表 5 对比了不同模型在多目标跟踪性能上的表现。通过数据可以看出,采用 YOLOv8-SN-DeepSORT(S-IoU)系统后,多目标跟踪的准确率显著增强。原有的 YOLOv5s-DeepSORT、YOLOv8s-DeepSORT 和 YOLOv8-SN-DeepSORT 系统的 MOTA 分别为 73.2%、78.7%和 80.9%,经过 S-IoU 优化后,各自增加了 0.7%、0.4%和 0.7%。此外,从编码变换次数来看,YOLOv8-SN-DeepSORT(S-IoU)系统在替换 S-IoU 损失函数以后跟踪的稳定性有显著的提升,IDS 从 14 次减少至 6 次,下降了 64%。因此综合二者来看,这些改进措施提高了路口多目标跟踪的效率与准确性。

#### 2) 实际路口可视化验证

为了呈现改进跟踪算法的效果,本文将该算法应用于路口监控视频,并使用自建路口监控数据集来追踪车辆的移动路径。如图 14 所示,该主控界面是一个基于 Qt 的图形用户界面(graphical user interface, GUI)应用程序,由 Python 编写。使用这个界面,监控操作员能够实时查看或回顾历史数据,从而准确获知车辆的运动轨迹。

图 15 展示了雨天和傍晚时,路口监控视频中的车辆跟踪效果。图 15(a)展示了在良好光照条件下,系统成功检测并在车辆移动过程中持续精确地跟踪和维持编号。图 15(b)则展示了在光线较弱且图像质量不佳的条件下,



图 14 主控界面检测图

Fig. 14 Main control interface detection diagram

该算法依旧能够准确识别并稳定跟踪所有车辆,且编号保持一致。这表明本文算法在不同光照和天气条件下的鲁棒性和可靠性。



(a) 雨天检测效果  
(a) Rainy day detection effect



(b) 夜间检测效果  
(b) Nighttime detection effect

图 15 不同环境检测效果图

Fig. 15 Different environment detection effect diagram

## 2.4 轨迹分析与坐标转换

### 1) 轨迹信息提取

在路口监控录像中,使用目标检测跟踪系统可以获取有关车辆的位置信息,图 16 为非真实场景轨迹示意图,通过记录特定编号目标在不同时间点的  $x/\text{pixel}$ (沿  $x$  轴的像素位置)和  $y/\text{pixel}$ (沿  $y$  轴的像素位置),可以获得该目标的移动轨迹信息。



图 16 车辆轨迹跟踪示意图

Fig. 16 Vehicle trajectory tracking schematic diagram

如图 17 所示,本文将使用 YOLOv8-SN-DeepSORT (S-IoU) 系统来实现多车辆目标的轨迹追踪。并基于此方法收集像素坐标系的车辆行驶轨迹数据,如表 6 所示。



图 17 数据提取区域图示

Fig. 17 Data extraction area diagram

表 6 轨迹信息提取表

Table 6 Trajectory information extraction table

| 时间/s | 目标编号(ID) | $\hat{x}/\text{px}$ | $x/\text{px}$ | $\hat{y}/\text{px}$ | $y/\text{px}$ | 目标类型 |
|------|----------|---------------------|---------------|---------------------|---------------|------|
| 0.02 | 1        | 506                 | 509           | 357                 | 360           | 机动车  |
| 0.02 | 2        | 581                 | 583           | 289                 | 294           | 机动车  |
| 0.02 | 3        | 601                 | 604           | 244                 | 248           | 机动车  |
| ⋮    | ⋮        | ⋮                   | ⋮             | ⋮                   | ⋮             | ⋮    |
| 6.14 | 16       | 466                 | 471           | 366                 | 370           | 机动车  |
| 6.14 | 17       | 532                 | 536           | 512                 | 516           | 机动车  |
| 6.14 | 18       | 598                 | 604           | 423                 | 427           | 机动车  |
| ⋮    | ⋮        | ⋮                   | ⋮             | ⋮                   | ⋮             | ⋮    |

通过计算每帧图像中跟踪目标的像素坐标预测值与真实值之间的差异,本文可以得出像素坐标上  $x$  和  $y$  方向的平均绝对误差。

根据表 6 的数据,实验结果显示  $x$  方向的 MAE 为 3 像素, $y$  方向的 MAE 为 5 像素。由于视频中每辆车的宽

度大约为 120~150 像素,因此,这些不到 5% 的相对轨迹误差在应用场景中是可以接受的。

### 2) 图像坐标转换

表 6 中的像素坐标经过转换后,可以得到车辆在世界坐标系下的位置和速度信息,具体轨迹数据如表 7 所示。

表7 车辆轨迹数据样例

Table 7 Trajectory sample vehicle trajectory data

| 时间/s | 编号 | $x/\text{px}$ | $y/\text{px}$ | $X_w/\text{m}$ | $Y_w/\text{m}$ | $V_x/(\text{m}\cdot\text{s}^{-1})$ | $V_y/(\text{m}\cdot\text{s}^{-1})$ |
|------|----|---------------|---------------|----------------|----------------|------------------------------------|------------------------------------|
| 0.02 | 1  | 506           | 357           | -3.61          | 12.03          | 3.122                              | 8.667                              |
| 0.02 | 2  | 581           | 289           | -5.82          | 15.22          | 1.233                              | 9.845                              |
| 0.02 | 3  | 601           | 244           | -6.40          | 17.33          | 2.243                              | 13.782                             |
| ⋮    | ⋮  | ⋮             | ⋮             | ⋮              | ⋮              | ⋮                                  | ⋮                                  |
| 6.14 | 16 | 466           | 366           | -2.43          | 11.60          | 3.344                              | 11.678                             |
| 6.14 | 17 | 532           | 512           | -4.37          | 4.75           | 5.420                              | 12.456                             |
| 6.14 | 18 | 598           | 423           | -6.28          | 8.92           | 2.435                              | 10.192                             |
| ⋮    | ⋮  | ⋮             | ⋮             | ⋮              | ⋮              | ⋮                                  | ⋮                                  |

本文通过观察(沿Y方向的速度)的正负值来判断车辆的行驶方向。在研究区域内,车辆向摄像头方向前进,Y坐标逐渐增大,对应的y值为正。

通过实验结果证明,本文改进的模型和算法有效,能够提升精度且可以提取出目标位置移动信息并成功转换为目标轨迹信息,提出的方法满足交通冲突研究对数据的要求,具有实际应用价值。

### 3 结 论

本文基于改进的多目标跟踪算法系统,结合监控摄像头参数,建立水平和垂直视场的投影面,从而推导出像素与世界坐标系之间的转换关系,进而形成车辆行驶轨迹数据的提取方法。基于YOLOv8s模型,本文引入了多分支卷积模块并设计了一种新的特征融合机制,这些改进在不影响帧数的情况下显著提升了对小尺寸目标和遮挡物的检测精度。在跟踪阶段,通过量化描述与距离相关的变量数,提高了DeepSORT模型的收敛速度和检测效率。最后通过得到的目标移动轨迹信息,利用几何平面转换方法,将移动轨迹信息转化为实际轨迹数据集。通过使用路口数据集检测及跟踪数据集进行测试,实验结果表明:本文改进的算法相较于原模型在mAP、召回率、MOTA上分别提升了2.9%、5.6%和0.7%,编码变换次数(IDS)降低了64%,提出的路口车辆轨迹提取方法能够有效提取道路上机动车的轨迹信息。后续研究将会考虑提取行人和非机动车的轨迹,同时结合实地调研的数据作进一步验证。

### 参考文献

- [1] 公安部部署2021年道路交通事故预防“减量控大”工作[J]. 道路交通管理, 2021(3): 6.  
The Ministry of Public Security deploys the “reduce and control” road traffic accident prevention work in 2021[J]. Road Traffic Management, 2021(3): 6.
- [2] 郭延永. 基于交通冲突理论的信号交叉口安全评价技术[D]. 南京: 东南大学, 2016.  
GUO Y Y. Signalized intersection safety evaluation techniques based on traffic conflict theory [D].

Nanjing: Southeast University, 2016.

- [3] 王正武, 谢静怡, 王杰, 等. 基于轨迹数据的非机动车左转信号灯安全效应评估[J]. 长沙理工大学学报(自然科学版), 2024, 21(1): 164-173.  
WANG ZH W, XIE J Y, WANG J, et al. Traffic safety evaluation of left-turning non-motor vehicle signal lights at intersections based on trajectory data [J]. Journal of Changsha University of Science & Technology (Natural Science), 2024, 21(1): 164-173.
- [4] 任丽丽, 吴江玲, 郭旭亮, 等. 无信号环形交叉口机非冲突机器学习预测方法[J]. 科学技术与工程, 2023, 23(31): 13592-13600.  
REN L L, WU J L, GUO X L, et al. Traffic conflict prediction of motorized and non-motorized vehicles at unsignalized roundabouts using machine learning [J]. Science Technology and Engineering, 2023, 23(31): 13592-13600.
- [5] KRAJEWSKI R, BOCK J, KLOEKER L, et al. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems [C]. 2018 21st International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2018: 2118-2125.
- [6] 丁玲, 缪小然, 胡建峰, 等. 改进YOLOv8s与DeepSORT的矿工帽带检测及人员跟踪[J]. 计算机工程与应用, 2024, 60(5): 328-335.  
DING L, MIAO X R, HU J F, et al. Improved miner chin strap detection and personnel tracking with YOLOv8s and DeepSORT [J]. Computer Engineering and Applications, 2024, 60(5): 328-335.
- [7] JIAO L CH, ZHANG F, LIU F, et al. A survey of deep learning-based object detection [J]. IEEE Access, 2019, 7: 128837-128868.
- [8] REN SH Q, HE K M, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks [J]. IEEE Transactions on

- Pattern Analysis and Machine Intelligence, 2016, 39(6): 1137-1149.
- [9] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2014: 580-587.
- [10] GIRSHICK R. Fast R-CNN[C]. IEEE International Conference on Computer Vision, 2015: 1440-1448.
- [11] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2016: 779-788.
- [12] WU X W, SAHOO D, HOI S C H. Recent advances in deep learning for object detection[J]. Neurocomputing, 2020, 396: 39-64.
- [13] WU F, JIN G Q, GAO M Y, et al. Helmet detection based on improved YOLOv3 deep model[C]. 2019 IEEE 16th International Conference on Networking, Sensing and Control(ICNSC), IEEE, 2019: 363-368.
- [14] JOCHER G, STOKEN A, BOROVEC J, et al. Ultralytics/YOLOv5: v3.1-bug fixes and performance improvements[J]. Zenodo, 2020, DOI: 10.5281/ZENODO.4154370.
- [15] 王嘉琳. 基于 YOLOv5 和 DeepSORT 的多目标跟踪算法研究与应用[D]. 济南: 山东大学, 2021.  
WANG J L. Research and application of multi-target tracking algorithm based on YOLOv5 and DeepSORT[D]. Jinan: Shandong University, 2021.
- [16] LUO W H, XING J L, MILAN A, et al. Multiple object tracking: A literature review[J]. Artificial Intelligence, 2021, 293: 103448.
- [17] 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, 2023: 7464-7475.
- [18] 宋绍京, 陆婷婷, 孙翔, 等. 面向自动驾驶的多任务环境感知算法[J]. 电子测量技术, 2023, 46(24): 157-163.  
SONG SH J, LU T T, SUN X, et al. Multi-task environment sensing algorithms for autonomous driving[J]. Electronic Measurement Technology, 2023, 46(24): 157-163.
- [19] 王夏黎. 视频交通流检测及车辆识别系统的设计与实现[D]. 西安: 西北大学, 2001.  
WANG X L. Design and implementation of video traffic flow detection and vehicle recognition system[D]. Xi'an: Northwest University, 2001.
- [20] WOJKE N, BEWLEY A, PAULUS D. Simple online and realtime tracking with a deep association metric[C]. 2017 IEEE International Conference on Image Processing(ICIP), IEEE, 2017: 3645-3649.
- [21] ZHANG Y F, WANG CH Y, WANG X G, et al. FairMOT: On the fairness of detection and re-identification in multiple object tracking [J]. International Journal of Computer Vision, 2021, 129: 3069-3087.
- [22] BEWLEY A, GE Z Y, OTT L, et al. Simple online and realtime tracking[C]. 2016 IEEE International Conference on Image Processing(ICIP), IEEE, 2016: 3464-3468.
- [23] ZHANG SH F, CHI CH, YAO Y Q, et al. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 9759-9768.
- [24] LIU X Y, PENG H W, ZHENG N X, et al. Efficientvit: Memory efficient vision transformer with cascaded group attention[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023: 14420-14430.
- [25] ZHU J, HU T, ZHENG L H, et al. YOLOv8-C2f-Faster-EMA: An improved underwater trash detection model based on YOLOv8 [J]. Sensors, 2024, 24(8): 2483.

### 作者简介

唐继杰, 硕士研究生, 主要研究方向为智能交通、目标检测。

刘金广(通信作者), 博士, 研究员, 主要研究方向为道路交通安全、道路交通组织管控。

E-mail: jgliu010@163.com

欧晓放, 硕士研究生, 主要研究方向为智能交通、道路交通安全。