

多目标跟踪技术在隧道复杂交通环境中的应用

杨 焙¹, 卢春静², 陈正池², 陈 双^{3*}

¹贵州中南交通科技有限公司技术研发中心, 贵州 贵阳

²道坦科技股份有限公司智能交通事业部, 贵州 贵阳

³贵州理工学院大数据学院, 贵州 贵阳

收稿日期: 2026年5月12日; 录用日期: 2026年6月16日; 发布日期: 2026年6月26日

摘 要

针对隧道复杂交通环境中多目标车辆跟踪存在的遮挡频繁、光照骤变等问题, 本文对, 基于YOLOv5 + ByteTrack的多目标跟踪技术。通过分析隧道环境特性, 设计了包含目标检测与跟踪关联的算法框架, 采用YOLOv5作为目标检测器, ByteTrack作为跟踪算法。实验结果表明, 在遮挡与车流密度极端情况下, YOLOv5 + ByteTrack的ID保持率(95.7%)显著高于YOLOv5 + DeepSORT (71.4%); 在车辆速度变化与运动模糊影响下, ByteTrack算法对图像模糊的依赖性更小, 追踪连续性表现更优。该技术为隧道交通监控提供了有效解决方案, 对提升隧道交通安全监测与异常行为识别能力、推动隧道交通向自动化、智能化管控转型具有重要现实意义。

关键词

隧道交通环境, 多目标跟踪, YOLOv5, ByteTrack, ID保持率

Application of Multi Object Tracking Technology in Complex Traffic Environment of Tunnels

Bei Yang¹, Chunjing Lu², Zhengchi Chen², Shuang Chen^{3*}

¹Guizhou Zhongnan Transportation Technology Co., Ltd. Technology R&D Center, Guiyang Guizhou

²Daotantan Technology Co., Ltd. Intelligent Transportation Division, Guiyang Guizhou

³School of Big Data, Guizhou University of Technology, Guiyang Guizhou

Received: May 12, 2026; accepted: June 16, 2026; published: June 26, 2026

*通讯作者。

文章引用: 杨焙, 卢春静, 陈正池, 陈双. 多目标跟踪技术在隧道复杂交通环境中的应用[J]. 计算机科学与应用, 2026, 16(6): 191-201. DOI: 10.12677/csa.2026.166219

Abstract

To address issues such as frequent occlusions and sudden changes in illumination in multi-object vehicle tracking within complex tunnel traffic environments, this paper focuses on multi-object tracking technology based on YOLOv5 ByteTrack. By analysing the characteristics of tunnel environments, an algorithm framework incorporating object detection and tracking association was designed, using YOLOv5 as the object detector and ByteTrack as the tracking algorithm. Experimental results show that under extreme conditions of occlusion and traffic density, the ID retention rate of YOLOv5 ByteTrack (95.7%) is significantly higher than that of YOLOv5 DeepSORT (71.4%); under the influence of vehicle speed changes and motion blur, the ByteTrack algorithm is less dependent on image clarity and exhibits better tracking continuity. This technology provides an effective solution for tunnel traffic monitoring and is of significant practical importance for enhancing tunnel traffic safety monitoring, recognising abnormal behaviours, and promoting the transition of tunnel traffic towards automated and intelligent management.

Keywords

Tunnel Traffic Environment, Multi-Object Tracking, YOLOv5, ByteTrack, ID Retention Rate

Copyright © 2026 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

城市化发展促进了交通运输需求量，智能交通系统(ITS)在交通管控中的作用也越发重要，复杂交通场景对精准监控、高效预警及自动化管控提出了更高的要求。在西南地区高速公路隧道在公路占比中是高占比的存在。因空间封闭、光照骤变、车辆遮挡频繁、视野受限等固有特性，是 ITS 落地的难点场景之一。隧道出入口存在明暗交替的剧烈光照波动，目标漏检率为 15%至 20%。车辆密集连续被遮挡时，追踪丢失率甚至超过 30%。

传统监控系统与常规算法在隧道复杂环境中表现不佳，检测与跟踪误差频发。而实时、精准的多目标车辆检测与追踪技术，是突破隧道监控瓶颈的核心。其能够有效提升隧道内交通安全监测与异常行为识别能力，面对突发情况及时预警故障、拥堵等。同时，提供交通流优化、应急救援调度数据支撑，进而提升通行效率、强化安全预警能力，推动隧道交通向自动化、智能化管控转型，对完善智能交通体系、保障道路通行安全具有重要现实意义。

2. 多目标跟踪现状

2.1. 交通多目标跟踪研究进展

国内研究侧重于多模态数据融合以及特征学习的优化。研究者赵敏等人利用扩展后的 VeRi 数据集作为车辆重识别的预训练模型[1]，并结合中心损失和交叉熵损失的联合优化方法，显著提升了在遮挡场景下的特征区分能力。而且在标准测试集上，IDF1 指标提高了 1%，MOTA 增益约 0.8%。该研究是对遮挡场景的针对性优化与隧道环境频繁遮挡的特点高度契合，其采用的联合损失函数为本文的检测算法在检测目标遮挡条件下提升 ID 保持率提供了重要参考。研究者着眼于隧道场景光照多变、遮挡频繁等复杂环境提出视觉惯性与毫米波雷达三级融合架构[2]。该架构核心采用改进的卡尔曼滤波算法，通过建立多

传感器时间映射机制。实现视觉图像、惯性测量数据与毫米波雷达信号在时间维度的精准同步。

国外车辆追踪研究逐渐从传统视觉多目标跟踪转向多源数据融合与智能感知系统协同的方向发展。转变的核心在于通过多模态传感器数据的互补性提升复杂场景下的跟踪性能。研究者发现单纯依赖视频数据在光照条件不足时存在显著局限性，而激光雷达与毫米波雷达的深度信息能够有效弥补此缺陷。因此基于摄像头与激光雷达融合的目标检测与跟踪方法成为研究热点，尤其在自动驾驶领域。相较于本文采用的单模态视觉方案，多传感器融合在极端光照和恶劣天气下环境适应性能力更强，但其系统复杂度、硬件成本及标定难度也显著增加。本文旨在验证轻量化、易部署的视觉方案在隧道环境中的性能上限，为后续多传感器融合方案提供可对比的视觉基线。

2.2. 隧道交通环境下的监控与跟踪现状

高速公路隧道内部的车辆检测及车辆追踪始终是技术难点，昏暗且不规则的照明变化、水雾以及车辆遮挡等因素导致传统检测方法的准确率大幅下降。普通摄像头在强逆光或雾气弥漫时容易漏检或误判车辆。研究者对 YOLO 等算法进行了针对性优化[3]。通过结合图像增强技术和坐标注意力机制，在隧道的实际测试中夜间车辆检测的准确率提高了 12%。多传感器融合成为结构健康监测方面的主流方案，设备能在极低能见度环境下通过声波反射识别隧道环境是否破损。新型巡检机器人不仅能拍摄高清图像还能实时分析环境数据。当机器人检测到抛洒物或违停车辆时能立即通过广播发出警告，其效率远超传统人工巡检。

3. 隧道复杂交通环境分析

交通运输部统计显示截至 2023 年底我国高速公路总里程达 18.36 万公里[4]。截至 2022 年底全国已建成公路隧道 24850 座总延长 2678.43 万米其中特长隧道 1752 座，总长 795.11 万米。高速公路隧道在几何结构和工程设计方面都普遍表现为车道窄、交通封闭、单向行驶的特点[5]。在隧道环境中(图 1)，车辆在隧道内发生故障的几率比较大，容易导致比较严重的事故发生。并且因为空间原因，开展救援工作困难。对交通运输系统的实时监控、应急响应能力提出了更高要求，也加大了隧道管理的技术难度与工作压力。



Figure 1. Tunnel environmental conditions
图 1. 隧道环境情况

隧道内光线因车灯和信号灯忽明忽暗，相邻车辆贴得太近检测算法在进行检测时检测的目标框容易粘连。以及隧道内部摄像头常年暴露在空气中，积累一定的灰尘，这样会导致部分摄像头无法及时清晰查看隧道内部的情况。

4. 技术架构

4.1. 算法总体框架

从输入图片数据开始到检测阶段输出所有目标检测框，并将其按置信度划分为高置信目标和低置信目标，使用 IOU 距离，在高置信检测框与活跃轨迹之间进行第一次数据关联，将剩余轨迹与低置信检测框进行 IOU 匹配，恢复可能遗漏的轨迹。并更新成功匹配的轨迹状态，如果未匹配轨迹进入“失活”状态，一段时间未恢复即终止。算法流程图如图 2 所示。

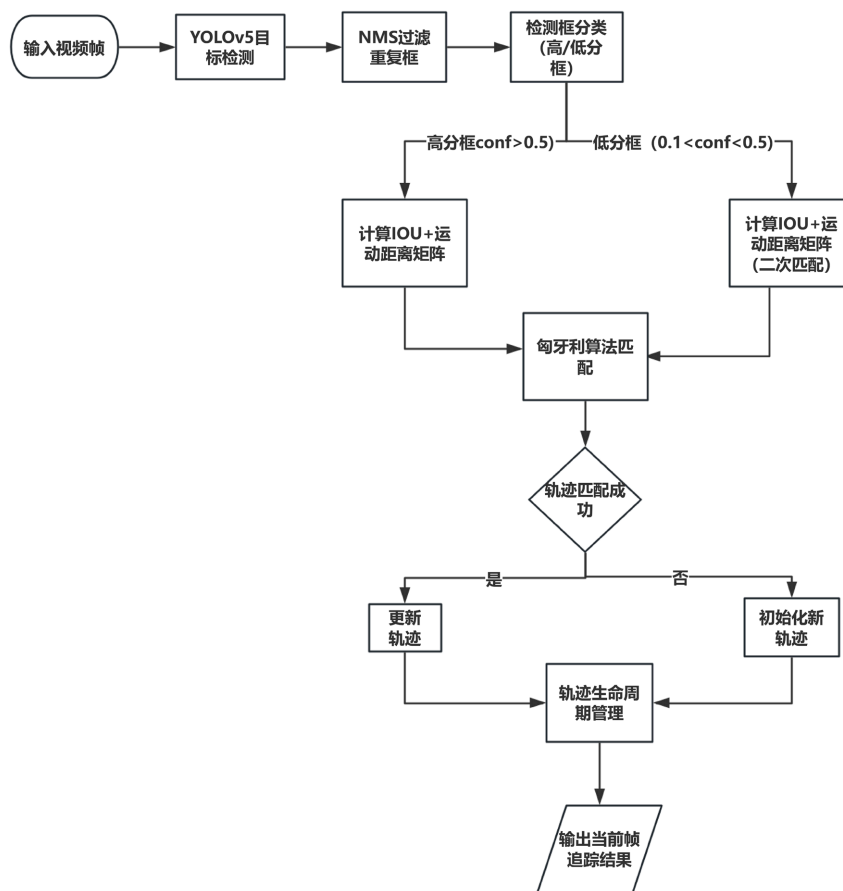


Figure 2. Overall algorithm structure diagram
图 2. 算法总体结构图

4.2. 目标检测模块

YOLOv5 是 Ultralytics 公司在 2020 年推出的开源轻量化目标检测算法[6]。YOLOv5 的核心思想在于将物体检测任务视为一个回归问题，即通过神经网络从图像直接预测边界框及其对应类别，同时模型小、能够达到实时的效果。YOLOv5 的模型结构是输入数据从主干、特征融合模块、检测头依次按照顺序输出，如下图 3 所示。

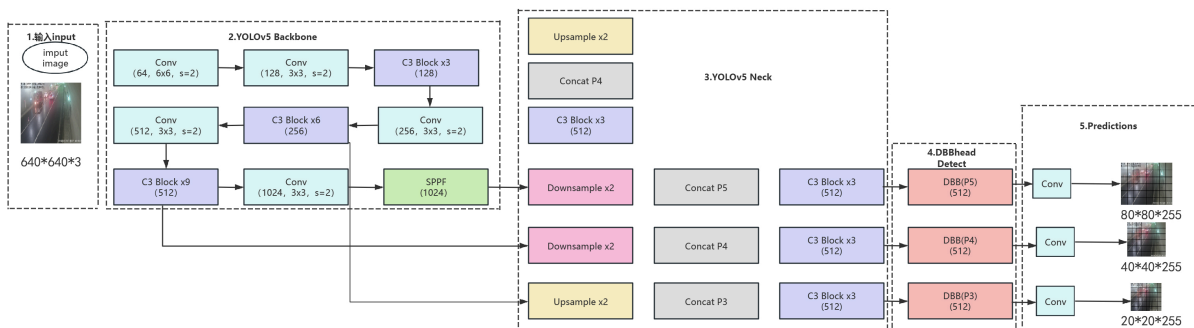


Figure 3. YOLOv5 algorithm architecture diagram
图 3. YOLOv5 算法架构图

4.3. 跟踪与关联方法

ByteTrack 算法是一种基于目标检测的追踪算法，如下图 4 所示，和其他非 ReID 的算法一样，使用目标追踪所得到的 bbox 进行追踪[7]。追踪算法使用卡尔曼滤波预测边界框，然后使用匈牙利算法进行目标和轨迹间的匹配。ByteTrack 算法创新的处理了对低分框的使用，低分框可能是对物体遮挡时产生的框，直接对低分框抛弃会影响性能，使用低分框对追踪算法进行了二次匹配，有效优化了追踪过程中因为遮挡造成换 ID 的问题。

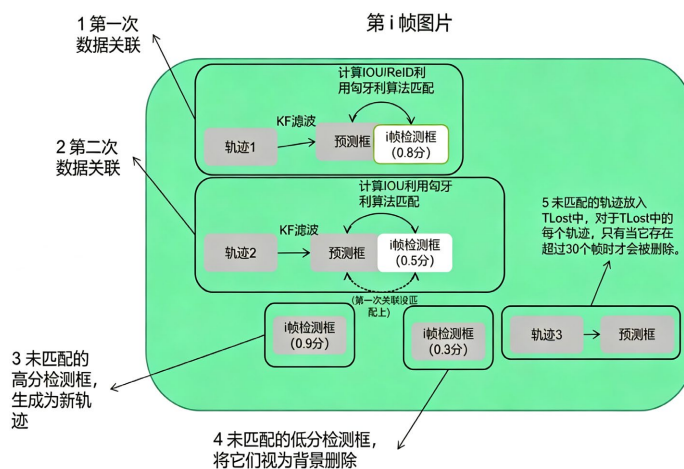


Figure 4. Bytetrack algorithm
图 4. Bytetrack 算法

5. 实验结果分析

5.1. 数据集与评价指标

本研究采用的数据是贵州高速公路内隧道的固定监控视频中随机抽关键帧，覆盖入口、中段、出口三种典型光照与车流密度场景。因为隧道内监控摄像机因安装位置差异导致车辆成像视角不同，所以本数据选取俯视角度较大约 30°至 45°入口和出口摄像机，车辆呈现近大远小特征，易捕捉车牌细节但车身易被局部遮挡。以及平视角度为主约 5°至 15°隧道中段摄像机，车辆侧面轮廓清晰，但相邻车道车辆易形成横向遮挡。其次是选取一定的弯道区域摄像机，广角镜头导致边缘畸变，车辆长宽比动态变化，部分数据展示如图 1。

评估指标将以下指标进行评估，具体如下：

(1) 精度(Precision)

精度衡量的是检测器预测为正样本的结果中，有多少比例是真正的正样本。

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (1)$$

表达式(1)中：

TP (True Positive): 真正例，表示被正确检测到的正样本数量。

FP (False Positive): 假正例，表示被错误地检测为正样本的负样本数量。

(2) 召回率(Recall)

召回率衡量的是所有真正的正样本中，有多少比例被检测器正确地检测。

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2)$$

表达式(2)中：

FN (False Negative): 假负例，表示没有被检测到的正样本数量。

(3) 平均精度(Average Precision, AP)

平均精度(AP)衡量的是检测器在不同召回率水平下的平均精度。对于每个类别，根据检测结果的置信度对检测框进行排序。计算不同召回率水平下的精度值。

绘制 Precision-Recall 曲线。计算 Precision-Recall 曲线下的面积，即为 AP 值，表达式(4-3) AP 值越高，表示检测器在该类别上的性能越好。

$$\text{AP} = \sum_{i=1}^N (R_i - R_{i-1}) \cdot P_i \quad (3)$$

表达式(3)中：

P_i 和 R_i 分别为第 i 个召回点对应的精度与召回率。

(4) 均值平均精度(mean Average Precision, mAP)

均值平均精度(mAP)是多个类别 AP 值的平均值，用于综合评估检测器在所有类别上的性能。mAP 值越高，表示检测器在所有类别上的平均性能越好。

$$\text{mAP} = \frac{1}{N_{\text{classes}}} \sum_{i=1}^{N_{\text{classes}}} AP_i \quad (4)$$

表达式(4)中：

N_{classes} ：类别的总数， AP_i ：第 i 个类别的平均精度。

(5) 身份切换次数(ID Switches, IDS)

当追踪器未能正确关联同一目标在不同帧中的检测结果时，会导致该目标被重新分配新的 ID，即发生身份切换。IDS 数值与追踪性能呈负相关关系，IDS 值越低，表明算法在目标身份保持方面的鲁棒性越强。

(6) 准确识别率(Correct Identification Rate)

准确识别率衡量系统在整个视频序列中正确追踪目标的程度，即目标 ID 保持一致且没有发生追踪丢失的比例。

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N_{\text{total}}} \quad (5)$$

表达式(5)中:

N_{correct} : 在整个视频中被正确追踪 ID 一致、未断失的车辆数, N_{total} : 视频中出现的总车辆数。

(7) 处理帧率(Frames Per Second, FPS)

处理帧率表示系统每秒能够处理的视频帧数, 是评估算法实时性能的重要指标。较高的 FPS 值表示系统能够更快地处理视频数据, 满足实时性要求。

5.2. 实验结果与分析

为验证 ByteTrack 算法各模块在隧道环境中的有效性, 本文设计两种实验分别为消融实验和不同算法对比实验进行验证。首先消融实验以 YOLOv5 + 卡尔曼滤波 + 匈牙利匹配作为基线(Base), 在此基础上逐步引入 ByteTrack 的核心改进模块, 共设置三个组:

(1) Base: 仅使用高置信度检测框进行卡尔曼滤波预测与匈牙利匹配, 丢弃低分框;

(2) Base + Low Init: 在 Base 基础上, 允许低分框用于新轨迹的初始化, 但不参与已有轨迹的二次匹配;

(3) Full ByteTrack: 完整引入 ByteTrack 的低分框二次匹配机制, 即先利用高分框与活跃轨迹匹配, 再利用剩余轨迹与低分框进行 IOU 匹配, 以恢复可能遗漏的轨迹。

实验在相同隧道遮挡密集场景下进行, 包含 70 辆目标车辆视频检测。结果如下表 1 所示。

Table 1. Experimental comparison table

表 1. 实验对比表

配置	ID 保持率(%)	ID 切换次数(次)	轨迹恢复率(%)	处理帧率(FPS)
Base	78.3	22	62.5	91.2
Base + Low Init	86.5	15	78.9	88.5
Full ByteTrack	95.7	9	94.2	83.6

结果表明: 仅引入低分框初始化轨迹可使 ID 保持率提升 8.2%, ID 切换次数下降 7 次; 而完整的低分框二次匹配机制进一步将 ID 保持率提升至 95.7%, 轨迹恢复率达到 94.2%, 验证了低分框二次匹配对遮挡场景下轨迹恢复的关键作用。处理帧率虽略有下降, 仍远高于隧道监控实时性要求。

此外, 本文对算法复杂度进行了分析。YOLOv5 + ByteTrack 的模型参数量约为 7.2M, 单帧推理计算量约为 16.4 GFLOPs, 在 NVIDIA Tesla T4 显卡上的处理帧率(FPS)为 83.6, 可满足隧道监控 25~30 帧/秒的实时性[8]要求。

其次, 算法对比实验, 两种算法 YOLOv5 + ByteTrack 和当下表现优异的算法 YOLOv5 + DeepSORT[9]进行了对比, 具体内容如下:

(1) 遮挡与车流密度极端情况分析

在隧道进出口或瓶颈段, 经常会出现车辆密集行驶甚至并行、超车等情况, 容易产生前后车遮挡或侧面遮挡现象, 严重干扰车辆追踪中的数据关联过程。在高密度车流场景中, YOLOv5 + DeepSORT 由于依赖 Re-ID 特征进行匹配, 能够一定程度上维持遮挡前后 ID 的一致性, 但当车辆颜色与形状相近时, Re-ID 特征容易混淆, 导致 ID 切换。YOLOv5 + ByteTrack 由于引入了低置信目标用于补全轨迹, 即便在短时间遮挡后重新检测到目标, 仍能较为顺畅地完成轨迹恢复。因此在 ID 保持率上优。

但在完全遮挡持续超过 1 秒以上时, 两者均出现了短暂的目标丢失。遮挡情况导致目标边界框重叠或被完全覆盖, 在匹配阶段误关联频繁, 尤其在车辆颜色、尺寸、行驶方向相近时, 高车流状态下检测

阶段产生大量冗余框，增加 ID 混乱概率。如图所示，展示了同一车辆在 DeepSORT (图 5(a))与 ByteTrack (图 5(b))两种算法下的追踪效果对比。可以观察到，ByteTrack 能更稳定地保持车辆 ID 的一致性，而 DeepSORT 在部分帧中出现了 ID 波动或轻微丢失的情况，表明 ByteTrack 在本场景下具有更强的鲁棒性。



Figure 5. Comparison chart of vehicle target tracking consistency across different algorithms
图 5. 车辆目标在不同算法中的追踪一致性对比图

(2) 车辆速度变化与运动模糊影响

在隧道内车辆往往会出现加速或减速行为，特别是在进出隧道或出现拥堵缓解时，车辆速度波动较大。高速运动时，摄像头帧率较低或快门时间不匹配时容易产生运动模糊，导致目标边缘不清晰，严重时目标特征被拉伸或变形，影响检测精度与 Re-ID 提取的图像外观特征。

在此场景下，DeepSORT 依赖的外观特征受损显著，误匹配频率上升，导致 ID Switch 率增加，该场景累计 ID 切换次数随帧数变化曲线。如图 6 中可清晰看出，DeepSORT 在遮挡发生后的 100~150 帧内频繁出现 ID 切换，而 ByteTrack 的累计 ID 切换次数始终平稳上升，验证了其在遮挡场景下的身份保持稳定性。

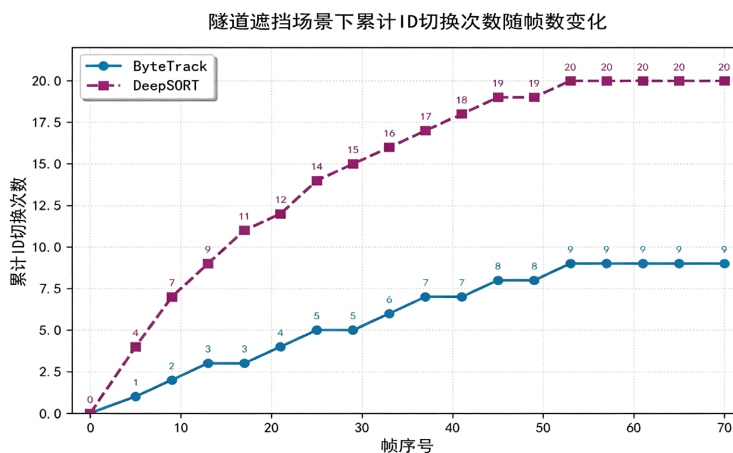


Figure 6. ID switch count chart
图 6. ID 切换次数图

相比之下，ByteTrack 在目标匹配时主要依靠 IOU 与运动预测模型，对图像模糊的依赖性较小，其对短时运动模糊具有更强的鲁棒性，追踪连续性表现更优。如图 7 展示了同一辆车在 DeepSORT 和 ByteTrack 两种算法下的追踪表现。可以看到，DeepSORT 存在 ID 不准确的问题，即车辆在追踪过程中 ID 发生切

换，导致同一车辆被分配了多个不同的 ID。而 ByteTrack 能够保持 ID 一致，追踪更加稳定和连续。

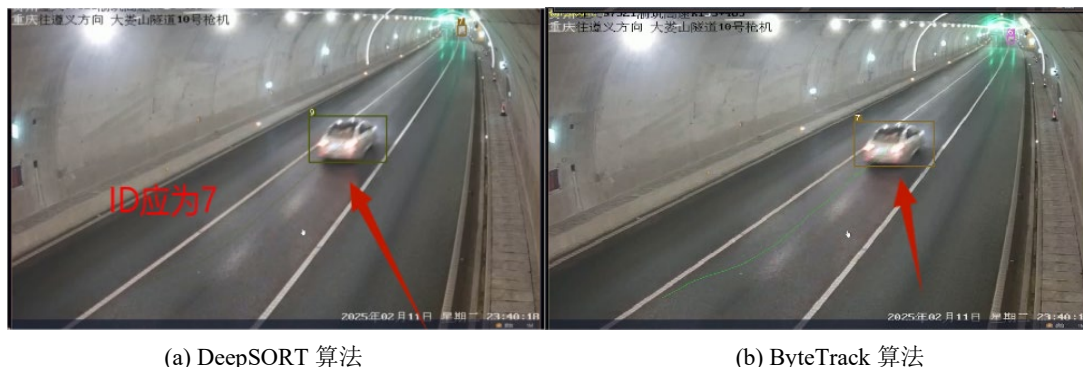


Figure 7. Comparison Chart of Tracking Performance in High-Speed Scenarios
图 7. 车速过快场景中的追踪表现对比图

(3) ID 稳定和准确率

ByteTrack 在检测准确率(93.3% vs 59.3%)和 ID 保持率(95.7% vs 71.4%)上显著优于 DeepSORT，体现其更强的追踪稳定性和目标识别精度；而 DeepSORT 因检测 ID 总数更多、ID 切换次数更高，导致整体稳定性与准确性较低。

Table 2. Table of ID algorithm stability and detection accuracy
表 2. ID 算法稳定性与检测准确性表

算法	实际车辆数	检测 ID 总数	ID 切换次数	MOTA (%)	ID 保持率(%)
ByteTrack 算法	70	75	9	93.3	95.7
DeepSORT 算法	70	118	20	59.3	71.4

表 2 中检测准确率(93.3% vs 59.3%)与 ID 保持率(95.7% vs 71.4%)已在实验视频关键帧中逐帧标注验证。

(4) 不同子场景性能分析

选取入口、中段、出口三个典型子场景，结合核心性能数据如下图 8 所示。对比 YOLOv5 + ByteTrack 与 YOLOv5 + DeepSORT 的适配效果。整体来看，ByteTrack 算法在所有子场景中均展现出显著优势，ID 保持率(%)高于 DeepSORT 算法，ID 切换次数仅变化小，充分体现了 ByteTrack 在隧道复杂场景中的追踪稳定性与准确性。



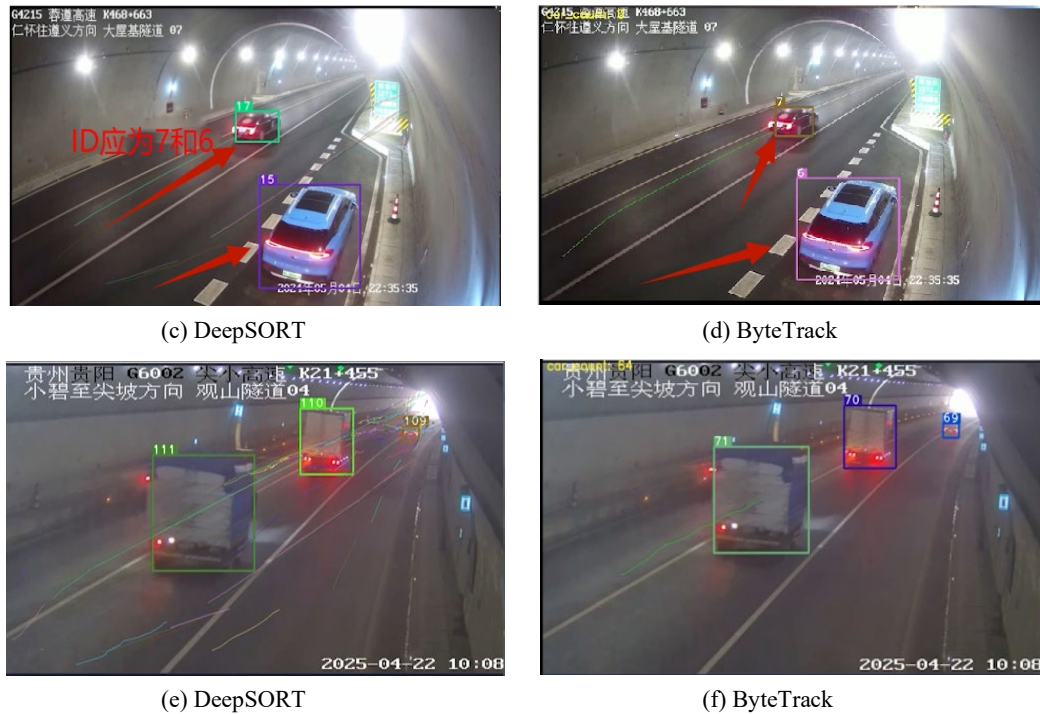


Figure 8. Comparison of detection effects in different segments
图 8. 不同段检测效果对比

此外，与近年来提出的 OC-SORT 算法[10]相比，YOLOv5 + ByteTrack 在隧道遮挡密集场景下的 ID 保持率略低(OC-SORT 为 96.3%)，但在视频处理帧率上更高(OC-SORT 为 78.2 FPS)，更加适合隧道监控的实时性需求。OC-SORT 在轨迹平滑性具有很好的优势，未来可结合两种算法的优势进一步优化。OC-SORT 算法由 Cao 等人提出，核心是通过优化关联策略提升遮挡场景下的轨迹稳定性[10]，其性能表现与本文算法形成互补。

6. 结论

本文针对隧道复杂交通环境中的多目标跟踪问题，提出并验证了基于 YOLOv5 + ByteTrack 的多目标跟踪技术方案。研究发现：

- (1) 隧道环境特有的空间封闭、光照骤变、车辆遮挡频繁等特性导致传统监控系统在检测与跟踪方面表现不佳，目标漏检率和追踪丢失率较高。
- (2) YOLOv5 + ByteTrack 算法在隧道复杂环境中表现出色，特别是在遮挡与车流密度极端情况下，ID 保持率(95.7%)显著高于 YOLOv5 + DeepSORT (71.4%)；在车辆速度变化与运动模糊影响下，ByteTrack 算法对图像模糊的依赖性更小，追踪连续性更优。
- (3) 该技术有效解决了隧道环境中目标漏检、ID 切换等问题，为隧道交通监控提供了更精准、更稳定的解决方案。

未来研究方向可进一步探索多传感器融合技术，如视觉 - 惯性 - 毫米波雷达融合架构，以进一步提升隧道环境下的多目标跟踪性能，为智能交通系统的完善提供更强大的技术支持。

参考文献

[1] 金立生, 华强, 郭柏苍, 等. 基于优化 DeepSort 的前方车辆多目标跟踪[J]. 浙江大学学报(工学版), 2021, 55(6):

- 1056-1064.
- [2] Xie, Y.F., Li, X.H., Wang, D.H., *et al.* (2026) Robust 3D Multi-Object Tracking via 4D mmWave Radar-Camera Fusion and Disparity-Domain Depth Recovery. *Sensors*, **26**, Article No. 2096. <https://doi.org/10.3390/s26072096>
- [3] 彭红星, 袁畅, 柯威曳, 等. 基于改进 YOLOv5 的高速公路隧道车辆和人员检测[J]. 科学技术与工程, 2024, 24(6): 2453-2461.
- [4] 中华人民共和国交通运输部. 2023 年交通运输行业发展统计公报[R/OL]. 2024-06-18. https://xxgk.mot.gov.cn/2020/jigou/zhghs/202406/t20240614_4142419.html, 2026-06-16.
- [5] 张安, 潘怡静, 赖志坚, 等. 基于 YOLOv8 的隧道交通监控车辆检测与跟踪算法研究[J]. 装备制造技术, 2025(1): 35-38.
- [6] 许璧麒, 马志强, 宝财吉拉呼, 等. 基于 YOLOv5 的高速公路小目标车辆逆行检测模型[J]. 国外电子测量技术, 2022, 41(11): 146-153.
- [7] 李福进, 黄志伟. 联合 YOLOv3 检测和 ReID 的目标跟踪方法[J]. 华北理工大学学报(自然科学版), 2021, 43(3): 110-118.
- [8] Ma, N., Wang, J., Qin, K. and Han, Y. (2024) A Real-Time Vehicle Detection and Tracking System Based on an Improved YOLOv5 with Lightweight Convolutions and Multi-Scale Fusion Combined with BYTETTrack. *2024 3rd International Conference on Artificial Intelligence, Human-Computer Interaction and Robotics (AIHCIR)*, Hong Kong, 15-17 November 2024, 203-209. <https://doi.org/10.1109/aihcir65563.2024.00042>
- [9] Robotics & Machine Learning Daily News (2022) Beijing Academy of Agriculture and Forestry Sciences Researchers Report Recent Findings in Robotics Tracking and Counting of Tomato at Different Growth Period Using an Improving YOLO Deepsort Network for Inspection Robot.
- [10] Cao, Y., Wang, C., Zhang, Y., *et al.* (2022) OC-SORT: Observation-Centric SORT for Multi-Object Tracking.