

文章编号:1001-5078(2025)01-0138-07

· 图像与信号处理 ·

融合多传感器的行人识别和多目标追踪算法

李京鑫^{1,2}, 孙健³, 罗家毅¹, 吴海波¹

(1. 马鞍山职业技术学院机械工程系, 安徽 马鞍山 243031; 2. 南京林业大学汽车与交通工程学院, 江苏 南京 210037;

3. 江苏大学电气信息工程学院, 江苏 镇江 212013)

摘要: 车辆自动驾驶过程中不仅需要完成运动规划, 实现行人识别, 还需实现更高精度的多目标追踪。针对车辆自动驾驶行人识别响应速度慢、目标追踪精度差的问题, 提出一种融合多传感器的行人识别和多目标追踪算法。采用 Lattice 算法进行路径规划, 通过损失函数与碰撞检测得到最佳驾驶轨迹, 将传感器检测到的障碍物位置转换到全局坐标系中, 在全局栅格地图中以高斯分布绘制在地图中, 通过阈值初定可视行人。并基于检测跟踪策略设计多目标追踪遮挡处理算法, 实现自动驾驶车辆中被遮挡目标的运动估计。对多目标跟踪挑战数据集的定量、定性和消融研究, 验证了该算法的有效性。实验结果表明, 该算法能够准确估计遮挡过程中的目标运动, 生成完整、高质量的运动轨迹。

关键词: 行人识别; 多目标追踪; 遮挡处理; 算法; Lattice

中图分类号: TP391.4; U461.91 **文献标识码:** A **DOI:** 10.3969/j.issn.1001-5078.2025.01.020

A pedestrian recognition and multi-target tracking algorithm integrating multiple sensors

LI Jing-xin^{1,2}, SUN Jian³, LUO Jia-yi¹, WU Hai-bo¹

(1. Department of Mechanical Engineering, Maanshan Technical College, Maanshan 243031, China;

2. College of Automobile and Traffic Engineering, Nanjing Forestry University, Nanjing 210037, China;

3. School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, China)

Abstract: During the process of autonomous vehicle driving, it is not only necessary to complete motion planning and achieve pedestrian recognition, but also to implement higher precision multi-target tracking. A pedestrian recognition and multi-target tracking algorithm integrating multiple sensors is proposed to address the issues of slow response speed and poor target tracking accuracy in autonomous vehicle driving. Using the Lattice algorithm for path planning, the optimal driving trajectory is obtained through loss function and collision detection. The obstacle position detected by the sensor is converted to the global coordinate system, and Gaussian distribution is drawn on the global grid map. The visible pedestrian is initially determined through threshold. And a multi-objective tracking occlusion processing algorithm is designed based on the detection and tracking strategy to achieve motion estimation of occluded targets in autonomous vehicles. Quantitative, qualitative, and ablation studies on the multi target tracking challenge dataset validate the effectiveness of the algorithm. The experimental results show that the algorithm can accurately estimate the target motion during occlusion and generate complete, high-quality motion trajectories.

Keywords: pedestrian recognition; multi target tracking; occlusion processing; algorithm; Lattice

基金项目: 2020年安徽省高校自然科学研究重点项目(No. KJ2020A0929); 2021年安徽省高校自然科学研究重点项目(No. KJ2021A1338); 2023年安徽省高等学校自然科学研究项目(No. 2023AH052789)资助。

作者简介: 李京鑫(1991-), 男, 硕士, 讲师, 研究方向为人工智能, 自动驾驶技术。E-mail: 499045147@163.com

收稿日期: 2024-05-14; **修订日期:** 2024-06-17

1 引言

不同程度的障碍物遮挡识别是自动驾驶领域的难点之一,也是地面军事目标识别技术的研究热点^[1]。尽管目前基于计算机视觉的目标跟踪效率较高,但当目标被遮挡时,跟踪轨迹的完整性和准确性并不理想^[2-3]。此外,计算机视觉中的目标跟踪通过遮挡前后目标的外观特征来进行关联,但缺乏目标遮挡时目标运动的准确表征^[4]。江苏大学杨绍卿设计出一种用于提高自动驾驶效率的基于策略梯度的轨迹规划算法,利用 CARLA 仿真模拟器构建多车道场景,实现了控制算法、规划算法和交通环境的有机耦合,但没有应用于真实的交通环境,缺乏实际应用性^[5]。重庆理工大学陈志远等选用 YOYOX-nano 识别目标类型、位置及大小等信息,采用 PID 控制算法调整自适应转向角度,搭建交通模拟路况,成功以近 30 FPS 的推理速度实时识别出路况中的车道线和障碍物,但控制策略较为简单,无法有效识别被遮挡目标^[6]。Duguleana 等将神经网络算法应用于复杂场景下自动驾驶车辆的主动避障设计,实现了并行处理各种障碍物信息数据,完成有效避障和自主运行^[7]。西安工业大学刘康等以军事环境为交通背景,提出基于注意力机制的 YOLOv5 目标识别算法,采用 Alpha_IoU 对目标识别分类器的损失函数进行改进,加速模型收敛,有效提升了复杂军事战场环境下的目标识别精度^[8]。

自动驾驶中的多目标跟踪(MOT)需要向车辆提供被遮挡目标的运动状态,包括位置和速度^[1,9]。目标运动是自动驾驶中的关键技术之一,了解目标的移动方式比了解目标类型更重要,故搭建模拟交通道路,采用 Lattice 算法进行路径规划,通过损失函数与碰撞检测得到最佳驾驶轨迹,完成智能小车循迹。并在此基础上,并基于检测跟踪策略设计多目标追踪遮挡处理算法,实现真实环境下自动驾驶车辆中被遮挡目标的运动估计。

2 障碍物融合处理

2.1 行人识别算法流程

融合摄像头、激光雷达、毫米波雷达等多种传感器进行障碍物检测^[10-11],将传感器检测到的障碍物位置转换到全局坐标系中,再将检测到的障碍物位置在全局栅格地图中以高斯分布绘制在地图中,构建势场函数根据不同场景确定车与障碍物间的距离与方向阈值,势场与距离方向成反比,势场小于阈值的区域标记为可行驶区域,从而实现通过阈值来最终确定融合后的障碍物。

轨迹点通过三次插值方程生成参考线,后续算法均以该参考线为基准将笛卡尔坐标转换为 Frenet 坐标系。规划算法采用 Lattice 算法:先找到车身对应的最近参考线上参考点,通过该参考点的法线与切线,得到车身的横向距离与纵向距离(Frenet 坐标系下的位置),通过带约束(起始与结束的速度与位置)的 4 次多项式与 5 次多项式在横纵向进行规划,合成横纵向多项式,横纵向规划多条候选轨迹,最后通过损失函数与碰撞检测得到最佳的候选轨迹。算法基本流程如图 1 所示。

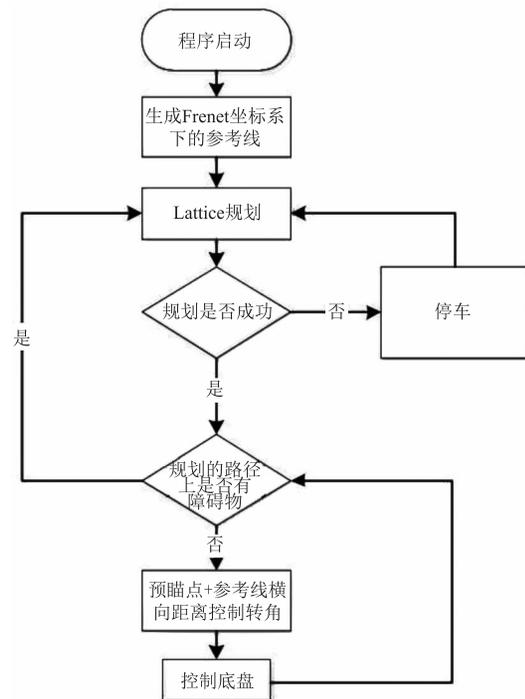


图 1 Lattice 算法基本流程

Fig. 1 The basic process of Lattice algorithm

2.2 参考线生成

参考线的生成需要在地图上标记出轨迹点,获得轨迹点后,通过三次样条插值(cubic spline)获得连续轨迹点。以车辆车身中点坐标 (x, y) 表示车辆在地图中的参考位置,第 i 个区间对应的 cubic spline 函数记为 $S_i(x)$,则:

插值和连续性:

$$S_i(x_i) = y_i, i = 0, 1, \dots, n-1 \quad (1)$$

$$S_i(x_{i+1}) = y_{i+1}, i = 0, 1, \dots, n-1 \quad (2)$$

微分连续性:

$$S'_i(x_{i+1}) = S'_{i+1}(x_{i+1}) \quad (3)$$

$$S''_i(x_{i+1}) = S''_{i+1}(x_{i+1}), i = 0, 1, \dots, n-2 \quad (4)$$

样条曲线的微分式:

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 +$$

$$d_i (x - x_i)^3 \quad (5)$$

$$S'_i(x) = b_i + 2c_i(x - x_i) + 3d_i(x - x_i)^2 \quad (6)$$

$$S''_i(x) = 2c_i + 6d_i(x - x_i) \quad (7)$$

其中, a, b, c, d 分别为函数 $S_i(x)$ 的边界条件参数。

将步长 $h_i = x_{i+1} - x_i$ 代入样条曲线的条件为:

$$a. \text{ 由 } S_i(x_i) = y_i (i = 0, 1, \dots, n-1)$$

推出:

$$a_i = y_i \quad (8)$$

$$b. \text{ 由 } S_i(x_{i+1}) = y_{i+1} (i = 0, 1, \dots, n-1)$$

推出:

$$y_i + h_i b_i + h_i^2 c_i + h_i^3 d_i = y_{i+1} \quad (9)$$

$$c. \text{ 由 } S'_i(x_{i+1}) = S'_{i+1}(x_{i+1}) (i = 0, 1, \dots, n-2)$$

推出:

$$\begin{aligned} S'_i(x_{i+1}) &= b_i + 2c_i(x_{i+1} - x_i) + 3d_i(x_{i+1} - x_i)^2 \\ &= b_i + 2c_i h_i + 3d_i h_i^2 \end{aligned} \quad (10)$$

$$\begin{aligned} S'_{i+1}(x_{i+1}) &= b_{i+1} + 2c_i(x_{i+1} - x_{i+1}) + \\ &3d_i(x_{i+1} - x_{i+1})^2 = b_{i+1} \end{aligned} \quad (11)$$

由此可得:

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ h_0 & 2(h_0 + h_1) & h_1 & 0 & 0 \\ 0 & h_1 & 2(h_1 + h_2) & h_2 & 0 \\ 0 & 0 & h_2 & 2(h_2 + h_3) & h_3 \\ \vdots & & & \ddots & \vdots \\ 0 & \dots & 0 & h_{n-2} & 2(h_{n-2} + h_{n-1}) & h_{n-1} \\ 0 & & & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \\ \vdots \\ m_n \end{bmatrix} = 6 \begin{bmatrix} 0 \\ \frac{y_2 - y_1}{h_1} - \frac{y_1 - y_0}{h_0} \\ \frac{y_3 - y_2}{h_2} - \frac{y_2 - y_1}{h_1} \\ \frac{y_4 - y_3}{h_3} - \frac{y_3 - y_2}{h_2} \\ \vdots \\ \frac{y_n - y_{n-1}}{h_{n-1}} - \frac{y_{n-1} - y_{n-2}}{h_{n-2}} \\ 0 \end{bmatrix} \quad (17)$$

将数据节点和指定的首尾端点条件代入矩阵方程。计算样条曲线的系数为:

$$a_i = y_i \quad (18)$$

$$b_i = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{2} m_i - \frac{h_i}{6} (m_{i+1} - m_i) \quad (19)$$

$$c_i = m_i/2 \quad (20)$$

$$d_i = \frac{m_{i+1} - m_i}{6h_i} \quad (21)$$

3 基于 MOT 的多目标跟踪遮挡处理算法设计

3.1 多目标跟踪遮挡处理框架设计

研究提出一种遮挡处理框架来估计遮挡过程中目标的运动状态,通过轨迹平滑和运动估计来保持

$$b_i + 2h_i c_i + 3h_i^2 d_i - b_{i+1} = 0 \quad (12)$$

$$d. \text{ 由 } S''_i(x_{i+1}) = S''_{i+1}(x_{i+1}) (i = 0, 1, \dots, n-2)$$

推出:

$$2c_i + 6h_i d_i - 2c_{i+1} = 0 \quad (13)$$

设 $m_i = S''_i(x_i) = 2c_i$, 则可推出:

$$d_i = \frac{m_{i+1} - m_i}{6h_i} \quad (14)$$

将 c_i, d_i 代入式(9), 可得:

$$b_i = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{2} m_i - \frac{h_i}{6} (m_{i+1} - m_i) \quad (15)$$

将 b_i, c_i, d_i 代入式(12), 可得:

$$\begin{aligned} h_i m_i + 2(h_i + h_{i+1}) m_{i+1} + h_{i+1} m_{i+2} \\ = 6 \left[\frac{y_{i+2} - y_{i+1}}{h_{i+1}} - \frac{y_{i+1} - y_i}{h_i} \right] \end{aligned} \quad (16)$$

由 i 的取值范围可知, 共有 $n-1$ 个公式, 有 $n+1$ 个未知量 m , 需对两端点 X_0 和 X_n 的微分加以限制, 根据研究对象的无规律性和自由性, 选取自由边界限制法 (Natural spline)。首尾两端不存在弯曲线, 即 $S'' = 0$, 则方程组可写为:

可靠性, 即采用检测策略跟踪目标。获取目标检测结果后, 取边界框的位置 (α, β) , 宽度和高度 (λ, ρ) , 以及相应的变化速度 $(v_\alpha, v_\beta, v_\lambda, v_\rho)$ 来表示所观察目标的状态。

目标的状态 j 在时间 τ 时可以表示为:

$$\mu_\tau^j = \{\alpha_\tau^j, \beta_\tau^j, \lambda_\tau^j, \rho_\tau^j, v_{\alpha\tau}^j, v_{\beta\tau}^j, v_{\lambda\tau}^j, v_{\rho\tau}^j\} \quad (22)$$

观测到的限时区域 τ 表示为 z_τ , 并且通过卡尔曼预测得到了预测范围 $\mu_\tau^j |_{\tau-1}$, 再通过匈牙利算法以交叉口联合 (IOU, Intersection Over Union) 和交叉口目标面积总和作为损失矩阵进行关联 z_τ^j 和 $\mu_\tau^j |_{\tau-1}$ 。如果预测框与检测框不匹配, 则采用卡尔曼平滑器对不匹配的目标轨迹进行平滑处理, 平滑

状态表示为 $\tilde{\mu}_\tau$ 。建立相对运动模型,确定遮挡目标属于哪种类型,并对每种类型采用不同的运动估计策略。如果被遮挡的目标是独立的,则将平滑的轨迹输入到自动回归综合移动平均模型(Autoregressive Integrated Moving Average Model, ARIMA)中进行运动估计。如果被遮挡的物体是一个集体,则使用相对运动模型进行运动预测。

3.2 轨迹平滑

由于探测器的作用,探测轨迹是凹凸不平的,通常会在边界盒的位置和比例上产生噪声。故采用卡尔曼滤波去噪,依据历史数据产生估计。当目标可见时,观察 z_τ 用于更新 $\mu_{\tau|\tau-1}$ 。当目标被遮挡时,基于一阶马尔可夫假设,由于没有观测值来更新预测的状态,若先验预测 $\mu_{\tau|\tau-1}$ 而不是后验 $\mu_{\tau|\tau}$,会导致重大错误。故降低误差的一种策略是通过向后传播关系 $p(\mu_k | z_{1:\tau}, \mu_{k+1:\tau})$ 平滑状态 μ_k 的估计,利用历史轨迹对当前状态进行卡尔曼平滑,减少目标被遮挡前的状态估计误差。

考虑到固定时刻 k ,卡尔曼平滑利用之前和之后的数据来生成估计。首先给定先验序列 $\mu_{\tau+1|\tau}$ 、后验序列 $\mu_{\tau|\tau}$ 以及相应的协方差 $\sum_{\tau+1|\tau}$ 和 $\sum_{\tau|\tau}$ 。在轨迹平滑模块中,当 τ 时刻发生遮挡时,采用反向传播 $\mu_{\tau|\tau}$ 和 $\sum_{\tau|\tau}$ 以获得平滑状态 $\sum_{\tau|\tau}^s$ 。

卡尔曼平滑法如下:

$$\mu_{\tau|\tau}^s = \mu_{\tau|\tau} + G_\tau(\mu_{\tau+1|\tau+1}^s - \mu_{\tau+1|\tau}) \quad (23)$$

$$\sum_{\tau|\tau}^s = \sum_{\tau|\tau} + G_\tau(\sum_{\tau+1|\tau+1}^s - \sum_{\tau+1|\tau})G_\tau^T \quad (24)$$

式中, $\mu_{\tau|\tau}^s$ 是目标在 τ 时刻的平滑状态; $\sum_{\tau|\tau}^s$ 是目标在 τ 时刻的平滑协方差; G_τ 为卡尔曼平滑矩阵。经过平滑后,可以得到一个更平坦的状态估计 $\mu_{\tau|\tau}^s$ 。

3.3 遮挡物分类

当遮挡发生时,由于没有观测值来更新先验,卡尔曼方程不再服从马尔科夫和线性模型假设。故将遮挡目标分为 Companion 和 Loner。Companion 指在遮挡发生之前与目标 A 一起行走的目标 B,而 Loner 指独立的目标。对于 Companion 和 Loner,分别采用了 Relation Motion 和运动预测模型。与其他相对运动网络不同,所设计的相对运动模型只关注遮挡目

标和可见目标。且由于观测噪声的影响,并未在卡尔曼滤波中引入相关模型,而是将其作为群 Companion 和 Loner 的判据。在 τ 时刻,遮挡目标可记为 $o_\tau^A \in O_\tau, A = 1, 2, \dots, N, N$ 表示遮挡目标总数,可见目标可表示为 $v_\tau^B \in V_\tau, B = 1, 2, \dots, M, M$ 表示可见目标总数。当目标 A 在 $\tau + 1$ 时刻被遮挡时,基于 τ 时刻的可见目标 A 和 B 建立相对运动模型:

$$\begin{aligned} r_\tau^{(A,B)} &= o_{\mu\tau}^A - o_{\mu\tau}^B = [\alpha_\tau^A - \alpha_\tau^B, \beta_\tau^A - \beta_\tau^B, \\ &\lambda_\tau^A - \lambda_\tau^B, \rho_\tau^A - \rho_\tau^B, v_{\alpha\tau}^A - v_{\alpha\tau}^B, v_{\beta\tau}^A - v_{\beta\tau}^B, \\ &v_{\lambda\tau}^A - v_{\lambda\tau}^B, v_{\rho\tau}^A - v_{\rho\tau}^B], A, B \in V_\tau, A \in O_{\tau+1} \end{aligned} \quad (25)$$

为寻找 τ 时刻被遮挡目标 A 的 Companion,遮挡目标 A 与 Companion B 应具有相似的大小,故使 B 尺寸损失函数最小化为:

$$C_{\text{size}} = -\ln\left(1 - \frac{|\rho^A - \rho^B|}{2(\rho^A + \rho^B)} - \frac{|\lambda^A - \lambda^B|}{2(\lambda^A + \lambda^B)}\right), \quad A, B \in V_\tau \quad (26)$$

对于 Companion 来说, A、B 速度相近,所以 B 也必须满足:

$$B = \operatorname{argmin}_{(A,B)} \sqrt{(v_\tau^A - v_\tau^B)^2} \quad (27)$$

对于 Companion 而言,假设 A 和 B 是相对静止的,故在遮挡过程中轨迹的恢复式如下所示:

$$\mu_\tau^A = \mu_{\tau-1}^A + r_{\tau-1}^{(A,B)} \quad (28)$$

3.4 Loner 的运动估计

如果在任何轨迹上均无法找到 Companion,目标 A 则被定义为 Loner。对于 Loner,没有可靠的 Companion 参照来建立相对运动模型。由于没有可靠的数据来校正预测值,故针对 Loner 使用高阶马尔可夫假设。由于多层 LSTM 和单层 LSTM 在轨迹预测中的作用相似,因此,采用如图 2 所示的结构进行轨迹预测,使用 mol5、mol6、mol7 和 mo20 作为训练集来预测接下来的 5 帧与 10 帧历史轨迹的运动状态。

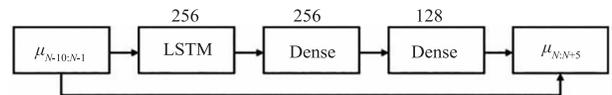


图2 运动预测网络结构

Fig. 2 Structure of motion prediction network

4 实验验证

4.1 巡航模拟实验

基于计算机视觉的红绿灯检测模块中,主要采用深度学习方法进行检测,包括红绿灯数据采集、数据标注、模型训练、模型转换、ncnn 环境部署

等^[12-13]。选取了轻量级的 NanoDet - m 模型,对单阶段检测模型三大模块(Head、Neck、Backbone)进行轻量化。使用 ATSS 进行目标采样,应用 Generalized Focal Loss 损失函数执行分类和边框回归(box regression),该函数能够去掉 FCOS 的 Centerness 分支,省去这一分支上的大量卷积,从而减少检测头的计算开销,非常适合移动端的轻量化部署。

建立模拟道路沙盘,对摄像头进行标定后,求出基于图像坐标系下像素点与世界坐标系下的目标点位置的转换关系,已知目标物位置,输出目标距离车辆实际位置信息。采集红绿灯数据与训练模型,模型推理步骤中确保模型准确检测出红灯、绿灯、黄灯等类别,检测结果如图 3 所示。结果表明:智能小车能够在白线左侧保持一定偏移距离进行绕行,行驶轨迹平直而稳定,没有出现明显的偏离或摇摆现象。当检测到红灯或黄灯、且小车与红绿灯实际距离约 30 cm 时,小车停车等候;当红绿灯为绿灯时,小车正常通行。



图 3 模拟道路交通灯检测结果

Fig. 3 Simulated road traffic light detection results

在控制车辆安全通过人行道模块中,主要表现为小车正常行驶过程中,当行驶即将经过人行道路段,判断前方是否有行人,无行人则正常通行;若检测到行人,在行人前方约 30 cm 处车辆自动停止;当行人消失,车辆再次启动,正常行驶。实现过程中主要需要行人识别结果及行人距离信息,基于以上两个处理结果,控制车辆安全通行。

基于网络模型准确识别行人及行人距离信息输入,示例行人识别效果如图 4 所示,实验表明:加入控制逻辑后,成功判断出前方存在行人,并在距离约 30 cm 时,停止通行;当行人不在本车道或距离大于 30 cm 时,车辆完成正常通行。

4.2 多目标跟踪遮挡处理实验

4.2.1 指标与数据

为评估算法在遮挡过程中的跟踪性能,选择了行人模型数据集 MOTChallenge17 进行评估和可视化。MOT17 包含 7 个训练集和 7 个测试集,提供来自不同场景、视图、照明和环境条件的数据,包括行人之间更多的相互遮挡。

为了更全面地分析遮挡处理对目标跟踪性能的影响,使用 Clear-MOT 和 ID 度量作为评价指标。如表 1 所示为每个指标的简要描述,所设计的方法主要考虑遮挡过程中轨迹的完整性和目标运动的准确性,故 ID 度量是主要比较指标。



图 4 模拟道路行人识别结果

Fig. 4 Simulation of pedestrian recognition results on roads

表 1 目标跟踪性能评价指标与描述

Tab. 1 Target tracking performance evaluation indicators and description

指标	描述
IDF1	正确识别的检测结果与平均值的比率
IDTP	鉴定真假阳性
IDFP	鉴定假阳性
IDTN	识别真阴性
IDFN	鉴定假阴性
IDP	识别精度
IDR	身份识别召回
ID Sw.	标识交换机的数量

所设计的方法专注于处理遮挡过程中的遗漏检测和估计它们的运动,以提高自动驾驶汽车的安全性。因此,提取了 MOTChallenge17 中完全不可见的目标,并计算了主要跟踪指标(MT)、部分跟踪指标(PT)和大部分丢失指标(ML),表明跟踪算法对轨迹的恢复。选择 TrackEval 提供的官方代码进行评估,并使用 MOT17 中的被遮挡对象作为 ground truth 来模拟遮挡性能的评估。利用 MOT 提供的

DPM、FRCNN 和 SDP 的检测结果进行跟踪。选取 MOT17 排名前 40 位的开源代码和我们的基线方法 SORT 进行比较,结果如表 2 所示。在 MOT17 数据集中有 831 个被遮挡的目标。由于使用了未来帧,离线算法在跟踪指标中表现良好。ARIMA 和 LSTM 作为 Loner 的运动估计模型在 MT、PT 和 ML 指标上均有优势,证明了本算法在目标遮挡过程中恢复隐形轨迹的能力,即本算法在线跟踪被遮挡目标的能力更强。

表 2 遮挡评估

Tab. 2 Occlusion assessment

评估方法	状态	MT	PT	ML
Lif_T	脱机状态	91	126	614
LPC_MOT		6156	175	500
MPNTrack		168	179	484
CTTrackPub	在线状态	78	132	621
Trackorv2		59	100	672
SORT17		47	103	681
ARIMA		96	205	530
LSTM		109	230	492

4. 2. 2 实验结果

1. 实验定性结果

由于在 MOTChallenge 中有大量的小目标很难通过重叠来关联,定量实验不能清楚地说明遮挡处理。为了直观地优化该算法对遮挡处理的影响,故选择 MOTChallenge17-02 的典型遮挡进行可视化。如图 5 所示,当目标被遮挡时,首先判断目标是 Companion 还是 Loner。假设是 Companion,如图 5(a)

中的目标 A,基于 A、B 的相对运动模型,利用式(28)恢复轨迹。如果是 Loner,如图 5(b)中的目标 C,在物体被遮挡之前的最后一刻,无法找到满足式(27)的目标 B,则使用 LSTM 运动估计模型进行轨迹预测。

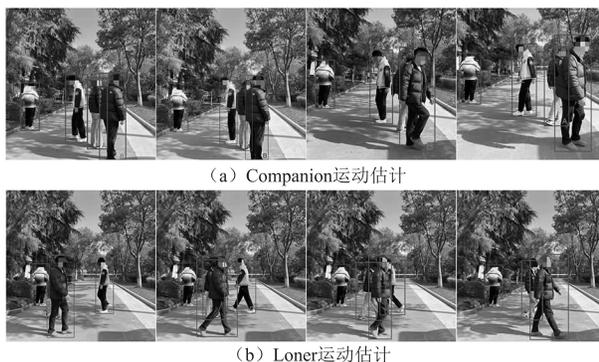


图 5 同伴和单独的遮挡处理的可视化

Fig. 5 Visualization of peer and individual occlusion processing

2. 消融实验

为进一步优化处理遮挡的交互性、减少在遮挡期间的缺失和错误检测,并生成完整和准确的轨迹,利用 POI 检测数据库进行消融实验,并选择 SORT 和 DeepSORT 作为基线模型进行对比,消融结果如表 3 所示。SORT 算法利用卡尔曼滤波和线性运动模型来预测运动状态,不需要遮挡处理。而 DeepSORT 引入了外观特征匹配模块与重识别技术,在 ID 度量上取得了显著的改进。与前两者方法相比,所设计的方法在 ID 度量上的表现比基线 SORT 更优化,可以保证多目标轨迹检测的完整性。

表 3 消融实验结果

Tab. 3 Results of ablation experiment

方法	MOTA	MOTP	IDSW	Frag	IDFI	IDR	IDP	IDTP	IDFN	IDFP
SORT	56.69	81.279	1893	1523	54.975	48.74	63.04	53812	56596	31548
DeepSORT	56.987	81.043	957	1415	61.972	56.335	68.859	62198	48210	28129
ARIMA	55.735	81.89	554	862	60.913	50.903	75.823	56202	54206	17921
LSTM	54.398	81.733	560	824	59.981	51.154	72.493	56477	53932	21433

综上,所设计的方法专注于自动驾驶场景中的运动估计以确保车辆的安全,并为自动驾驶中的 MOT 提供了新的遮挡处理框架。通过对输入数据进行去噪和平滑进行跟踪,将被遮挡目标分类为 Companion 和 Loner,并采用不同的运动估计策略进行隐形轨迹恢复和预测,从而提高自动驾驶汽车的安全

性。

5 结语

为提升复杂交通环境下行人识别的效率和精度,设计一种基于 Lattice 算法的路径规划方法,通过损失函数与碰撞检测得到最佳驾驶轨迹,融合多传感器完成交通环境识别和行人识别。并针对

现有算法无法高精度追踪遮挡的缺陷,基于检测跟踪策略设计一种多目标追踪遮挡处理算法,实现自动驾驶车辆中被遮挡目标的运动估计,准确估计遮挡过程中的运动目标,提升了车辆自动驾驶安全性。

参考文献:

- [1] Wang J, Chen Y, Dong Z, et al. Improved YOLOv5 network for real-time multi-scale traffic sign detection [J]. *Neural Computing and Applications*, 2022, 35: 7853 – 7865.
- [2] Zhang Rongxia, Wu Changxu, Zhao Zengshun, et al. Deep reinforcement learning and its research progress in path planning [J]. *Computer Engineering and Applications*, 2021, 57(19): 13. (in Chinese)
张荣霞, 武长旭, 赵增顺, 等. 深度强化学习及在路径规划中的研究进展 [J]. *计算机工程与应用*, 2021, 57(19): 13.
- [3] Spielberg N A, Brown M, Kapania N R, et al. Neural network vehicle models for high-performance automated driving [J]. *Science Robotics*, 2019, 4(28): eaaw1975.
- [4] HE J, ERFANI S, MA X, et al. Alpha-IoU: a family of power intersection over union losses for bounding box regression [J]. *Advances in Neural Information Processing Systems*, 2021(34): 20230 – 20242.
- [5] YANG Shao-qing. Path planning and control method of autonomous vehicle based on reinforcement learning [D]. Zhengjiang: Jiangsu University, 2022. (in Chinese)
杨绍卿. 基于强化学习的自动驾驶汽车轨迹规划与控制方法 [D]. 镇江: 江苏大学, 2022.
- [6] Chen Zhiyuan. Research on cruise avoidance of intelligent connected vehicles based on deep learning [D]. Chongqing: Chongqing University of Technology, 2023. (in Chinese)
陈志远. 基于深度学习的智能网联小车巡航避障研究 [D]. 重庆: 重庆理工大学, 2023.
- [7] Duguleana M, Mogan G. Neural networks based reinforcement learning for mobile robots obstacle avoidance [J]. *Expert Systems with Applications*, 2016, 62(15): 104 – 115.
- [8] Liu Kang, Song Xiaoru, Gao Song. Improved YOLOv5 ground military target recognition algorithm [J]. *Fire Control & Command Control*, 2023, 48(11): 58 – 66. (in Chinese)
刘康, 宋晓茹, 高嵩, 等. 改进的 YOLOv5 地面军事目标识别算法 [J]. *火力与指挥控制*, 2023, 48(11): 58 – 66.
- [9] Wang Wei, Chang Yanwen, Yu Tao, et al. A multi intelligent vehicle formation motion planning method based on lattice algorithm [J]. *Journal of Detection and Control*, 2022, 44(5): 105 – 111. (in Chinese)
王伟, 常彦文, 余涛, 等. 基于 Lattice 算法的多智能车编队运动规划方法 [J]. *探测与控制学报*, 2022, 44(5): 105 – 111.
- [10] Zhang Z, Zhang D, Qiu R C. Deep reinforcement learning for power system applications: an overview [J]. *CSEE Journal of Power and Energy Systems*, 2019, 6(1): 213 – 225.
- [11] Kattenborn T, Leitloff J, Schiefer F, et al. Review on convolutional neural networks (CNN) in vegetation remote sensing [J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2021, 173: 24 – 49.
- [12] Zhu D D, Sun J Q. A new algorithm based on Dijkstra for vehicle path planning considering intersection attribute [J]. *IEEE Access*, 2021, 9: 19761 – 19775.
- [13] Lu W, Li J, Wang J, et al. A CNN-BiLSTM-AM method for stock price prediction [J]. *Neural Computing and Applications*, 2021, 33: 4741 – 4753.