

动态场景优化 ORB-SLAM3 算法

徐淑萍, 杨定哲, 房嘉翔, 蒋硕

(西安工业大学 计算机科学与工程学院, 西安 710021)

摘要: 针对动态场景中移动物体引发的机器人位姿估计偏差及地图构建不完善问题, 提出一种动态场景优化 ORB-SLAM3 算法。首先通过改进的 YOLOv5s 算法检测动态物体并初步剔除关联特征点, 随后联合 LK 光流跟踪与基于基本矩阵的极线几何约束分析, 进一步滤除漏检的动态特征点, 从而提升环境感知与位姿估计精度。同时, 通过滤除动态信息的关键帧生成对应的点云信息, 实现三维稠密静态地图构建。室内动态场景下的测试结果表明: 相较于传统 ORB-SLAM3, 所提算法的绝对轨迹误差和相对位姿误差在办公室环境下分别减小 55.2% 和 93.7%, 走廊场景中分别减小 24.3% 和 40.2%, 验证了其在动态场景中的鲁棒性优势。

关键词: 改进 YOLOv5s; LK 光流; 极线约束; 三维稠密点云地图

中图分类号: TP391.4

文献标志码: A

ORB-SLAM3 algorithm for dynamic scene optimization

XU Shuping, YANG Dingzhe, FANG Jiexiang, JIANG Shuo

(School of Computer Science and Engineering, Xi'an Technological University, Xi'an 710021, China)

Abstract: Aiming at the problem of robot pose estimation bias and imperfect map construction caused by moving objects in dynamic scenes, an ORB-SLAM3 algorithm for dynamic scene optimization is proposed. Firstly, the dynamic object is detected by the improved YOLOv5s algorithm and the associated feature points are preliminarily removed. Then, the missing dynamic feature points are further filtered by combining LK optical flow tracking and epipolar geometric constraint analysis based on fundamental matrix, so as to improve the accuracy of environment perception and pose estimation. At the same time, the corresponding point cloud information is generated by filtering the key frames of dynamic information to realize the construction of 3D dense static map. The test results in indoor dynamic scenes show that compared with the traditional ORB-SLAM3, the absolute trajectory error and relative pose error of the proposed algorithm are reduced by 55.2% and 93.7% respectively in the office environment, and by 24.3% and 40.2% in the corridor scene, which verifies the robustness advantage of the proposed algorithm in dynamic scenes.

Key words: improve YOLOv5s; LK optical flow; epipolar constraint; 3D dense point cloud map

同步定位与地图构建 (Simultaneous Localization And Mapping, SLAM) 由 Smith 等人于 1986 年提出^[1]。旨在通过机器人搭载的传感器实时获取环境信息, 同步实现自身位姿估计与环境地图构建。根据所使用的传感器类型不同, SLAM 技术主要分为激光 SLAM 与视觉 SLAM 两类: 前者依赖激光雷达获取障碍物信息完成建图^[2]; 后者则通过相机采集环境信息, 实现自身定位与场景地图构建。得益于相机传感器体积紧凑

且能获得丰富的环境信息, 视觉 SLAM 技术备受关注^[3]。然而在实际场景中, 运动物体引发的观测数据异常会导致系统出现位姿估计偏差与地图失真, 如何有效抑制此类干扰一直是困扰视觉 SLAM 的难题^[4]。

ORB-SLAM3 是当前经典的视觉 SLAM 系统之一, 支持单目、双目和 RGB-D 相机, 兼容针孔与鱼眼镜头^[5]。该系统采用基于最大后验估计的紧耦合视觉-惯性融合技术, 优化了惯性测量单元 (Inertial

收稿日期: 2024-10-15; **修回日期:** 2025-05-13

基金项目: 陕西省科技厅重点产业链工业领域一般项目 (2022GY-239)

作者简介: 徐淑萍 (1974—), 女, 教授, 硕士研究生导师, 从事机器人 SLAM、智能检测控制、信息处理等方面研究。

Measurement Unit, IMU) 初始化阶段的位姿精度及系统鲁棒性, 同时通过改进的多地图管理机制增强了回环检测的可靠性。然而, ORB-SLAM3 未设计针对动态物体的专用处理模块, 在动态场景中难以区分动态目标与静态背景, 导致运动物体的特征点匹配误差及位姿漂移, 引发特征点跟踪失效与回环检测紊乱, 最终显著降低系统的地图构建质量。为应对动态环境干扰, RDS-SLAM^[6]在 ORB-SLAM3 框架中增设语义线程与基于语义的优化线程。该系统通过语义分割方法^[7]检测动态物体, 使用移动概率更新和传播语义信息, 从而实时剔除动态特征。然而该方法依赖形态滤波器扩张掩模覆盖动物体边缘的策略存在缺陷, 当动态物体边界未被完整包裹时特征剔除效果显著下降, 致使构建地图质量与传统 ORB-SLAM3 相比无明显提升。DynPL-SVO^[8]则采用动态网格算法检测异常值, 通过融合动态特征垂直和平行方向重投影误差的联合代价函数优化位姿估计, 但这种方法增加了算法运算量, 可能导致建图过程中的实时精度下降。刘胤真等^[9]基于 ORB-SLAM2 框架提出融合语义与几何约束的 SLAM 系统, 通过集成 GhostNet-YOLOv7 目标检测网络^[10]与对极几何约束模块提取动态物体 2D 语义信息以实现动态特征剔除, 但其在弱纹理环境下特征点提取能力不足。针对此问题, 蔡显奇等提出视觉-惯性 SLAM 优化算法^[11], 引入深度学习中的注意力机制与

传统特征检测技术相结合, 建立相机深度置信模型来减少远距离特征的漂移误差, 提高了弱纹理场景下的定位精度, 但该算法仍存在动态特征及环境信息剔除不彻底的问题。

综上所述, 针对动态场景下 SLAM 系统定位精度不足以及 ORB-SLAM3 所构建的稀疏点云地图缺乏关键信息不能应用于导航等问题, 本文提出了一种基于动态目标检测的改进 ORB-SLAM3 算法。在视觉前端, 基于改进的 YOLOv5s 检测算法^[12], 结合光流约束剔除动态特征点, 显著提升系统定位精度。同时改进建图策略, 利用静态关键帧生成对应的点云信息以完成地图构建, 并进一步优化点云匹配策略, 最终形成具备高精度与强鲁棒性的三维稠密点云地图。

1 动态环境下的改进 ORB-SLAM3

1.1 系统整体框架

传统 ORB-SLAM3 框架可以划分为跟踪、局部地图、回环检测和地图构建四个线程^[13]。针对动态物体干扰导致的地图构建误差问题, 本文算法通过添加动态点去除模块, 优化机器人位姿与稠密地图的生成过程, 图 1 为改进的 ORB-SLAM3 框架, 其中蓝色填充框表示改进与添加的模块。

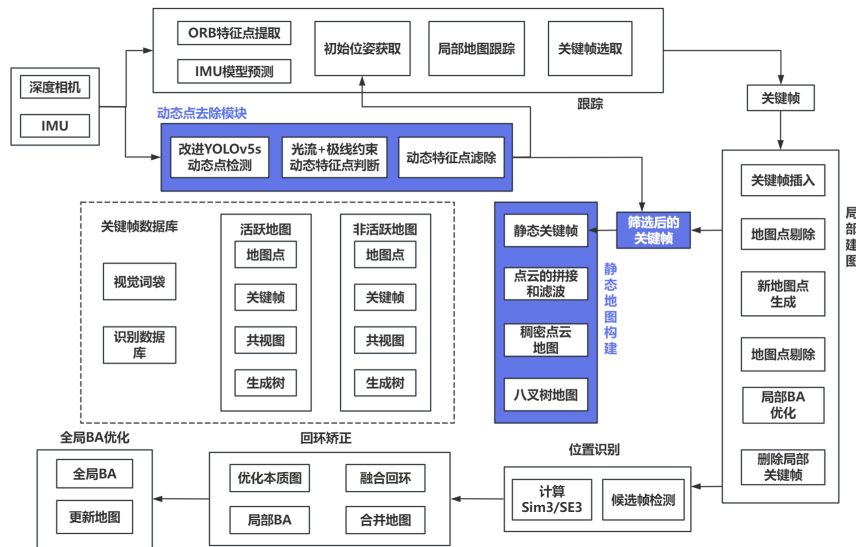


图 1 改进的 ORB-SLAM3 系统架构图

Fig.1 Improved ORB-SLAM3 system architecture

为提高视觉里程计在动态环境中的定位精度与建图完整性, 本文在传统 ORB-SLAM3 的视觉里程计中增加动态点去除模块, 该模块将改进的 YOLOv5s 目标检测算法与基于光流与极线约束的动态特征点剔除方法相结合。首先, 使用改进的 YOLOv5s 模型识别当前帧图像中的先验动态物体, 并提取各类物体的目

标边界框信息。对 RGB 图像进行预处理后, YOLOv5s 算法将 640×640 像素尺寸的图像输入网络结构以检测先验动态物体, 输出图像中存在的动态物体边界框坐标。在连续图像处理过程中, 系统将上一帧构建的地图特征点投影到当前帧, 并通过 ORB 特征点提取与匹配算法进行特征点匹配。根据匹配结果, 将检测框

内的特征点标记为动态特征点并剔除，而检测框外的特征点则标记为静态特征，继续用于定位与地图构建。基于这些静态特征点，系统计算出更准确的基础矩阵。随后，利用对极几何约束筛选所有已匹配的特征点，进一步区分并剔除潜在在动态物体的特征点。

1.2 YOLOv5s 目标检测算法及其改进

本文采用 YOLOv5s 目标检测算法来识别和剔除场景中的动态物体特征点，仅保留静态特征点以用于后续处理。YOLOv5s 是一种专为实现轻量化、高效目

标检测而设计的模型结构^[14]，主要由 Input、Backbone、Neck 以及 Head 四部分组成。虽然 YOLOv5s 能够识别出诸如电脑、桌椅和行人等多种物体类别，但在室内动态 SLAM 场景中，行人是最主要的动态干扰源，相较于静态物体，其位置变化会引发环境信息的快速更新，影响系统定位精度与地图构建质量。为此，本文对 YOLOv5s 算法进行优化，使其针对环境中的行人进行检测，以便更有效地识别出潜在的运动目标。改进的 YOLOv5s 框架如图 2 所示。

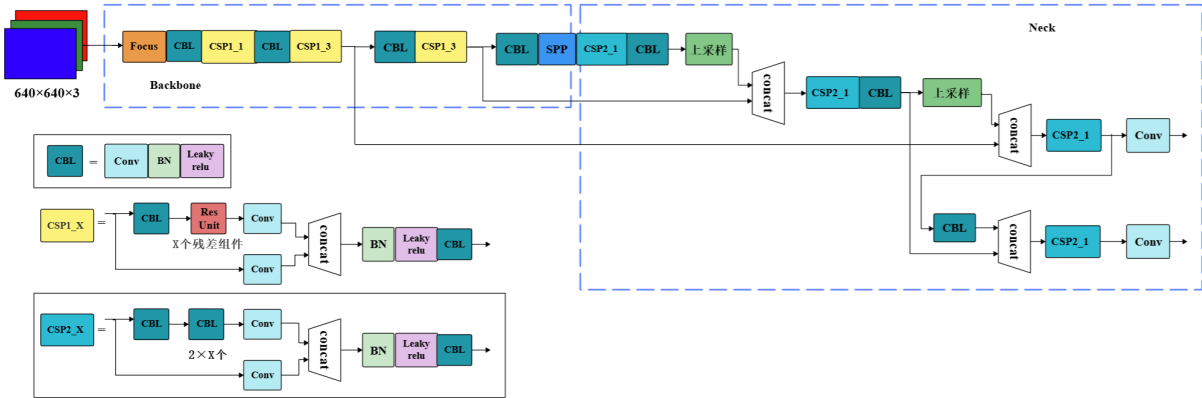


图 2 改进 YOLOv5s 网络结构

Fig.2 Improved YOLOv5s network structure

YOLOv5s 算法通过多尺度特征层预测实现目标检测，其输出结构包含大、中、小三种尺度的特征层。如图 2 所示，本文在 YOLOv5s 的 Neck 结构中移除了用于微小目标检测的特征融合模块，以简化模型架构并提高特定场景下的检测效率。改进后的网络在公开数据集上的测试结果如图 3 所示。图 3(a)为优化前的检测结果，其中包含多个类别的物体，如椅子、键盘和人；而图 3(b)仅检测出了行人作为可能的动态目标。优化后的模型在仅检测行人这一类动态目标时表现出了更高的检测可靠性和效率。



(a) 改进前 (b) 改进后
(a) Before improvement (b) After improvement

图 3 数据集中 YOLOv5s 算法改进效果对比

Fig.3 Comparison of improvement effect of YOLOv5s algorithm by using dataset

同样地，将 YOLOv5s 与改进后的网络应用于现实环境进行对比，检测结果如图 4 所示。可以看出，图 4(a)可以检测出走廊尽头的小目标，而图 4(b)在删除小目标对应的网络提取层之后仅检测出近处的行人目标。这表明，忽略图像中的小目标虽会在一定程度

上削弱对远距离或无关目标的敏感度，但能有效减少冗余检测对主要目标精度的干扰，从而提升整体检测的可靠和实用性。



(a) 改进前 (b) 改进后
(a) Before improvement (b) After improvement

图 4 现实环境中 YOLOv5s 算法改进效果对比

Fig.4 Comparison of improvement effect of YOLOv5s algorithm in real world environment

图 5 展示了经过优化的网络训练结果，以平均精度 (mean Average Precision, mAP) 为核心评估指标。由于此次改进专注于行人这一类检测目标，因此将 cls_loss 的值设为 0，表示类别损失在优化过程中被消除，使模型更专注于检测精度和速度的提升。为全面对比改进前后的模型性能，本文选取 mAP 0.5、mAP 0.5:0.95 及单张图像检测速度作为主要评估指标。其中，mAP 0.5 表示在交并比 (Intersection over Union,

IoU) 阈值设为 0.5 时, 若预测的边界框与真实边界框的重叠度达到或超过 0.5, 则视为一次正确检测; mAP 0.5:0.95 则是通过在多个 IoU 阈值 (从 0.5 到 0.95) 下评估模型并计算每个阈值 mAP 后求均值得到的指标。使用这两种指标可以判断模型在不同精确度条件下是否能保持较高的准确性和性能。图 5 中, 训练迭代次

数 (*epochs*) 为 3000 次, 其中标签为 train 的三张图分别代表训练过程中边界 (box)、目标 (objectness) 及分类 (classification) 损失的变化情况; 标签为 val 的三张图为验证集上的边界、目标和分类的损失变化情况; precision 和 recall 分别代表训练过程中精度和召回率的变化情况。

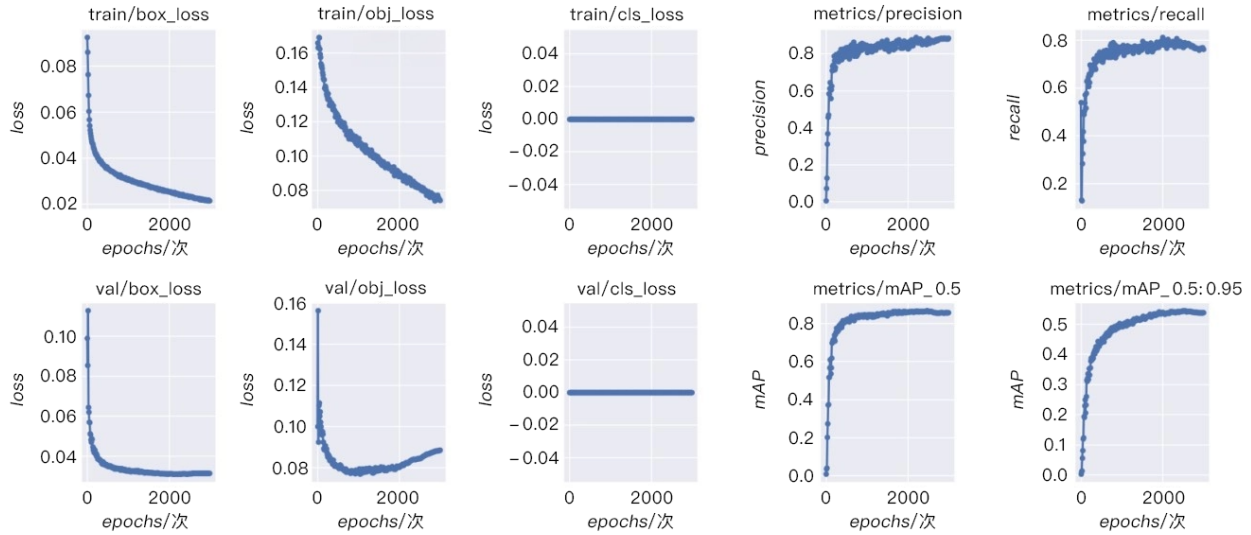


图 5 改进 YOLOv5s 的训练结果

Fig.5 Improved YOLOv5s training results

表 1 对比了传统 YOLOv5s 算法与本文改进算法在尺寸 (1,3,640,640) 下的非极大值抑制 (Non-Maximum Suppression, NMS) 处理速度。其中, (1,3,640,640) 中 1 表示 1 张图像, 3 表示 RGB 三种颜色通道, 两个 640 表示图像的高度和宽度均为 640 像素。可以看出, 改进的 YOLOv5s 网络在检测精度有

所提升的同时, 单张图像的前向传播时间及对图片预测结果进行非极大值抑制所需的时间均得到优化。尽管提升程度不大, 但对需要处理海量图像的 SLAM 系统而言, 这种提升增强了系统的实时性, 进而提升整体效率和响应速度。

表 1 改进前后网络模型的测试结果对比

Tab.1 Comparison of test results of the network model before and after improvement

算法	mAP 0.5	mAP 0.5:0.95	尺寸(1,3,640,640)下单张图像的处理速度
YOLOv5s	0.849	0.525	预处理耗时 0.3 ms, 推理耗时 8.1 ms, NMS 处理耗时 1.2 ms
改进的 YOLOv5s	0.862	0.543	预处理耗时 0.3 ms, 推理耗时 6.3 ms, NMS 处理耗时 0.9 ms

1.3 LK 光流法

光流法是一种在相邻帧间追踪物体位置变化的方法, 其核心原理基于视觉灰度保持相对稳定的假设^[15], 如图 6 所示, 其中 M_1 、 M_2 和 M_3 代表三帧连续的图像画面。该方法通过分析图像序列中的像素变化, 计算表征物体运动轨迹的二维矢量场。

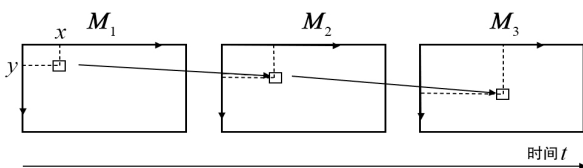


图 6 光流运动状态

Fig.6 Optical flow motion state

光流估计方法可分为稀疏光流与稠密光流两类。稀疏光流通过跟踪图像中的特定特征点, 估计这些点在连续帧间的运动^[16]; 而稠密光流则为图像中的每个像素计算光流, 可捕捉全局的运动信息^[17], 但其高计算复杂度会降低实时性。基于上述方法的对比, 本文采用 Lucas-Kanade (LK) 算法优化的稀疏光流进行运动估计。利用图像序列中连续两帧图像的灰度不变性原理来估计像素点的运动。通过最小化以下光流约束方程来估计每个像素点的运动:

$$I(x+i, y+j, t+1) \approx I(x, y, t) \quad (1)$$

其中, $I(\cdot)$ 为图像灰度函数, (x, y) 为当前帧像素坐标, (i, j) 为该像素点在两帧时间间隔内的运动矢量, t 为

时间变量。采用图像灰度一阶导数构建光流方程，并基于最小化平方误差函数建立 LK 模型，从而在短时间内完成图像处理。

1.4 动态特征的极线约束

极线约束是立体视觉与多视图几何的基本几何原理^[18]，主要描述从两个不同视点观测同一物理点时，其在两张相邻图像帧上的投影点必须满足的几何关系，如图 7 所示。图中 O_1 、 O_2 分别为两相机的光心，两光心与点 P 围成的平面 π 为极平面；光心 O_1 、 O_2 的连线为基线，基线 O_1O_2 与像平面 N_1 、 N_2 的交点 e_1 、 e_2 为极点；极平面 π 与像平面 N_1 、 N_2 所形成的交线 l_1 、 l_2 为极线。

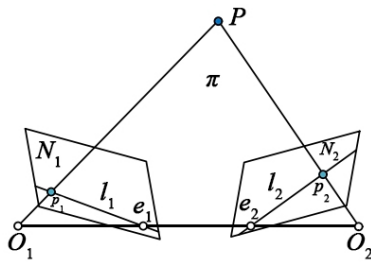


图 7 相邻图像帧的几何关系

Fig.7 Geometric relationships between adjacent image frames

假设世界坐标系中的点 P 投影到两个相机的成像平面上，可以得到一对相互匹配的特征点 p_1 和 p_2 ，其齐次坐标为：

$$\begin{cases} p_1 = [u_1 & v_1 & 1] \\ p_2 = [u_2 & v_2 & 1] \end{cases} \quad (2)$$

其中， u 和 v 分别为像素对应的水平和垂直坐标。

点 p_1 对应的极线 l_1 为：

$$l_1 = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Fp_1 \quad (3)$$

其中， F 表示对应的基本矩阵， X 、 Y 为极线 l_1 在图像平面坐标系下的方向分量， Z 为偏移项。

根据点到直线的距离公式，可以得到点 p_2 到极线 l_1 的距离 d 为：

$$d = \frac{|p_2 F p_1|}{\sqrt{X^2 + Y^2}} \quad (4)$$

其中， $\sqrt{X^2 + Y^2}$ 代表极线方向向量的长度。

设置阈值 θ ，如果 $d > \theta$ ，则滤除对应匹配点。

其中， θ 可通过下式得到：

$$\theta = \frac{\sum_{i=0}^N e^{-d_i}}{N} \quad (5)$$

其中， N 表示 ORB 特征点总提取数， d_i 表示第 i 个特征点到其极线的距离。基于上述方法，可快速剔除两帧间不匹配或违反极线约束（由动态因素导致）的特征点，实现动态干扰的有效滤除。

2 实验验证

2.1 动态特征点滤除测试

实验使用的电脑配置 Intel Core i5-12400F 处理器、NVIDIA GeForce RTX 3060 显卡及 12 GB 内存，软件环境为 Ubuntu 18.04 操作系统。为验证动态点去除算法有效性，本文基于 TUM RGB-D 数据集的 fr3 序列开展实验，该序列包含低动态场景的“sitting”序列和高动态场景的“walking”序列。这两种序列均存在四种相机运动模式：static（几乎静止）、xyz（三轴平移）、halfsphere（半球面运动）以及 rpy（三轴旋转）。本文选取 static 和 xyz 的两种极端运动模式进行测试，既评估动态环境干扰的影响，又验证相机大幅运动与环境动态共存时算法的鲁棒性。

图 8 和图 9 展示了 ORB_SLAM3 算法与本文算法在低动态场景和高动态场景中的特征点去除效果，其中“s”代表“sitting”序列，“w”代表“walking”序列，“Ours”代表本文动态特征点去除算法，275、288 等数字表示数据集中选取的特定帧编号。可以看出：低动态场景中，人物肢体轻微摆动区域的特征点被精确去除，而其余区域的静态特征则完整保留；高动态场景中，快速移动的行人动态特征被有效剔除，仅保留检测框内与背景重合的静态特征点。

为评估算法改进效果，本文对比了 ORB-SLAM3 算法与本文算法在低动态场景和高动态场景中的定位精度，结果如图 10 和图 11 所示。图中黑色线条为真实轨迹，蓝色线条为估计轨迹，红色区域为轨迹误差。



(a) s_static/275/ORB_SLAM3

(b) s_static/288/ORB_SLAM3

(c) s_static/435/ORB_SLAM3

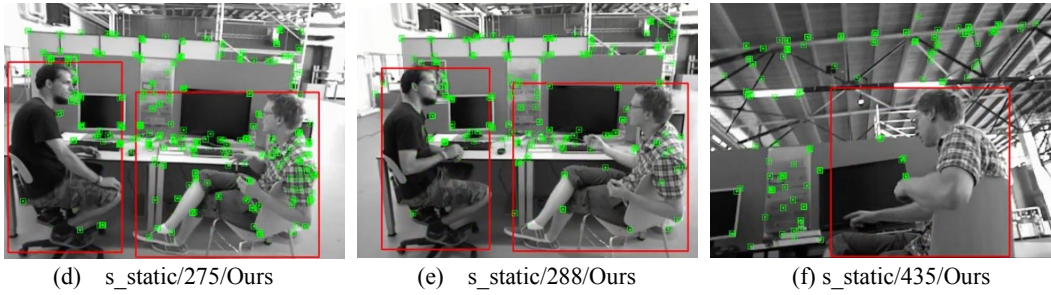


图 8 低动态特征点去除效果

Fig.8 Low dynamic feature point removal effect

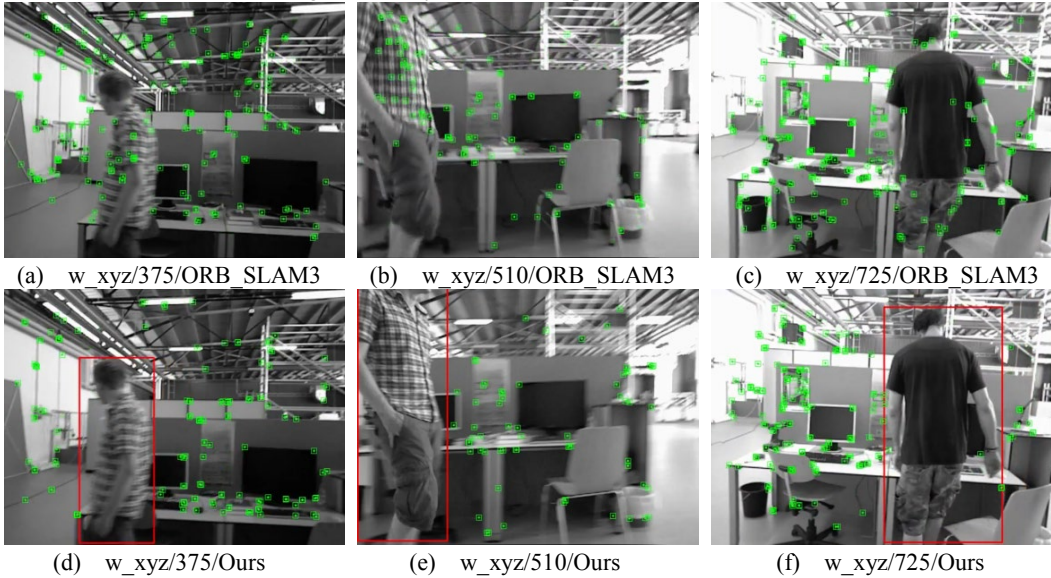


图 9 高动态特征点去除效果

Fig.9 High dynamic feature point removal effect

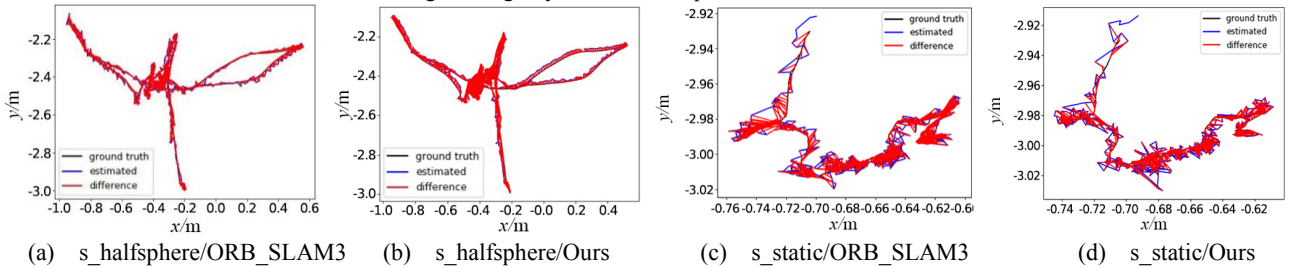


图 10 低动态场景绝对轨迹误差对比

Fig.10 Comparison of absolute trajectory error in low dynamic scene

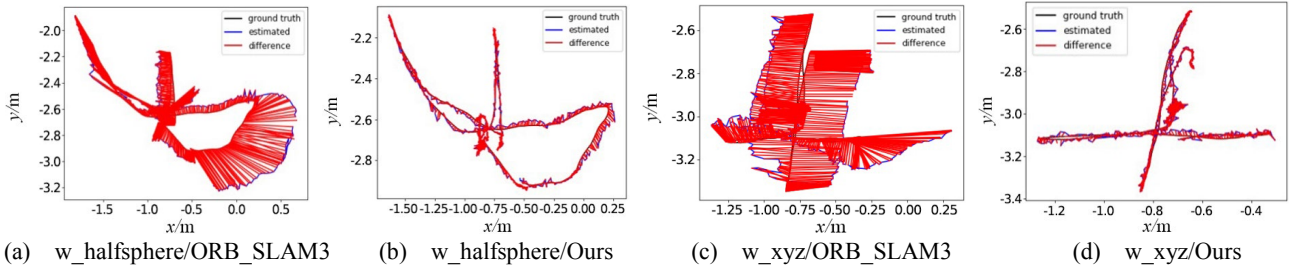


图 11 高动态场景绝对轨迹误差对比

Fig.11 Comparison of absolute trajectory error in high dynamic scene

图 10 与图 11 的实验结果表明: 在低动态场景中, 因运动幅度较小对建图干扰有限, ORB-SLAM3 与本文算法的轨迹误差较为接近; 而在高动态场景中, 快速运动的物体导致 SLAM 位姿估计偏差加剧, 二者轨迹误差显著扩大。本文算法通过动态特征点去除模块有效抑制干扰, 从而获得更优的定位精度。

通过图 12 和图 13 的轨迹误差对比及表 2、表 3 的定量分析可知: 在 s_halfsphere 序列中, 本文算法的三轴平均绝对误差均值为 0.022 m, 较 ORB-SLAM3 的 0.023 m 减小 4.3%, 其余三项指标均值分别减小 10.7%、4.0%和 14.6%; 在 s_static 序列中, 本文算法的三轴平均绝对误差均值为 0.005 m, 较 ORB-SLAM3

的 0.010 m 减小 50.0%，其他三项指标均值分别减小 46.4%、59.0%和 44.3%。实验结果表明，本文算法在

低动态场景中的各精度指标均优于 ORB-SLAM3。

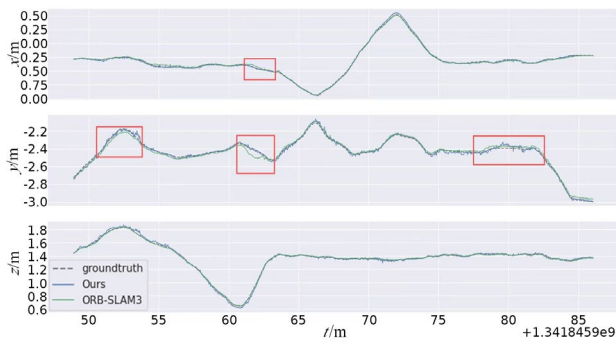


图 12 s_halfsphere 序列三轴误差对比

Fig.12 Three-axis error comparison of s_halfsphere sequence

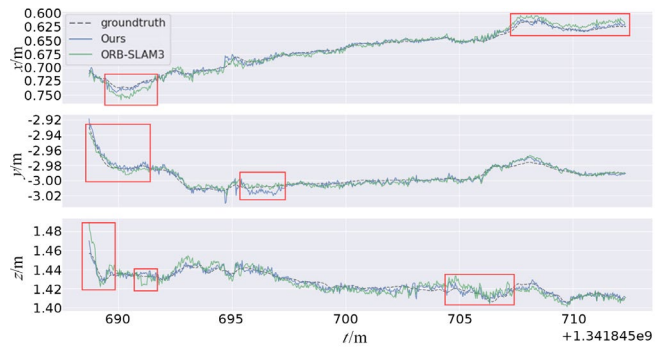


图 13 s_static 序列三轴误差对比

Fig.13 Three-axis error comparison of s_static sequence

表 2 s_halfsphere 序列定量结果 (单位: 米)

Tab.2 Quantitative results of s_halfsphere sequence (Unit: m)

算法	平均绝对误差	最大绝对误差	平均相对误差	最大相对误差
ORB-SLAM3 (x)	0.034	0.031	2.201	1.468
Ours (x)	0.031	0.024	2.075	1.095
ORB-SLAM3 (y)	0.033	0.111	1.249	4.273
Ours (y)	0.032	0.105	1.242	3.911
ORB-SLAM3 (z)	0.003	0.026	0.324	3.563
Ours (z)	0.003	0.022	0.306	2.942

表 3 s_static 序列定量结果 (单位: 米)

Tab.3 Quantitative results of s_static sequence (Unit: m)

算法	平均绝对误差	最大绝对误差	平均相对误差	最大相对误差
ORB-SLAM3 (x)	0.011	0.029	1.611	4.716
Ours (x)	0.004	0.017	0.662	2.778
ORB-SLAM3 (y)	0.013	0.019	0.916	0.830
Ours (y)	0.007	0.011	0.241	0.427
ORB-SLAM3 (z)	0.008	0.037	0.590	2.561
Ours (z)	0.005	0.018	0.376	1.311

如图 14 和图 15 所示，在高动态场景中，ORB-SLAM3 的估计轨迹与真实轨迹偏差显著增大，而本文算法能有效抑制误差。结合表 4 与表 5 的定量分析可知：在 w_halfsphere 序列中，本文算法的三轴平均绝对误差均值为 0.008 m，较 ORB-SLAM3 的 0.117 m 减小 93.2%，其余三项指标均值分别减小

91.0%、54.7%和 57.8%；在 w_xyz 序列中，本文算法的三轴平均绝对误差均值为 0.004 m，较 ORB-SLAM3 的 0.089 m 减小 95.5%，其余三项指标均值分别减小 94.8%、88.6%和 65.5%。实验结果表明，ORB-SLAM3 在高动态场景中的定位精度明显不足，而本文算法能更有效地处理物体高速且大幅度运动带来的干扰。

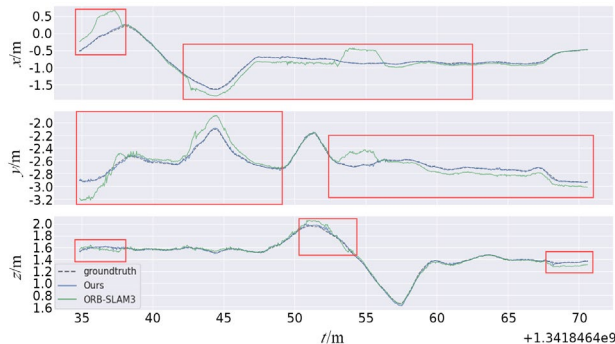


图 14 w_halfsphere 序列三轴误差对比

Fig.14 Three-axis error comparison of w_halfsphere sequence

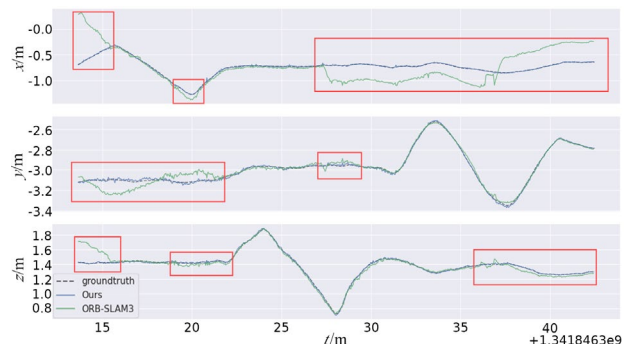


图 15 w_xyz 序列三轴误差对比

Fig.15 Three-axis error comparison of w_xyz sequence

表 4 w_halfsphere 序列定量结果 (单位: 米)

Tab.4 Quantitative results of w_halfsphere sequence (Unit: m)

算法	平均绝对误差	最大绝对误差	平均相对误差	最大相对误差
ORB-SLAM3 (x)	0.205	0.629	0.766	5.764
Ours (x)	0.002	0.009	0.358	2.639
ORB-SLAM3 (y)	0.135	0.523	0.872	8.269
Ours (y)	0.018	0.075	0.436	3.284
ORB-SLAM3 (z)	0.011	0.080	0.714	4.933
Ours (z)	0.004	0.027	0.271	2.080

表 5 w_xyz 序列定量结果 (单位: 米)

Tab.5 Quantitative results of w_xyz sequence (Unit: m)

算法	平均绝对误差	最大绝对误差	平均相对误差	最大相对误差
ORB-SLAM3 (x)	0.178	0.887	2.589	3.443
Ours (x)	0.001	0.008	0.206	1.489
ORB-SLAM3 (y)	0.053	0.147	1.786	4.773
Ours (y)	0.005	0.035	0.183	1.280
ORB-SLAM3 (z)	0.035	0.284	2.772	6.150
Ours (z)	0.005	0.027	0.425	2.193

为系统评估基于动态特征点去除的 ORB-SLAM3 改进算法性能, 在 TUM RGB-D 数据集的 fr3 序列上对比测试了 ORB-SLAM3、DS-SLAM、DynaSLAM 及本文算法, 通过绝对轨迹误差 (Absolute Trajectory

Error, ATE)、相对位姿误差 (Relative Pose Error, RPE) 和算法平均运行时间三项指标进行量化分析, 结果如表 6 和表 7 所示。

表 6 不同算法在 fr3 序列的定位精度对比

Tab.6 Comparison of localization accuracy of different algorithms in fr3 sequence

序列	ORB-SLAM3		DS-SLAM		DYnaSLAM		Ours	
	ATE/m	RPE/m	ATE/m	RPE/m	ATE/m	RPE/m	ATE/m	RPE/m
w_xyz	0.274	0.182	0.023	0.033	0.017	0.028	0.024	0.016
w_halfsphere	0.553	0.256	0.038	0.046	0.024	0.035	0.049	0.022
w_static	0.039	0.071	0.020	0.029	0.007	0.012	0.002	0.046
w_rpy	0.589	0.215	0.039	0.048	0.043	0.085	0.046	0.019
s_static	0.012	0.025	0.013	0.022	0.014	0.025	0.007	0.003
s_halfsphere	0.040	0.035	0.032	0.026	0.028	0.044	0.025	0.025

表 7 算法平均运行时间 (单位: 秒)

Tab.7 Average algorithm running time (Unit: s)

算法	平均计算时间
ORB-SLAM3	43.2
DS-SLAM	124.2
DYnaSLAM	382.5
Ours	86.7

由表 6 和表 7 可知, 在低动态场景 s_static 与 s_halfsphere 中, 与传统的 ORB-SLAM3 相比, 本文算法性能提升较低; 但在高动态场景 w_xyz、w_halfsphere 和 w_rpy 中, 由于存在快速移动的目标, 会对位姿估计产生较大影响, 使得 ORB-SLAM3 的轨迹误差较大。结合表 6 中的数据可以算出, 本文算法的 ATE 和 RPE 分别平均下降了 91.5% 和 91.3%。实验结果表明本文算法能够有效提升 SLAM 系统定位和地

图构建的精度。另外, 与 DYnaSLAM 和 DS-SLAM 相比, 本文改进的 SLAM 系统在定位精度上与 DYnaSLAM 相当, 但运行时间快了近 5 倍; 与 DS-SLAM 相比, 轨迹误差相近, 但计算耗时降低了 30.2%。在实时性方面, 本文算法具有显著优势。综合运行时间和定位精度, 本文改进的动态视觉 SLAM 系统展现出更优的综合性能。

2.2 真实环境稠密点云地图构建测试

为了验证建图算法在真实动态环境下的有效性, 在如图 16 所示的办公室场景和图 17 所示的走廊场景中, 分别使用传统 ORB-SLAM3 和本文算法进行建图实验。在办公室场景中行人快速移动, 建图效果对比如图 18 所示。其中, 图 18(a)与图 18(b)使用了未滤除动态特征的关键帧进行地图构建, 导致行人在地图中

清晰可见并对建图质量造成干扰;而图 18(c)与图 18(d)则基于滤除了动态特征点的静态关键帧构建点云,行人轨迹被有效剔除,使地图与静态环境的匹配度更高。

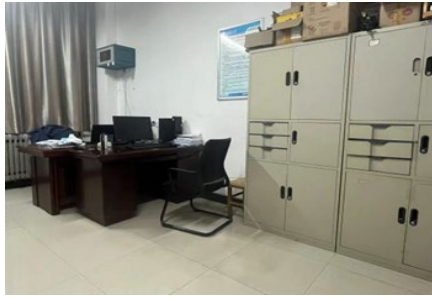


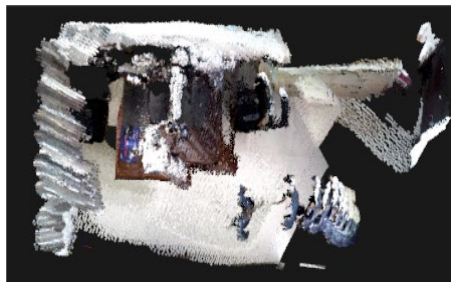
图 16 办公室场景
Fig.16 Office scene



图 17 走廊场景
Fig.17 Corridor scene



(a) ORB-SLAM3 水平视角
(a) ORB-SLAM3 horizontal perspective



(b) ORB-SLAM3 俯视角
(b) ORB-SLAM3 top-down perspective



(c) Ours 水平视角
(c) Ours horizontal perspective



(d) Ours 俯视角
(d) Ours top-down perspective
图 18 办公室场景下稠密地图构建

Fig.18 Construction of dense maps in office scene

在走廊场景中,行人和机器人相向而行,并做奔跑、肢体摆动等动作。由于走廊环境存在一定的局限性,所以无法展现俯视视角。在走廊环境下分别使用关键帧和滤除动态信息的关键帧所构建的稠密点云地图如图 19 所示。



(a) ORB-SLAM3



(b) Ours

图 19 走廊稠密地图构建

Fig.19 Construction of dense maps in corridor scene

通过以上建图效果看出,在图 19(a)中,行人的快速移动会在地图中形成重叠的动态伪影,并遮挡部分环境细节,导致地图可读性下降;而在图 19(b)中,环境信息清晰完整,背景板、桌子等静态物体没有被遮挡或重影干扰,环境整体轮廓清晰可见。

根据表 8 所示的 EVO 工具轨迹误差评估结果,本文算法在办公室和走廊场景均实现显著精度提升:办公室场景 ATE 和 RPE 分别减小 55.2%、93.7%;走廊场景 ATE 和 RPE 分别减小 24.3%、40.2%。实验表明,本文算法构建的地图与真实场景结构保持了更高一致,验证了算法的有效性。

表8 不同场景中的轨迹误差对比
Tab.8 Comparison of trajectory error in different scenarios

算法	办公室场景		走廊场景	
	ATE/m	RPE/m	ATE/m	RPE/m
ORB-SLAM3	0.058	0.143	0.037	0.087
Ours	0.026	0.009	0.028	0.052

3 结论

针对传统视觉SLAM算法在动态环境中难以有效剔除快速移动目标导致定位精度低及地图构建不完善的问题, 本文提出了一种基于动态目标检测的改进ORB-SLAM3算法。首先对YOLOv5s网络进行精简, 保留中大目标检测分支并仅检测行人, 从而更加高效地识别室内动态干扰源; 随后结合LK光流与极线几何, 对先验动态区域以外的特征点进行二次过滤, 最大程度减少高动态场景下的错误匹配; 最后通过在静态关键帧上进行稠密建图, 剔除动态特征点对三维地图的干扰。在公开的TUM RGB-D数据集fr3序列中进行测试得出: 本文算法在低动态场景下保持与ORB-SLAM3相当的精度, 而在高动态场景w_xyz、w_halfsphere、w_rpy下, ATE与RPE较ORB-SLAM3分别平均减小91.5%和91.3%, 显著优化了动态环境定位性能。真实场景测试进一步表明: 办公室场景中ATE与RPE分别减小55.2%和93.7%, 走廊场景中相应指标分别减小24.3%和40.2%, 所建地图与真实结构高度吻合, 验证了算法在动态目标检测、特征过滤及场景建模方面的有效性与鲁棒性。本文主要针对行人这一动态目标进行检测, 针对其他动态目标的检测将是后续研究的重点。

参考文献 (References):

- [1] Smith R C, Cheeseman P. On the representation and estimation of spatial uncertainty[J]. The International Journal of Robotics Research, 1986, 5(4): 56-68.
- [2] 徐晓苏, 高佳誉, 周帅, 等. 基于改进GICP配准的激光-惯性SLAM算法[J]. 中国惯性技术学报, 2023, 31(08): 814-822.
Xu X, Gao J, Zhou S, et al. Lidar-inertial SLAM based on improved GICP registration[J]. Journal of Chinese Inertial Technology, 2023, 31(08): 814-822.
- [3] 赵伟博, 田军委, 王沁, 等. 基于点线特征融合的视觉惯性SLAM算法[J]. 计算机应用研究, 2023, 40(02): 445-449.
Zhao W, Tian J, Wang Q, et al. Visual inertial SLAM algorithm based on point-line feature fusion[J]. Application Research of Computers, 2023, 40(02): 445-449.
- [4] 闫河, 王旭, 雷秋霞, 等. 动态场景结合稀疏场景流和加权特征的视觉SLAM方法[J]. 中国惯性技术学报, 2024, 32(09): 891-897.
Yan H, Wang X, Lei Q, et al. Visual SLAM method combining sparse scene flow and weighted features in dynamic environment[J]. Journal of Chinese Inertial Technology, 2024, 32(09): 891-897.
- [5] Campos C, Elvira R, Rodríguez J J G, et al. ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM[J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-1890.
- [6] Liu Y, Miura J. RDS-SLAM: Real-time dynamic SLAM using semantic segmentation methods[J]. IEEE Access, 2021, 9: 23772-23785.
- [7] Lee J, Back M, Hwang S S, et al. Improved real-time monocular SLAM using semantic segmentation on selective frames[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 24(3): 2800-2813.
- [8] Zhang B, Ma X, Ma H, et al. DynPL-SVO: A robust stereo visual odometry for dynamic scenes[J]. IEEE Transactions on Instrumentation and Measurement, 2024, 73: 1-10.
- [9] 刘胤真, 徐向荣, 张卉, 等. 动态场景下基于语义和几何约束的视觉SLAM算法[J]. 信息与控制, 2024, 53(03): 388-399.
Liu Y, Xu X, Zhang H, et al. Visual SLAM algorithm based on semantic and geometric constraints under dynamic scenes[J]. Information and Control, 2024, 53(03): 388-399.
- [10] Lei X, Wen X, Li Z. A multi-target cow face detection model in complex scenes[J]. The Visual Computer, 2024: 1-22.
- [11] 蔡显奇, 王晓松, 李玮. 一种室内弱纹理环境下的视觉SLAM算法[J]. 机器人, 2024, 46(03): 284-293+304.
Cai X, Wang X, Li W. A visual SLAM algorithm in indoor weak texture environment[J]. Robot, 2024, 46(03): 284-293+304.
- [12] Yuan X, Kuerban A, Chen Y, et al. Faster light detection algorithm of traffic signs based on YOLOv5s-A2[J]. IEEE Access, 2022, 11: 19395-19404.
- [13] Zang Q, Zhang K, Wang L, et al. An adaptive ORB-SLAM3 system for outdoor dynamic environments[J]. Sensors, 2023, 23(3): 1359.
- [14] Ashraf A H, Imran M, Qahtani A M, et al. Weapons detection for security and video surveillance using CNN and YOLO-v5s[J]. Computers, Materials & Continua, 2022, 70(2): 2761-2775.
- [15] Qin L, Wu C, Chen Z, et al. RSO-SLAM: A robust semantic visual SLAM with optical flow in complex dynamic environments[J]. IEEE Transactions on Intelligent Transportation Systems, 2024, 25(10): 14669-14684.
- [16] 陈志环, 王祖傲, 李想成. 室内动态场景下基于深度相机的VSLAM方法[J]. 中国惯性技术学报, 2023, 31(04): 390-400.
Chen Z, Wang Z, Li X, et al. Visual SLAM method based on depth camera in indoor dynamic scene[J]. Journal of Chinese Inertial Technology, 2023, 31(04): 390-400.
- [17] 王磊, 张茗宇, 潘明然, 等. 基于多元离群点检测的动态目标去除SLAM方法[J]. 探测与控制学报, 2024, 46(05): 64-70.
Wang L, Zhang M, Pan M, et al. Dynamic target removal SLAM method for multivariate outlier detection[J]. Journal of Detection & Control, 2024, 46(05): 64-70.
- [18] Zhou C, Su S, Chen Q, et al. E3CM: Epipolar-constrained cascade correspondence matching[J]. Neurocomputing, 2023, 559: 126788.