

屈佩佩, 曾新华, 龙伟, 等. 动态场景中基于目标检测的视觉 SLAM 算法研究[J]. 智能计算机与应用, 2026, 16(3): 19-26.
DOI: 10.20169/j.issn.2095-2163.24043003

动态场景中基于目标检测的视觉 SLAM 算法研究

屈佩佩¹, 曾新华^{2,3}, 龙伟¹, 蒋林华¹, 林传渠¹

(1 湖州师范学院 信息工程学院, 浙江 湖州 313000; 2 复旦大学 工程与应用技术研究院, 上海 200433;

3 复旦大学 义乌研究院, 浙江 义乌 322000)

摘要: 在高动态场景区域, 移动机器人在 ORB-SLAM3 算法跟踪线程提取的特征点误匹配率高, 造成系统精度不高问题。对 ORB-SLAM3 系统的前端视觉里程计部分实现设计了双线程并行的方法, 通过 Socket 实现 2 个系统在同一主机中信息交互, 即在目标检测线程中进行目标分类和特征点筛选, 在跟踪线程中提取特征。通过结合对极几何约束算法, 利用帧间匹配极线距离进一步滤除特征点, 增强系统在动态环境下的鲁棒性。对视觉 SLAM 系统进行优化加速, 一方面采用基于信息熵的特征提取与匹配算法, 在提升视觉 SLAM 系统的稳定性的同时能提高特征点提取效率, 另一方面对更加轻量化的 YOLOv5 算法进行微调, 实现快速目标检测并保持较高精度。为了验证本文算法的有效性, 使用慕尼黑理工大学的 TUM 数据集对比 ORB-SLAM3, 表明在高动态序列中本文方法的绝对轨迹误差和相对轨迹误差大都减少明显, 与 DS-SLAM、DynaSLAM 和 YOLO SLAM 相比, 在保持精度的同时, 实时性能方面有一定的提升, 该算法具有满足实际工程要求的潜力。

关键词: 视觉 SLAM; 轻量化目标检测; 动态场景; 对极几何; 特征提取

中图分类号: TP242.2

文献标志码: A

文章编号: 2095-2163(2026)03-0019-08

Research on visual SLAM algorithm based on object detection in dynamic scenes

QU Peipei¹, ZENG Xinhua^{2,3}, LONG Wei¹, JIANG Linhua¹, LIN Chuanqu¹

(1 School of Information Engineering, Huzhou University, Huzhou 313000, Zhejiang, China;

2 Academy for Engineering and Technology, Fudan University, Shanghai 200433, China;

3 Yiwu Research Institute, Fudan University, Yiwu 322000, Zhejiang, China)

Abstract: In high dynamic scene areas, the mismatch rate of feature points extracted by the ORB-SLAM3 algorithm tracking thread for mobile robots is high, resulting in low system accuracy. A dual thread parallel method is designed for the front-end visual odometry part of the ORB-SLAM3 system, and information exchange between the two systems in the same host is achieved through Socket. That is, target classification and feature point filtering are performed in the target detection thread, and features are extracted in the tracking thread. By combining the epipolar geometric constraint algorithm and utilizing inter frame matching epipolar distance, feature points are further filtered out to enhance the robustness of the system in dynamic environments. Optimizing and accelerating the visual SLAM system by adopting an information entropy based feature extraction and matching algorithm, which not only improves the stability of the visual SLAM system, but also improves the efficiency of feature point extraction. On the other hand, the more lightweight YOLOv5 algorithm is fine-tuned to achieve fast object detection and maintain high accuracy. In order to verify the effectiveness of the algorithm proposed in this paper, the TUM dataset from the Technical University of Munich is used to compare with ORB-SLAM3. The results demonstrate that the absolute and relative trajectory errors of the proposed method are significantly reduced in high dynamic sequences. Compared with DS-SLAM, Dyna SLAM, and YOLO-SLAM, this algorithm has the potential to meet practical engineering requirements while maintaining accuracy.

Key words: visual SLAM; lightweight object detection; dynamic scenes; epipolar geometry; feature extraction

基金项目: 国家自然科学基金(62373148, 61972016)。

作者简介: 屈佩佩(1998—), 女, 硕士研究生, 主要研究方向: 智能信息处理技术; 龙伟(1978—), 男, 博士, 讲师, 主要研究方向: 目标检测与跟踪, 机器学习; 蒋林华(1977—), 男, 博士, 教授, 主要研究方向: 机器学习与人工智能, 自动驾驶; 林传渠(1998—), 男, 硕士研究生, 主要研究方向: 智能信息处理技术。

通信作者: 曾新华(1976—), 男, 博士, 特聘研究员, 主要研究方向: 视频异常检测, 目标识别与图像分割。Email: 2418144702@qq.com。

收稿日期: 2024-04-30

0 引言

同时定位与地图构建 (Simultaneous Localization and Mapping, SLAM) 技术^[1]是一种在未知环境中,通过视觉传感器实现移动机器人自身定位并构建周围环境地图的技术。这一技术在自主导航领域得到广泛应用,包括自动驾驶车辆、无人机和割草机器人等。然而,目前大多数 SLAM 研究基于静态场景的假设,而在面对动态场景时,传统 SLAM 系统面临一系列挑战。动态物体的存在可能对位姿估计造成干扰,导致轨迹误差和尺度漂移等问题。因此,如何有效地处理动态特征点、实现准确的位姿估计成为视觉 SLAM 技术面临的关键问题^[2]。

动态视觉 SLAM 主要分为 2 种研究路线。一种方法是通过直接利用图像中包含的信息以及几何约束来减轻场景中移动物体的影响。另一种方法是语义分割算法或目标检测算法与 SLAM 相结合来解决动态场景中 SLAM 的异常点问题。语义方法可以更直接、准确地获得动态对象所在的区域。

利用对极几何滤除动态的特征点,例如艾青林等学者^[3]提出了将改进几何与运动约束相结合的方法,先将特征点进行分类,再利用几何约束剔除动态点。Wang 等学者^[4]提出通过对深度图像聚类与对极几何相结合的方法识别场景中的异常点,但当场景中存在数量过多的动态点时存在一定误差。

与基于几何的动态视觉 SLAM 系统相比,基于深度学习的系统具有更高的精度和稳定性,可应用于更复杂的环境。如基于 ORB-SLAM3^[5]的 RDS-SLAM^[6]框架通过使用模型的分割结果初始化移动对象的移动概率^[7],利用数据关联算法来去除动态点。PSPNET-SLAM^[8]在 ORB-SLAM2 中加入语义分割网络金字塔场景解析网络 (PSP Net)^[9],得到具有语义信息的关键帧,并将语义信息用于动态区域的检测和动态特征点的拒绝。以上使用分割模型的动态对象轮廓检测的有效性在很大程度上取决于底层语义分割网络的性能和质量,这给同时满足速度和精度要求带来了挑战。因此,在基于深度学习的 SLAM 领域仍然有待深入的探讨研究。为了弥补这些缺点,最近的研究工作已经开始将几何和语义方法结合来有效去除动态特征点。例如, Dyna-SLAM^[10]是一种具有动态对象识别和修复功能的视觉 SLAM 系统,利用 Mask-RCNN^[11]对先验动态对象进行检测和滤除,并结合多视图几何约束来定位未检测到的动态对象。然而,由于该实例分割网络

的运行效率低下,系统无法实时运行。Wu 等学者^[12]提出了一种基于目标检测的动态视觉 SLAM 系统 - YOLO-SLAM。该系统提出轻量级的 DarkNet19-YOLOv3 网络主要用于加速语义信息的生成。但 DarkNet19-YOLOv3 模型依旧相对复杂,需要消耗的资源太多,达不到实时性要求。

因此,本文提出了一种基于信息熵特征提取、目标检测和基于对极几何约束的异常点滤除机制的语义 SLAM 系统,名为 DRV-SLAM,通过使用本文提出信息熵特征提取进行全局高质量特征提取,使用 YOLOv5_FNet 来检测动态环境中潜在的运动对象,并根据目标图像占比来判断是否结合基于对极几何^[13]的异常点检测机制来进一步识别和剔除动态点。实验表明,所提出的方法使得系统在动态环境中的定位精度得到提高,估计位姿误差得到有效降低。

1 系统框架

本文以 ORB-SLAM3 为基础框架,提出基于目标检测和对极几何约束的异常值滤除机制的视觉 SLAM 算法,结合目标检测,在原有的视觉里程计、局部建图、回环检测 3 个线程的基础上添加动态特征点滤除线程。总体框架如图 1 所示。由图 1 可知,首先 RGB 图像送入跟踪线程和动态特征点滤除线程,使用 YOLOv5s_FNet 进行检测获得语义信息,然后通过基于目标检测框占比确定是否需要进一步对极几何约束进一步确定特征点并滤除。尽可能保留大量静态特征点,进行特征匹配和位姿估计。

1.1 基于信息熵的特征提取

在 SLAM 系统中,特征点的提取质量与定位精度密切相关,低质量特征点可能在相邻帧中不稳定,从而导致系统跟踪失败。

旨在实现对图像纹理的自适应特征点提取,以达到更高的图像识别准确率。为此,提出了一种基于图像信息熵的快速特征提取算法,该方法能够根据图像本身纹理信息,依据信息熵^[14]占比自适应地分配提取点数。可用如下公式进行计算:

$$n_w = \text{ceil} \left(N \times \frac{E_w}{\sum_{w=1} E_w} \right) \quad (1)$$

其中, n_w 表示某层级金字塔中第 w 网格内需要提取的特征点数量; N 表示金字塔层级需要提取的特征点数量; E 表示所求信息熵。

本文算法具体步骤如下:

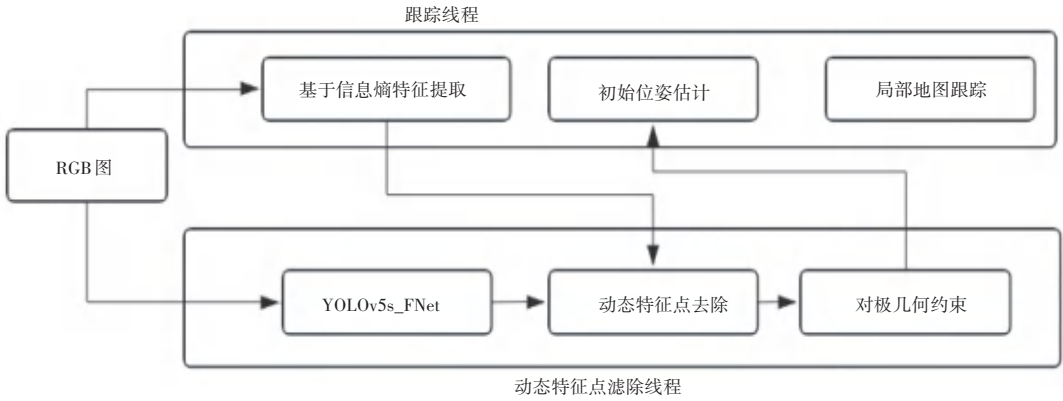


图 1 改进前端视觉里程计

Fig. 1 Improved front-end visual odometer

(1) 首先将图像灰度化, 为满足特征提取的尺度不变性构建 6 层金字塔, 每层需提取的特征点数量根据金字塔不同层级的面积计算, 然后根据 $m \times n$ 像素大小进行网格划分。

(2) 遍历金字塔每一层, 计算每层金字塔图像网格对应的信息熵 E_j 。这里用到的公式为:

$$E = - \sum_{i=1}^M \sum_{j=1}^N p_{ij} \log_2 p_{ij} \quad (2)$$

(3) 计算出所有网格的信息熵值后进行排序, 提取其中最大的信息熵值 MaxEntropy 并设置信息熵评估因子 (factor) 控制提取的目标区域。对此可以表示为:

$$\text{Entropy} \geq \text{factor} \times \text{MaxEntropy} \quad (3)$$

若该网格信息熵满足上述公式, 则该网格为目标区域, 需要对其进行快速角点检测。若网格信息熵不满足设定阈值, 则该网格为非目标区域, 直接滤除该区域。

(4) 计算出不同尺度空间中各目标图像网格需要提取的特征点数量, 将非目标图像块的特征点数

分配给信息熵值高的图像网格区域。

(5) 设置 FAST 特征提取的 2 个阈值, 首先用初始阈值对每个网格图像块进行 FAST 特征点提取, 判断提取的特征点数量是否足够, 如果足够则选取响应值排序较前的点作为特征点; 如果某个网格内提取的特征点数不够, 则降低阈值再进行 FAST 特征提取。

(6) 根据提取到的特征点集计算特征点描述子。

1.2 轻量化 YOLOv5s 目标检测网络

YOLO (You Only Look Once)^[15] 是一种在检测速度和精度上都有优异表现的单阶段目标检测算法。根据网络中卷积核的数量, YOLOv5s^[16] 分为 YOLOv5x、YOLOv5l、YOLOv5m、YOLOv5n 和 YOLOv5s。这 5 个模型在运算量和模型参数上逐渐减少。研究中选用了 YOLOv5s 模型, 在满足系统检测精度的同时, 具有较少参数量, 从而实现更快的模型速率。在本文中, 对 YOLOv5s 进行了优化来提高其性能。改进模型如图 2 所示。

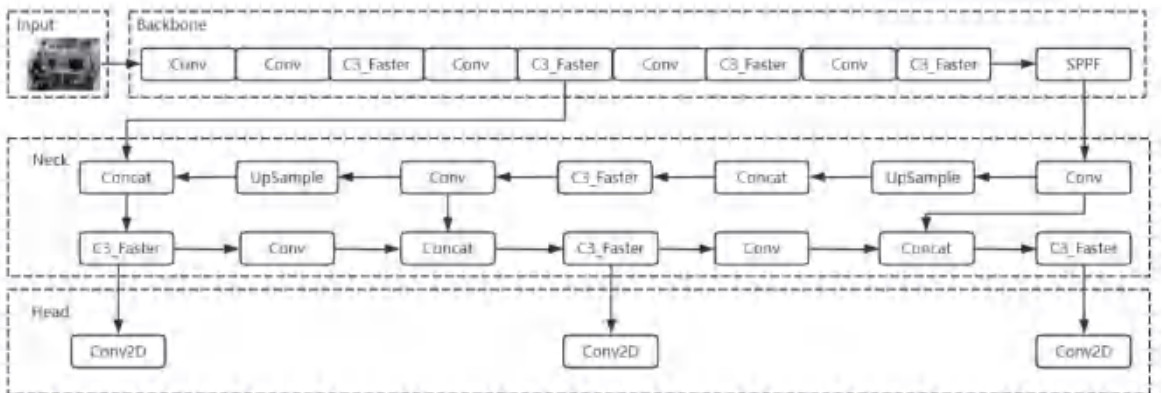


图 2 改进 YOLOv5s 模型

Fig. 2 Improved YOLOv5s model

YOLOv5s 模型 Backbone 和 Neck 采用 C3 模块, C3 模块包括 3 个 Conv 块,其增强了网络的深度和感受野^[17],导致目标检测进程缓慢。在 ORB-SLAM3 框架中引入目标检测网络实现定位精度的提升,需要增加一个线程,线程的运行时间对整个系统的实时性影响较大,参数复杂且模型庞大的检测网络却难以实现。为了实现轻量级、精度和实时性之间的平衡,研究在主干网络 C3 模块引入 FasterNet^[18],构成 C3-Faster。FasterNet 由一个特殊模块 PConv 和 2 个卷积层构成,通过并行计算卷积方式以减少计算冗余和内存访问,实现更低的延迟。PConv 只在特定输入通道应用常规卷积,同时保持其他通道不变。FLOPs_{PConv} 的定义公式如下:

$$\text{FLOPs}_{\text{PConv}} = h \times w \times k^2 \times c_p^2 \quad (4)$$

其中, h 和 w 分别表示特征的宽度和高度; k 表示卷积核大小; c_p 表示常规卷积操作的通道数。通常比率 $r = c_p/c = 1/4$,这意味着 PConv 的 FLOP 仅为常规卷积的 FLOP 的 1/16。FLOP 的减少大大提高了网络的推理速度,同时减少了模型的总体参数计算。

为了充分发挥所有通道信息的有效性,在部分卷积(Partial Convolution, PConv)之后引入了点积卷积(PointWise Convolution, PWConv)。FasterBlock 将层归一化(LN)替换为批归一化(BN),用 GELU 作为激活函数,减少运行时间,实现更快的推理速度。FasterNet 模块如图 3 所示。

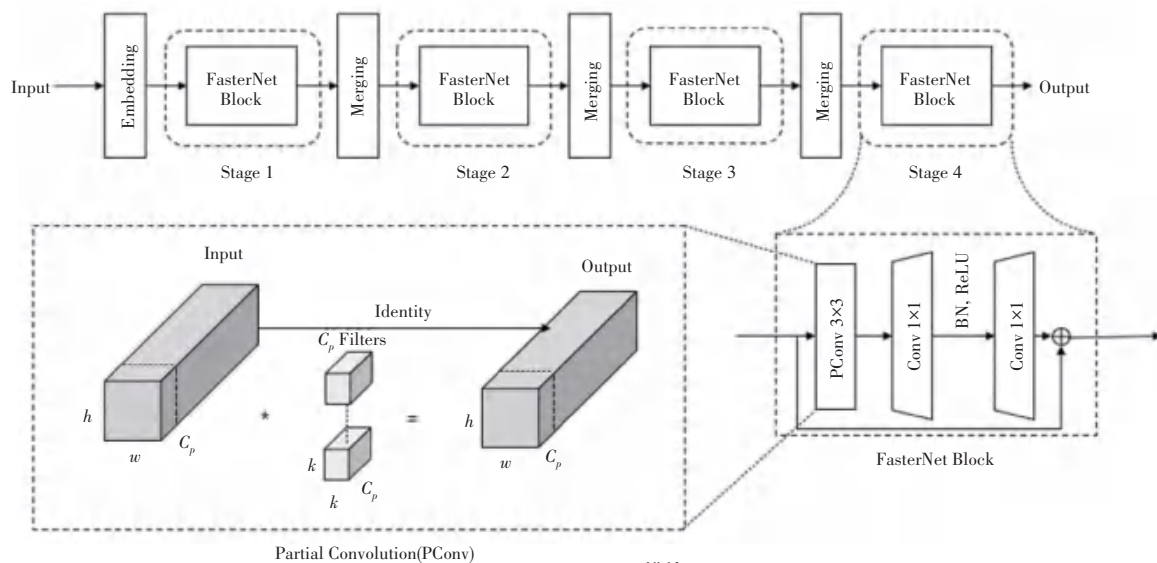


图3 FasterNet 模块

Fig. 3 FasterNet module

1.3 异常点滤除机制

ORB-SLAM3 使用视觉里程计提取连续帧中用于姿态估计的特征点,然而,当前帧动态目标框占比较大时,难以避免提取到动态特征点,这将导致位姿估计偏差,使得在动态场景下累计误差不断增加^[19]。

通过 ORB 特征提取算法和上文提出的 YOLOv5s_FNet 网络对帧进行图像处理,定位图像中特征点和潜在动态物体的位置,并返回由 4 个角点坐标组成的矩形。此时,矩形框包括动态物体、潜在运动物体以及背景中的静态特征点。因此需要去除动态特征点,同时最大限度地减少静态特征点的错误去除,减少定位误差。本文提出基于动态目标检测框占比和对极几何约束识别动态特征点并剔除。研究发现,经由目标检测输出的目标框边缘通常存在大量 FAST 检测得到的特征点,为了尽可能保留更多的高质量静态点,提升定位精度和系统鲁棒性,本文在得

到的边界框各方向进行拓展。

基于动态目标检测框占比和对极几何约束的异常点滤除机制流程如下。未对所有目标框内的特征点都使用对极几何算法进行重验证,为满足实时性要求通过基于当前帧先验对象占当前图像大小来确定。如果当前帧目标框占比超过 40%,则对框内经由 FAST 检测得到特征点进行边界框拓展,再利用对极几何的进一步约束,对特征点进行区分。相反,当前帧先验对象框占比低于 40%,则直接去除目标框内所有特征点,在这种情况下认为已经拥有足够的静态特征点用于匹配。下面介绍对极几何算法流程。

对极几何约束如图 4 所示。图 4 中, O_1 和 O_2 表示相机中心; P_1 和 P_2 表示点 P 在相邻帧成像平面映射的特征点; l_1 和 l_2 表示极线, P_1 和 P_2 的齐次坐标形式为:

$$P_1 = [x_1, y_1, 1] \quad P_2 = [x_2, y_2, 1] \quad (5)$$

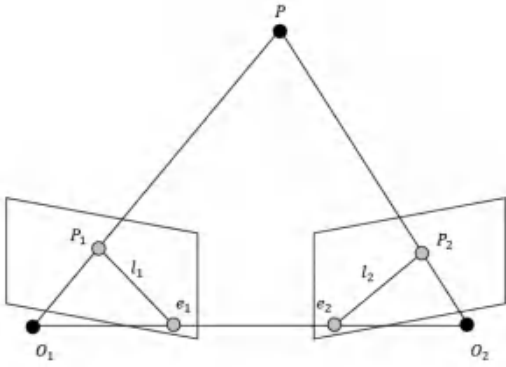


图4 对极几何约束

Fig. 4 Epipolar geometry constraints

极线向量 l_2 通过计算得到,用到的计算公式为:

$$l_2 = \begin{bmatrix} \hat{e}_2^T X \hat{u} \\ \hat{e}_2^T Y \hat{u} \\ \hat{e}_2^T Z \hat{u} \end{bmatrix} = FP_2 = F \begin{bmatrix} \hat{e}_2^T X \hat{u} \\ \hat{e}_2^T Y \hat{u} \\ \hat{e}_2^T Z \hat{u} \end{bmatrix} \quad (6)$$

其中, F 表示相机转换的基础矩阵。 P_2 到极线 l_2 的距离 d 为:

$$d = \frac{|P_2^T F P_1|}{\sqrt{X^2 + Y^2}} \quad (7)$$

设置阈值 ε , 通过判断偏移距离 (d) 和阈值 (ε):若 $d > \varepsilon$, 则表示该点为动态点,需要对特征点进行剔除;若 $d < \varepsilon$, 则表示该点为静态点,保留下来作为后续的特征匹配和地图重建的基础点。

2 实验结果与分析

在本节中,对改进的目标检测性能进行评估,并从2个方面对 DRV-SLAM 系统进行实验分析。首先,使用慕尼黑工业大学提供的 TUM RGB-D 数据集^[20],包含5个子集(sitting_xyz、walking_static、walking_xyz、walking_halfsphere、walking_rpy)评估ORB-SLAM3和DRV-SLAM绝对轨迹误差和相对

位姿误差,同时分析了本文算法相对于其他算法的性能优势。在此基础上,还对系统的实时性能进行了对比。

2.1 目标检测性能对比

本研究的训练参数设置如下:初始学习率为0.01,利用SGD优化。在训练过程中,使用0.937的动量来保持稳定的收敛性,迭代次数为300,权重衰减系数为 5×10^{-5} 。数据集为VOC2007,筛选包含人、动物和车等世界场景下具有潜在运动的目标图片,得到3500张训练集图片和1247张测试图片。

通过在YOLOv5s网络中使用C3-Faster模块替换参数量大的C3模块,提出了轻量化的目标检测算法YOLOv5s_FNet。YOLOv5s与改进YOLOv5s_FNet性能对比结果见表1。YOLOv5s_FNet相比于YOLOv5s加快了模型的处理速度,减少了参数数量。模型参数量减小了18.3%,精度下降0.4%,在满足轻量化的前提下损失了较小精度。

表1 YOLOv5s与改进YOLOv5s_FNet性能对比

Table 1 Performance comparison between YOLOv5s and improved YOLOv5s_FNet

网络模型	mAP@0.5	Params/M	GFLOPS
YOLOv5s	79.2	7.1	15.8
YOLOv5s_FNet	78.8	5.8	12.8

2.2 特征提取结果对比

本文的代码运行的操作系统使用发行版为64位的Ubuntu 20.04。将本文提出的基于信息熵的快速特征提取算法与基于二叉树的ORB特征点均匀提取算法进行实验。实验采用的图片为工厂场景下的TUM数据。

图5为在工厂环境下的特征点提取可视化。图5(a)为二叉树ORB算法提取结果,图5(b)为本文算法提取结果。



(a) 二叉树ORB算法提取结果

(b) 本文算法提取结果

图5 特征点提取结果

Fig. 5 Feature points extraction results

通过对比图 5(a) 和图 5(b) 发现, 本研究所提出的算法具备自适应调整特征点分配数量的能力, 这一调整基于图像中纹理的丰富程度进行。具体而言, 在纹理信息较为丰富的区域, 算法会提取较多的特征点, 以充分利用这些区域高度的信息密度; 而在纹理信息较少的区域, 则减少特征点的提取量。这种基于纹理丰富度的自适应特征点分配机制, 旨在优化特征提取的效率和准确性, 通过精细调整特征点的数量, 以适应图像不同区域的特性, 进而提升算法在视觉处理任务中的性能。

2.3 定位精度评估与分析

为了量化视觉 SLAM 算法在位姿估计上的精度, 本研究通过比较估计相机位姿与数据集提供的真实位姿之间的差异来进行评估。作为评价算法精度的主要指标, 采用了绝对轨迹误差 (Absolute Trajectory Error, ATE) 和相对位姿误差 (Relative Pose Error, RPE) 的平均值、均方根误差 (Root Mean

Square Error, RMSE) 和标准差。

ORB-SLAM3 与 DRV-SLAM 绝对轨迹误差对比见表 2, DRV-SLAM 与 ORB-SLAM3 相对轨迹误差对比见表 3。从表 2 和表 3 可以看出, 本文提出的方法在大多数高动态序列中绝对轨迹误差和相对轨迹误差几乎都减少了 90% 以上, 性能都达到量级的增强, 表明本文提出的异常点滤除机制可以有效降低动态特征点对位姿估计影响, 显著提高定位精度和鲁棒性。本文算法对于低动态序列 sitting_static 的轨迹精度提升不明显, 主要原因是在静态环境下, ORB-SLAM3 本身对静态场景下的位姿估计性能优越, 精度提升难度较大。而且在低动态环境或静态环境中可以使用目标对象上的点, 不会对跟踪线程性能造成影响, 所以在低动态序列下本文算法的改进并不明显。但基于纹理丰富度的自适应特征点分配机制, 优化了特征提取的效率和准确性。

表 2 ORB-SLAM3 与 DRV-SLAM 绝对轨迹误差对比

Table 2 Comparison of Absolute Trajectory Errors between ORB-SLAM3 and DRV-SLAM

序列	ORB-SLAM3			DRV-SLAM			Improvement		
	RMSE	Mean	S. D.	RMSE	Mean	S. D.	RMSE	Mean	S. D.
walking_xyz	0.678 7	0.393 4	0.169 9	0.014 1	0.011 8	0.008 9	0.979 2	0.970 0	0.947 6
walking_static	0.364 1	0.241 4	0.187 2	0.007 0	0.019 5	0.003 0	0.980 8	0.919 2	0.984 0
walking_rpy	0.786 9	0.690 3	0.286 1	0.026 8	0.026 9	0.021 5	0.965 9	0.961 0	0.924 9
walking_half	0.593 6	0.329 6	0.212 4	0.030 1	0.026 8	0.019 3	0.949 3	0.918 7	0.909 1
sitting_static	0.009 9	0.006 5	0.004 6	0.006 4	0.006 3	0.004 3	0.353 5	0.030 8	0.065 2

表 3 DRV-SLAM 与 ORB-SLAM3 相对轨迹误差对比

Table 3 Comparison of Relative Trajectory Errors between DRV-SLAM and ORB-SLAM3

序列	ORB-SLAM3			DRV-SLAM			Improvement		
	RMSE	Mean	S. D.	RMSE	Mean	S. D.	RMSE	Mean	S. D.
walking_xyz	0.555 8	0.347 5	0.336 3	0.008 8	0.008 5	0.008 0	0.984 2	0.975 5	0.976 2
walking_static	0.091 8	0.070 9	0.055 3	0.002 6	0.001 7	0.001 8	0.971 7	0.976 0	0.967 5
walking_rpy	0.336 8	0.318 5	0.269 7	0.041 9	0.030 9	0.025 2	0.875 6	0.903 0	0.906 6
walking_half	0.386 2	0.264 1	0.272 5	0.028 4	0.019 8	0.023 5	0.926 5	0.925 0	0.913 8
sitting_static	0.010 2	0.008 2	0.004 9	0.009 3	0.006 9	0.004 8	0.088 2	0.158 5	0.020 4

图 6 和图 7 分别展示了在 walking_xyz 和 walking_half 高动态序列下 ORB-SLAM3 与本文提出的 DRV-SLAM 的估计轨迹和真实轨迹的误差。ORB-SLAM3 容易受动态目标的影响, 从而使得估计轨迹与真实轨迹产生很大误差, 而本文算法实验得到的估计轨迹与真实轨迹基本重合, 表明动态异常点滤除使估计位姿精度得到了明显提高。

为了更全面检验本文所提出的算法, 将 DRV-SLAM 与 DynaSLAM 系统、DS-SLAM 系统和 YOLO-SLAM 系统进行对比。TUM RGB-D 数据集的绝对轨迹误差结果比较见表 4。基于实验结果可以得出结论, 本文的 SLAM 算法在低动态和高动态环境中都更具有稳定性。

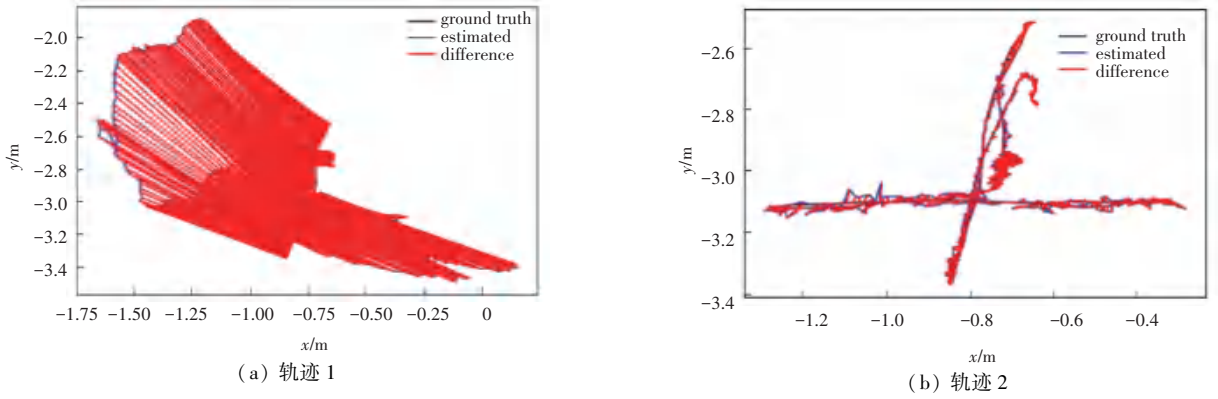


图 6 ORB-SLAM3 与 DRV-SLAM 在 w_xyz 轨迹

Fig. 6 ORB-SLAM3 and DRV-SLAM trajectories in w_xyz

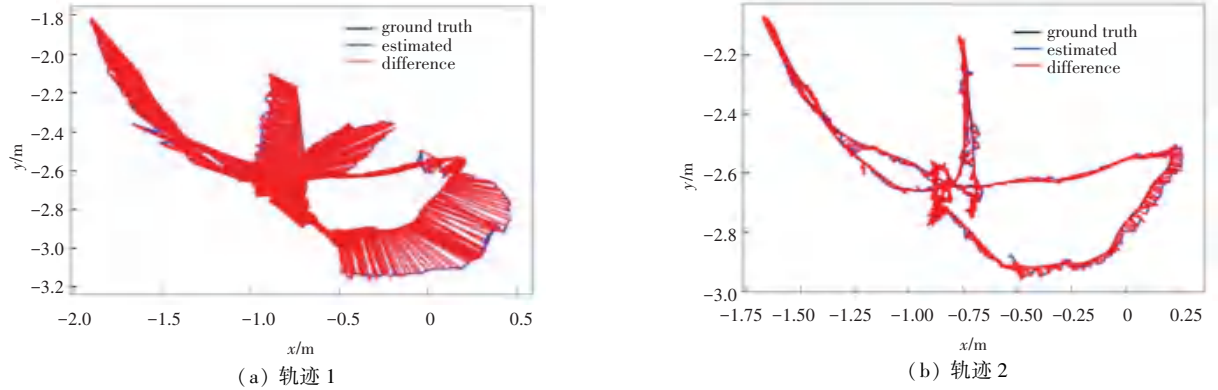


图 7 ORB-SLAM3 与 DRV-SLAM 在 w_half 轨迹

Fig. 7 ORB-SLAM3 and DRV-SLAM trajectories in w_half

表 4 TUM RGB-D 数据集的绝对轨迹误差结果比较

Table 4 Comparison of Absolute Trajectory Error results on the TUM RGB-D dataset

序列	DS-SLAM		DynaSLAM		YOLO-SLAM		DRV-SLAM	
	RMSE	S. D.	RMSE	S. D.	RMSE	S. D.	RMSE	S. D.
w_xyz	0.024 7	0.016 1	0.015 6	0.007 9	0.014 6	0.007 0	0.014 1	0.008 9
w_static	0.008 1	0.003 6	0.006 8	0.034 0	0.007 3	0.003 5	0.007 0	0.003 0
w_rpy	0.444 2	0.235 0	0.041 7	0.027 5	0.216 4	0.100 1	0.026 8	0.021 5
w_half	0.030 3	0.015 9	0.029 1	0.015 5	0.028 3	0.013 8	0.030 1	0.019 3
s_static	0.006 5	0.003 3	0.006 3	0.003 1	0.006 6	0.003 3	0.006 4	0.004 3

SLAM 系统的耗时影响定位和导航的实现。因此,测试了系统运行时处理每帧的时间,并将其与 ORB-SLAM3、DS-SLAM、DynaSLAM、YOLO-SLAM 对比,性能对比见表 5。由于 DynaSLAM 和 DS-SLAM 使用了语义分割算法进行动态特征点剔除,分割算法所耗费的时间更多。YOLO-SLAM 系统虽

然使用轻量化 YOLO 实现目标检测算法,但轻量化后的 YOLO 参数依旧较多,再加上硬件设备的限制,导致速度非常慢。本文改进 YOLOv5s,提出基于信息熵的特征提取算法、基于目标检测框占比和对极几何约束剔除动态特征点机制算法,耗时比 DS-SLAM 减少 47.7%,有较大的速度提升。

表 5 耗时结果对比

Table 5 Comparison of time consumption results

SLAM Systems	Tracking each time/ms	Hardware platform
ORB-SLAM3	23.8	Intel i7 CPU, NVIDIA Geforce GTX1650 GPU
DS-SLAM	57.1	Intel i7 CPU, P4000GPU
DynaSLAM	>200	Intel(R)Core(TM) i5-10400F CPU NVIDIA Geforce RTX2080Ti
YOLO-SLAM	696.09	Intel Core i5-4228U CPU
DRV-SLAM	39.2	Intel i7 CPU, NVIDIA Geforce GTX1650 GPU

3 结束语

针对动态场景中定位误差大问题,本文提出一种新的动态 SLAM 系统 DRV-SLAM。首先利用信息熵特征提取算法进行全局特征点提取,然后轻量化 YOLOv5s_FNet 完成语义信息的提取,并根据目标框所占比例进一步判断是否进行对极几何判断来识别所有动态特征点并滤除,尽可能地保留高质量静态特征点用于特征匹配以及位姿估计。通过使用 TUM 数据集进行实验,结果表明 DRV-SLAM 的定位精度方面有较大优势。但也有一些缺陷,稀疏地图不利于导航和避障任务,需要构建语义八叉树地图。

参考文献

- [1] DAVISON A J, REID I D, MOLTON N D, et al. MonoSLAM: Real-time single camera SLAM[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(6): 1052-1067.
- [2] ISLAM Q U, IBRAHIM H, CHIN P K, et al. ARD-SLAM: Accurate and robust dynamic SLAM using dynamic object identification and improved multi-view geometrical approaches [J]. Displays, 2024, 82: 102654.
- [3] 艾青林, 刘刚江, 徐巧宁. 动态环境下基于改进几何与运动约束的机器人 RGB-D SLAM 算法[J]. 机器人, 2021, 43(2): 167-176.
- [4] WANG Runzhi, WAN Wenhui, WANG Yongkang, et al. A new RGB-D SLAM method with moving object detection for dynamic indoor scenes [J]. Remote Sensing, 2019, 11(10): 1143.
- [5] CAMPOS C, ELVIRA R, RODRÍGUEZ J. 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 Yubao, MIURA Jun. RDS-SLAM: Real-time dynamic SLAM using semantic segmentation methods [J]. IEEE Access, 2021, 9: 23772-23785.
- [7] LIU Pengfei, ZHANG Huan, YANG Gaochao, et al. An improved image segmentation method of BiSenetV2 Network [C]// Proceedings of the 4th International Conference on Control and Computer Vision. New York: ACM, 2021: 13-17.
- [8] LONG Xudong, ZHANG Weiwei, ZHAO Bo. PSPNet-SLAM: A semantic SLAM detect dynamic object by pyramid scene parsing network [J]. IEEE Access, 2020, 8: 214685-214695.
- [9] ZHAO Hengshuang, SHI Jianping, QI Xiaojuan, et al. Pyramid scene parsing network [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 2881-2890.
- [10] BESCOS B, FÁCIL J, CIVERA J. DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes [J]. IEEE Robotics and Automation Letters, 2018, 3(4): 4076-4083.
- [11] HE Kaiming, GKIOXARI G, DOLLÁR P, et al. Mask RCNN [C] // Proceedings of the IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE, 2017: 2961-2969.
- [12] WU Wenxin, GUO Liang, GAO Hongli, et al. YOLO-SLAM: A semantic SLAM system towards dynamic environment with geometric constraint [J]. Neural Computing and Applications, 2022, 34: 6011-6026.
- [13] YAN Zhe, CHU Shuchun, DENG Liwei. Visual SLAM based on instance segmentation in dynamic scenes [J]. Measurement Science and Technology, 2021, 32(9): 095113.
- [14] LUO Kaiqing, LIN Manling, WANG Pengcheng, et al. Improved ORB-Slam2 algorithm based on information entropy and image sharpening adjustment [J]. Mathematical Problems in Engineering, 2020, 2020: 4724310.
- [15] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 779-788.
- [16] WANG Dandan, HE Dongjian. Channel pruned YOLO V5s-based deep learning approach for rapid and accurate apple fruitlet detection before fruit thinning [J]. Biosystems Engineering, 2021, 210: 271-281.
- [17] SHAN Chun, LIU Hongming, YU Yu. Research on improved algorithm for helmet detection based on YOLOv5 [J]. Scientific Reports, 2023, 13(1): 18056.
- [18] CHEN Jierun, KAO S H, HE Hao, et al. Run, don't walk: Chasing higher FLOPS for faster neural networks [C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2023: 12021-12031.
- [19] 李庚, 吴丽君. 基于实例分割的动态场景视觉 SLAM [J]. 微电子学与计算机, 2023, 40(10): 29-37.
- [20] LI Shile, LEE D. RGB-D SLAM in dynamic environments using static point weighting [J]. IEEE Robotics and Automation Letters, 2017, 2(4): 2263-2270.