

室内动态场景下基于目标检测算法的 语义视觉 SLAM

阮晓钢, 周 晨, 黄 静
(北京工业大学信息学部, 北京 100124)

摘 要: 为了解决基于静态环境假设的同步定位与地图构建(simultaneous localization and mapping, SLAM)在处理动态物体时系统定位精度降低、鲁棒性变差的问题,提出一种面向室内动态环境的语义视觉 SLAM 系统。该系统以 ORB_SLAM2 为基础,添加了动态目标检测线程。对输入图像提取 ORB 特征的同时,通过使用 YOLOv5s 网络进行动态目标检测,并结合光流法和极几何约束方法,共同筛选动态特征点,最后,使用静态特征点进行位姿估计。通过使用慕尼黑工业大学(Technical University of Munich, TUM)数据集将该系统和 ORB_SLAM2 进行比较,结果表明,该系统显著降低了轨迹误差。与 DS-SLAM、DynaSLAM 等动态环境下的系统相比,该系统可有效平衡语义视觉 SLAM 系统位姿估计的准确性、鲁棒性和快速性。

关键词: 同步定位与地图构建; 室内动态环境; 目标检测; 动态特征点; 位姿估计; YOLOv5s

中图分类号: TP 242.6

文献标志码: A

文章编号: 0254-0037(2023)08-0842-09

doi: 10.11936/bjtxb2021090025

Semantic Visual SLAM Based on Target Detection Algorithm in Indoor Dynamic Scenes

RUAN Xiaogang, ZHOU Chen, HUANG Jing

(Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China)

Abstract: To solve the problem of decreased positioning accuracy and poor robustness of simultaneous localization and mapping (SLAM) systems based on static environment assumptions when processing dynamic objects, a semantic vision SLAM system for indoor dynamic environments was proposed, which was based on ORB_SLAM2, and added with a dynamic target detection thread. While extracting ORB features from the input image, YOLOv5s network was used for dynamic target detection, combining optical flow method and epipolar geometric constraints to screen dynamic feature points, and finally static feature points were used for pose estimation. Using the data set of Technical University of Munich (TUM) to compare with ORB_SLAM2, results show that the system significantly reduces the trajectory error. Compared with the system in the dynamic environment such as DS-SLAM and DynaSLAM, this system can effectively balance the accuracy, robustness and rapidity of the pose estimation of the semantic visual SLAM system.

Key words: simultaneous localization and mapping (SLAM); indoor dynamic environment; target detection; dynamic feature points; pose estimation; YOLOv5s

收稿日期: 2021-09-27; 修回日期: 2022-02-19

基金项目: 国家自然科学基金资助项目(61773027, 62076014); 国家重点研发计划资助项目(2020YFB1005900)

作者简介: 阮晓钢(1958—),男,教授,主要从事机器人、自动控制与人工智能方面的研究, E-mail: adrxg@bjut.edu.cn

通信作者: 黄 静(1979—),女,副教授,主要从事认知机器人方面的研究, E-mail: huangjing@bjut.edu.cn

同步定位与地图构建(simultaneous localization and mapping, SLAM)是指主体搭载特定传感器,在无环境先验信息的前提下,在运动过程中建立环境模型并估计自身的运动状态^[1]。到目前为止,SLAM研究已历经3个阶段的发展,当前阶段的SLAM研究更加注重高层次的理解能力和鲁棒性^[2]。SLAM可以有效解决导航中的定位和建图问题,目前,在自动驾驶汽车^[3]、室内机器人、增强现实等方面有很多应用。

视觉SLAM系统应用了各类精确的视觉传感器,在相同条件下可以获得更多有效信息^[4]。机器视觉理论的诞生明显促进了视觉SLAM的发展,使视觉SLAM成为研究的主流方向之一。部分先进的视觉SLAM算法已被广泛应用。MonoSLAM^[1]是最早的实时高频单目视觉SLAM,是视觉SLAM研究发展中的重要里程碑。并行跟踪与建图(parallel tracking and mapping, PTAM)^[5]首次区分出前端、后端,可在不同线程下完成建图和跟踪,同时,提供了关键帧提取技术。基于直接法的LSD-SLAM^[6]可从姿态跟踪中恢复半稠密的3D地图。基于特征点法的ORB_SLAM2^[7]成为首个同时支持单目、双目和RGB-D这3种传感器形式的SLAM系统,该系统可实时计算出相机轨迹,并生成场景三维重建的结果。

尽管许多视觉SLAM算法的性能十分优越,但大部分算法都基于静态环境假设^[8]。动态环境是视觉SLAM算法应用在现实世界的一个重大挑战。场景中存在动态物体会造成错误的特征匹配,影响系统的定位和建图精度。在处理动态物体时,大多数视觉SLAM系统是对前端进行修改,具体方法是依据特征点的几何信息进行动静状态检测。Dai等^[9]通过地图点之间的相关性区分特征点的动静状态;Li等^[10]通过对关键帧边缘点使用静态加权的方法减少动态物体的影响;Zou等^[11]通过将地图特征投影到当前帧计算重投影误差,通过计算结果判断特征点是否为动态。以上方法均采用几何计算来判断特征点是否为动态特征点,因为没有考虑到使用图像帧中各物体的语义信息,所以在处理一些位于物体边缘等特殊位置的特征点时,容易出现状态判断错误,导致错误匹配数量增多,使系统在动态环境中的定位精度降低。

近年来,深度学习取得了重大发展,部分研究人员将深度学习和动态环境下的SLAM问题结合起来^[12],依靠语义信息处理动态物体成为新的选择方式。Xiao等^[13]通过使用单次多框检测器(single

shot multibox detector, SSD)有效去除了动态物体区域内的特征点;Zhong等^[14]提出Detect-SLAM,通过使用更加精准的目标检测器去除了动态物体中的不可靠特征点;Bescos等^[15]提出了DynaSLAM,将语义分割和多视角几何结合,利用MASK-R-CNN进行动态目标实例分割,对动态特征点进行精准去除,并修复背景;DS-SLAM将SegNet语义分割网络与运动一致性检测方法进行结合,减少了动态目标的影响,动态环境下的定位精度得到了显著提高^[16]。文献[17]中提到Detect-SLAM、DynaSLAM、DS-SLAM是近年来解决动态SLAM问题的有效方法,但同时也指出:Detect-SLAM使用的语义分割网络无法有效区分同类动态物体;DynaSLAM耗时严重,实时性差;DS-SLAM通过语义分割网络和几何检测相结合的方式可以有效平衡检测的速度和精度,但动态物体只能限定为人。

针对动态环境进行的SLAM研究虽然日趋完善,但依然存在各种问题。文献[18]指出RGB-D SLAM^[19]是目前通过结合深度学习解决动态SLAM的优秀系统,该系统语义分割精度高,但运动检测的鲁棒性和时间效率性能欠佳。与RGB-D SLAM系统类似,SaD-SLAM^[20]也是目前通过结合深度学习对动态物体进行精准分割的动态SLAM系统,该系统借助语义信息和深度信息,跟踪性能稳健,但该系统的速度同样不满足实时性要求。除此之外,文献[18]提出的RS-SLAM系统是目前本研究领域十分成功的鲁棒语义视觉SLAM系统,通过使用金字塔场景解析网络(pyramid scene parsing network, PSPNet)识别场景中的动态物体,显著提高了定位精度、鲁棒性和时间效率,但该系统将人限定为绝对动态物体,造成部分可用信息丢失。

本文提出的算法是将YOLOv5s目标检测网络和几何筛选进行有效结合,从而高效去除动态特征点。实验结果证明,本文算法在提高位姿估计准确性上效果明显且系统的运行速度较快。

1 系统框架

ORB_SLAM2延续PTAM的算法框架,使用ORB特征,拥有比PTAM更好的视角不变性,并且ORB_SLAM2具有更加鲁棒的关键帧和三维点选择机制,整体系统具备跟踪、局部建图、回环检测3个线程,系统在回环检测后进行全局束调整。ORB_SLAM2系统结构完善,性能较好,因此,目前众多研究动态场景的系统都是在ORB_SLAM2的

基础上进行改善的,本文提出的语义视觉 SLAM 也是以该系统为基础,具体的系统框架如图 1 所示。改进后的系统在原跟踪线程上并行添加目标检测线程。原始的图像帧在采集后同时进行目标检测和提取 ORB 特征点,目标检测线程在进行物体识别后划分出动静区域,并将划分结果传送至跟踪线程。跟踪线程根据划分结果和几何约束方法综合判断特征点的动静状态。YOLOv5s 和几何约束方法共同将提取的 ORB 特征点中的动态特征点进行精确剔除,使用剩余的静态特征点进行后续位

姿估计。文献[21]证明单独使用几何约束的方法在判断位于物体边缘等位置的特征点时容易出现错误。文献[16]通过使用语义信息和几何信息来联合估计位姿的方法显著提高了定位的准确性,故本文使用了目标检测算法和几何约束相结合的方式。YOLOv5s 目标检测算法可有效规避几何约束对物体边缘特征点判断能力弱的缺点,同时,几何约束也可以对目标检测无法检测到的潜在动态物体上所包含的动态特征点予以剔除,二者相互补充。

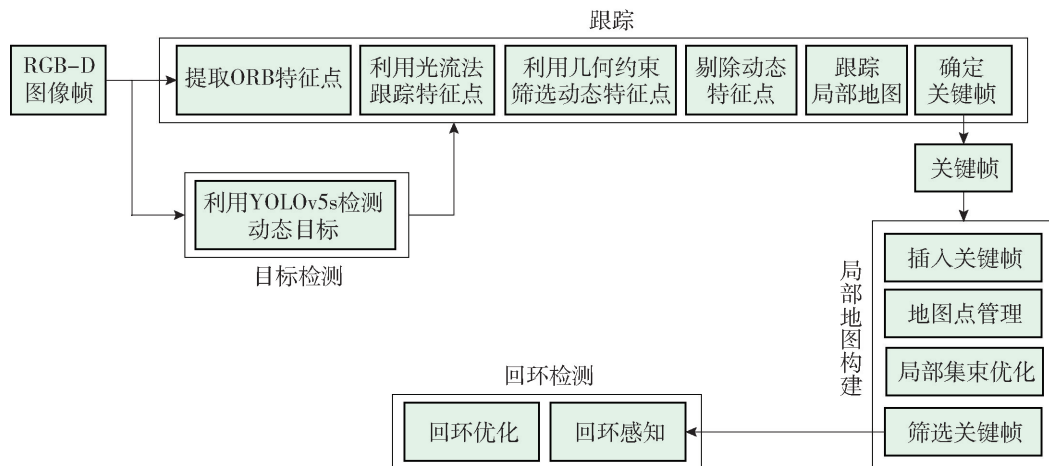


图1 系统算法框架

Fig.1 Framework of system algorithm

1.1 基于 YOLOv5s 的目标检测

YOLO 是基于神经网络的单阶段目标检测算法,可从输入的整张图片中直接得到多个目标的边界框及对应目标的分类概率。该算法是目前较为先进的实时目标检测算法^[22]。

YOLOv5 是目前 YOLO 系列最新的成果, YOLOv5 在 YOLOv3 的基础上进行了网络结构方面的改善,增强了对小目标的检测能力,可自适应计算出不同训练集的最佳锚框值。YOLOv5 共有 4 种模型,分别为 YOLOv5s、YOLOv5l、YOLOv5m 和 YOLOv5x,其中 YOLOv5s 的网络深度和特征图宽度均为最小,后 3 种模型在 YOLOv5s 的基础上进行加深和加宽,但处理速度越来越慢。通过文献[23]可得到 4 种模型的速度和精度关系,将以上 4 种模型和分类网络 EfficientDet 进行对比,其结果如图 2 所示。综合考虑各项数据,YOLOv5s 对本文要研究的室内环境十分适用。

本文算法在训练中使用的是 MS COCO^[24] 数据集训练权重,数据集包括 80 种物体,如人、自行车、显示器、笔记本电脑、冰箱、键盘、椅子、杯子、猫、狗

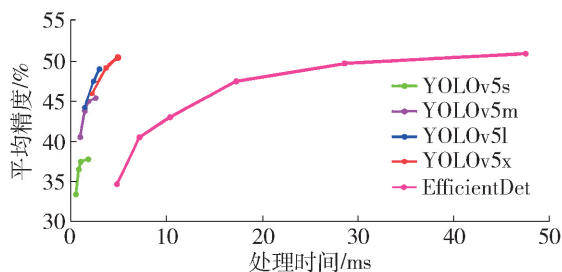


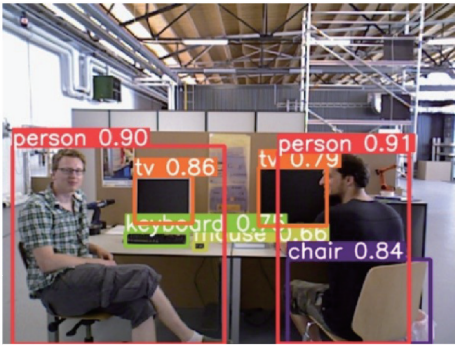
图2 YOLOv5 的 4 种模型的速度和精度

Fig.2 Speed and accuracy of four models of YOLOv5

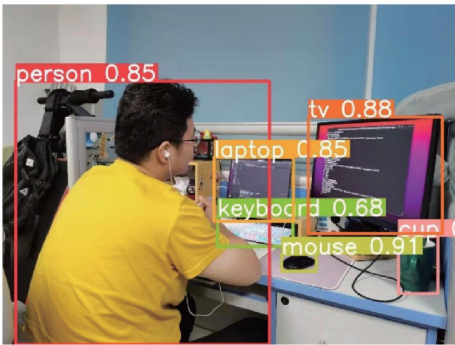
等,几乎包含了常见室内环境下的所有动态物体。YOLOv5s 网络在 Backbone 和 Neck 部分分别使用了不同的跨阶段网络结构,其特征融合能力很强。经过实际训练发现,YOLOv5s 的训练时间远远短于 YOLOv4。YOLOv5s 会对原始图像进行预处理,使之成为标准大小,然后,将图像划分为若干网格,每个网格分别预测多个边界框,每个边界框的预测值含 5 个元素: x 、 y 、 w 、 h 、 c 。其中: (x, y) 表示边界框中心位置坐标; (w, h) 表示预测边界框和整张图片在高度、宽度上的比例; c 表示置信度,置信度大小由边

界框包含的目标概率和边界框预测的准确度共同决定,而最终识别的物体类别概率就是在该置信度下产生的条件概率。为避免划分的多网格响应同一目标,YOLOv5s 通过使用非极大值抑制(non-maximum suppression, NMS)获得置信度最高的边界框。

常见场景中除 MS COCO 数据集列举的动态物体外,还存在一类潜在的动态物体,虽然这类物体无法依靠自身移动,但可以跟随人一起移动,如人拿着杯子,人移动会带动杯子运动,显然单独使用 YOLOv5s 检测到杯子是静态物体是不准确的,语义静态区域并非真实的静态区域,还需要与几何约束的判断相结合,共同判断动静区域。图3展示了对目标检测的实际效果。图3(a)为 YOLOv5s 对慕尼黑工业大学(Technical University of Munich, TUM)数据集的一张图像帧进行目标检测的效果;图3(b)为 YOLOv5s 对实际室内场景进行目标检测的效果。



(a) 对TUM数据集进行目标检测



(b) 对现实场景进行目标检测

图3 YOLOv5s 的检测效果

Fig. 3 Detection effect of YOLOv5s

1.2 几何约束筛选

在几何判断中,首先使用光流法进行特征点跟踪,光流不需要计算描述子和匹配特征即可实现跟踪,实时性好。Lucas-Kanada 光流算法^[16,25]基于相邻帧像素灰度不变假设,将图像视为时间的函数 $f(t)$, t 时刻坐标 (x, y) 的特征点灰度值为 $f(x, y, t)$, $t + dt$ 时刻的后续帧中特征点的灰度值 $f(x + dx, y +$

$dy, t + dt)$ 基于灰度不变假设应符合

$$f(x, y, t) = f(x + dx, y + dy, t + dt) \quad (1)$$

泰勒展开后满足

$$f(x, y, t) = f(x, y, t) + \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy + \frac{\partial f}{\partial t} dt \quad (2)$$

计算式(2)可得

$$\frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy + \frac{\partial f}{\partial t} dt = 0 \quad (3)$$

$$\frac{\partial f}{\partial x} \frac{dx}{dt} + \frac{\partial f}{\partial y} \frac{dy}{dt} = - \frac{\partial f}{\partial t} \quad (4)$$

式中: $\frac{\partial f}{\partial x}$ 和 $\frac{\partial f}{\partial y}$ 分别为特征点灰度值沿 x 轴和 y 轴的梯度变化; $\frac{dx}{dt}$ 和 $\frac{dy}{dt}$ 分别为特征点沿 x 轴和 y 轴的速度。令 $f_x = \frac{\partial f}{\partial x}$, $f_y = \frac{\partial f}{\partial y}$, $f_t = \frac{\partial f}{\partial t}$, $u = \frac{dx}{dt}$, $v = \frac{dy}{dt}$, 式(3)

(4)可合并为

$$\begin{bmatrix} f_x & f_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -f_t \quad (5)$$

Lucas-Kanada 光流算法假设同一子图像内像素点运动相同,基于这样的假设可计算得到 u 和 v ,也即得到像素点在后续帧中的位置,最终实现特征点跟踪。

在使用光流法获得相邻帧中的匹配特征点后,可利用匹配特征点计算基础矩阵。假定 p_1 和 p_2 为匹配的特征点,坐标依次表示为

$$\begin{cases} p_1 = [x_1, y_1, 1] \\ p_2 = [x_2, y_2, 1] \end{cases} \quad (6)$$

利用基础矩阵和特征点坐标可得到极线

$$I_1 = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = F p_1 \quad (7)$$

式中 F 为基础矩阵。 p_2 到极线 I_1 的距离 d 可表示为

$$d = \frac{|p_2^T F p_1|}{\sqrt{\|X\|^2 + \|Y\|^2}} \quad (8)$$

当距离 d 大于预定阈值时,可判断 p_2 为动态特征点。

2 实验部分

本文选用 TUM 数据集的多组序列进行 3 组实验。其中:第 1 组实验展示通过使用 YOLOv5s 识别动态目标进而剔除目标区域内动态特征点的效果;第 2 组实验用于检验本文算法在轨迹误差方面的精度;第 3 组实验用于检验本文算法的运行速度。

TUM-RGBD^[26] 开源数据集是评估 RGB-D SLAM 系统基准的常用数据集,该数据集包含 39 个序列,每个序列包含彩色图、深度图和相关的真实相机位姿。图像的采集分辨率为 640×480 像素,记录频率为 30 Hz。本文针对室内动态环境进行研究,故选用 walking 序列作为实验数据序列,walking 序列主要记录 2 个人围绕办公桌的走动。其中:walking_xyz、walking_static、walking_half 均为高动态序列,不同之处在于拍摄时的传感器运动方式不同;sitting_static 为低动态序列,主要记录 2 个人坐在桌前,仅手部有动作变化。文献[16,19,27-28]针对动态场景进行的视觉 SLAM 研究均以 ORB_SLAM2 为基础框架,对 ORB_SLAM2 算法的内部参数均未提及,鉴于 ORB_SLAM2 算法已被多次使用,是十分成熟的算法,内部参数已经实现充分优化。为了保证各组实验对比的公平性,本文有关 ORB_SLAM2 部分的参数均为其原始参数。

衡量 SLAM 系统定位精度的常用指标为相对位姿误差 (relative pose error, RPE) 和绝对轨迹误差 (absolute trajectory error, ATE)。RPE 可测量固定时间间隔内估计运动和真实运动之间的姿态变化量差值,ATE 通过比较估计运动和真实运动的绝对距离可评估全局一致性。RPE 需要同时考虑平移误差和旋转误差,ATE 只考虑平移误差。定义时间步长为 i ,时间间隔为 Δ ,则相对位姿误差为

$$E_i = (\mathbf{Q}_i^{-1} \mathbf{Q}_{i+\Delta})^{-1} (\mathbf{P}_i^{-1} \mathbf{P}_{i+\Delta}) \quad (9)$$

式中: $\mathbf{Q}_i$ 为第 i 帧的真实位姿; $\mathbf{P}_i$ 为第 i 帧的估计位姿。绝对位姿误差为

$$\mathbf{F}_i := \mathbf{Q}_i^{-1} \mathbf{S} \mathbf{P}_i \quad (10)$$

式中 $\mathbf{S}$ 为估计位姿到真实位姿的转换矩阵。

2.1 效果对比实验

选用 TUM 数据集中 walking_xyz 序列的一组图进行对比,图 4(a) 为传统 ORB_SLAM2 算法在室内动态场景下采集的特征点,图 4(b) 为本文算法采集的特征点。可以明显看出,本文算法可有效识别 2 个人为动态物体,并去除人所包含的动态特征点,而其余的静态特征点则被保留。

2.2 精度对比实验

本文实验使用 walking_xyz、walking_static、walking_half、sitting_static 共 3 个高动态序列和 1 个低动态序列进行精度对比,采用 ATE 和 RPE 进行定量估计,给出均方根误差 (root mean square error, RMSE)、标准差 (standard deviation, Std) 共 2 个具体指标。精度提高程度的计算公式为



(a) ORB_SLAM2



(b) 本文算法

图 4 动态特征点剔除效果对比

Fig. 4 Comparison of the effect of dynamic feature point removal

$$\eta = \frac{\alpha - \beta}{\alpha} \times 100\% \quad (11)$$

式中: α 为 ORB_SLAM2 的运行结果; β 为本文算法的运行结果。

图 5~7 依次显示了 ORB_SLAM2 和本文算法在使用 3 个高动态序列时 ATE 和 RPE 的对比。可以看出,本文算法的 ATE 和 RPE 在不同序列中均显著小于 ORB_SLAM2,但图 7(b) 显示当相机进行翻滚等快速运动时,本文算法容易出现跟踪丢失,这是因为目标检测网络所划分的动态区域过大,导致区域外的静态特征点数量太少,无法实现有效跟踪。

表 1~3 依次显示了 ORB_SLAM2 和本文算法在 4 个不同序列下的数据对比及对应数据的提高程度,表中各个数据是使用同一计算机得到的结果。可以看出:在高动态序列中,本文算法相较于 ORB_SLAM2 在各项数据中都有了显著提高;在低动态序列中,各项数据的提高程度并不明显,这是因为此时的动态物体较少,ORB_SLAM2 可以依靠自身的噪声剔除算法很好地处理低动态场景,本文算法在低动态场景中所能提升的性能并不明显。本文算法和 DS-SLAM、DynaSLAM 的 RMSE 对比结果见表 4,DS-SLAM、DynaSLAM 的相关数据分别见文献[16]和

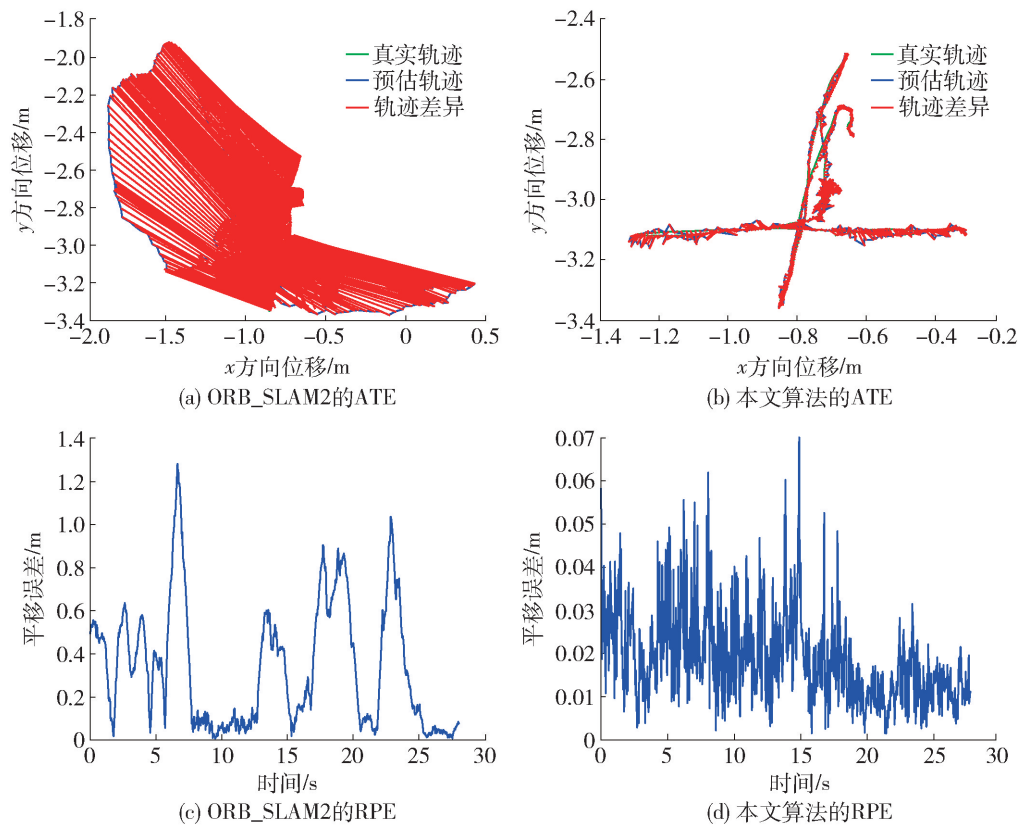


图 5 walking_xyz 的 ATE 和 RPE

Fig. 5 Absolute trajectory error and relative pose error of walking_xyz

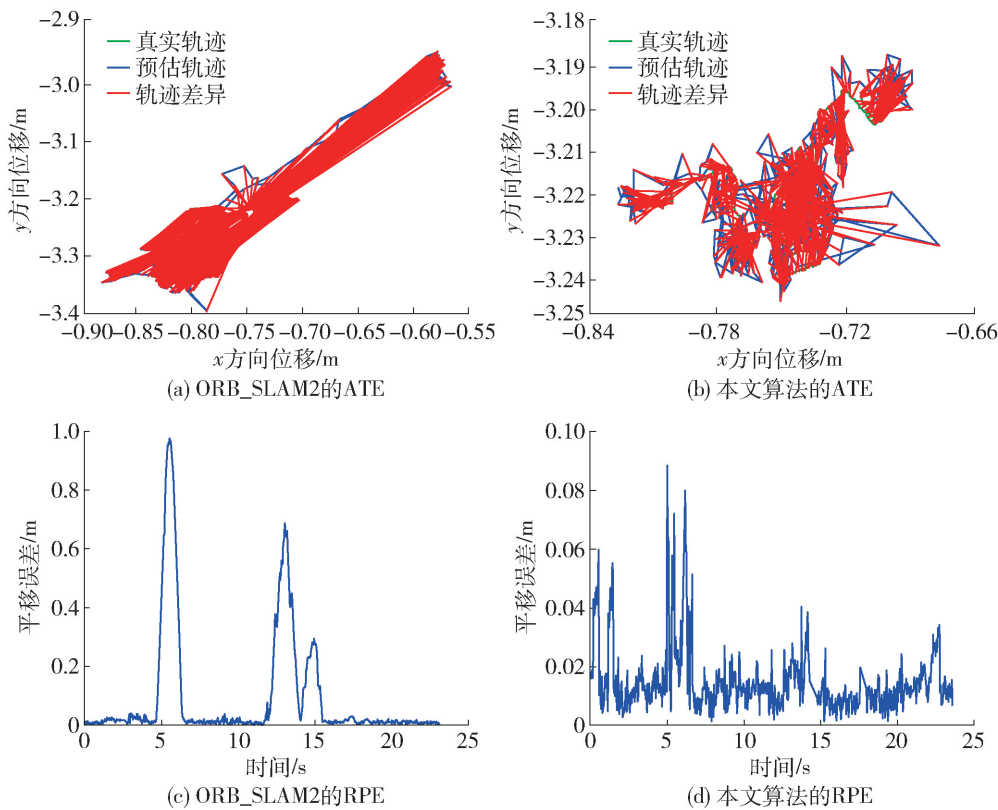


图 6 walking_static 的 ATE 和 RPE

Fig. 6 Absolute trajectory error and relative pose error of walking_static

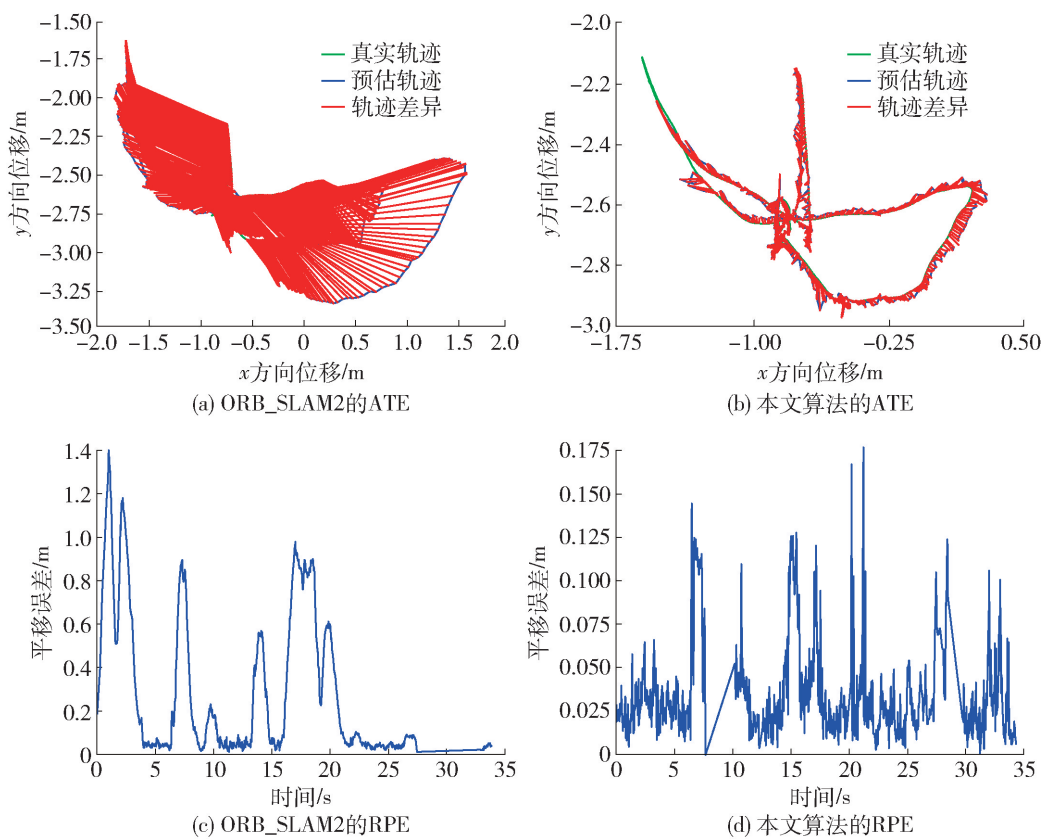


图7 walking_half 的 ATE 和 RPE

Fig.7 Absolute trajectory error and relative pose error of walking_half

表 1 ATE 对比

Table 1 Comparison of absolute trajectory errors

序列	RMSE/m		Std/m		精度提高程度/%	
	ORB_SLAM2	本文算法	ORB_SLAM2	本文算法	RMSE	Std
walking_xyz	0.721 1	0.019 1	0.404 6	0.010 1	97.35	97.50
walking_static	0.394 4	0.010 5	0.170 2	0.007 6	97.34	95.53
walking_half	0.742 6	0.037 8	0.340 1	0.023 1	94.91	93.21
sitting_static	0.008 3	0.007 6	0.003 8	0.003 6	8.43	5.26

表 2 RPE 中平移误差对比

Table 2 Comparison of translation error in relative pose error

序列	RMSE/m		Std/m		精度提高程度/%	
	ORB_SLAM2	本文算法	ORB_SLAM2	本文算法	RMSE	Std
walking_xyz	1.085 2	0.027 5	0.635 5	0.013 4	97.47	97.89
walking_static	0.561 7	0.030 7	0.398 3	0.023 8	94.53	94.02
walking_half	1.062 4	0.057 6	0.640 1	0.025 2	94.58	96.06
sitting_static	0.011 9	0.010 9	0.005 7	0.005 0	8.40	12.28

文献[15]。由表可知,本文算法在各序列中可实现与 DS-SLAM、DynaSLAM 相近的精度。在 walking_xyz 序列中,本文算法的 RMSE 小于 DS-SLAM,在 walking_static 和 walking_half 序列中 RMSE 大于

表 3 RPE 中旋转误差对比

Table 3 Comparison of rotation error in relative pose error

序列	RMSE/(°)		Std/(°)		精度提高程度/%	
	ORB_SLAM2	本文算法	ORB_SLAM2	本文算法	RMSE	Std
walking_xyz	17.535 8	0.703 3	10.036 7	0.420 9	95.99	95.81
walking_static	10.276 9	0.455 0	7.287 9	0.266 7	95.57	96.34
walking_half	25.368 1	0.924 4	14.640 1	0.576 8	96.36	96.06
sitting_static	0.351 0	0.330 3	0.154 9	0.142 9	5.90	7.75

DS-SLAM。与 DynaSLAM 对比后发现,本文算法的 RMSE 稍大于 DynaSLAM,这是因为 DynaSLAM 降低了系统实时性而提高了检测精度。本文算法在精度上虽然稍差,但速度快,改进了 DynaSLAM 系统在速度和精度上无法平衡的缺点。

2.3 速度对比实验

在处理动态环境的语义视觉 SLAM 研究中,速度是一个重要指标,表 5 对比了本文算法和 ORB_SLAM2、DS-SLAM 及 DynaSLAM 等在同一计算机上的运行时间。因为在 ORB_SLAM2 的系统框架上增加了目标检测线程和几何筛选部分,所以本文算法的跟踪时间长于 ORB_SLAM2,但与 DS-SLAM、

表 4 动态环境下的 SLAM 算法 RMSE 对比

Table 4 RMSE comparison of SLAM algorithms in a dynamic environment

序列	DS-SLAM ^[16]	DynaSLAM ^[15]	本文算法
walking_xyz	0.024 7	0.015 0	0.019 1
walking_static	0.008 1	0.006 0	0.010 5
walking_half	0.030 3	0.025 0	0.037 8
sitting_static	0.006 6		0.007 6

DynaSLAM 相比,本文算法在速度方面的性能有显著提升。综合考虑各项数据,本文算法在精度和速度上实现了很好的平衡。

表 5 动态环境下 SLAM 算法的图像处理时间

Table 5 Image processing time of SLAM algorithms in a dynamic environment

序列	ORB_SLAM2	DS-SLAM	DynaSLAM	本文算法
walking_xyz	0.039 6	0.305 3	1.224 8	0.077 2
walking_static	0.038 1	0.292 5	1.173 8	0.078 2
walking_half	0.041 9	0.275 6	1.283 6	0.080 3
sitting_static	0.032 4	0.326 7	1.251 6	0.078 4

3 结论

1) 本文提出一种面向室内动态环境的语义视觉 SLAM 系统,该系统在 ORB_SLAM2 的基础上添加了 YOLOv5s 目标检测线程,经过检测可以获得图像帧中动态物体所在区域,配合几何约束判断可准确筛选动态特征点。

2) 经过和其他 SLAM 算法在多方面的对比,结果显示本文算法显著提高了位姿估计的精度,并保证了运行速度,实现了速度和精度的平衡。

3) 本文算法在处理动态区域过大的图像帧时,容易造成特征点数量不够。当图像帧中动态物体移动过快时,容易造成目标检测线程和其他线程在图像帧处理上出现异步。如何解决以上问题,并且还能平衡未来动态场景下语义视觉 SLAM 算法的速度

和精度,成为下一步需要解决的问题。

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] CADENA C, CARLONE L, CARRILLO H, et al. Past, present, and future of simultaneous localization and mapping: towards the robust-perception age [J]. IEEE Transactions on Robotics, 2016, 32(6): 1309-1332.

[3] BRESSON G, ALSAYED Z, YU L, et al. Simultaneous localization and mapping: a survey of current trends in autonomous driving[J]. IEEE Transactions on Intelligent Vehicles, 2017, 2(3): 194-220.

[4] PARK C, SONG J B. Illumination change compensation and extraction of corner feature orientation for upward-

- looking camera-based SLAM[C] // International Conference on Ubiquitous Robots and Ambient Intelligence. Piscataway: IEEE, 2015: 224-227.
- [5] KLEIN G, MURRAY D. Parallel tracking and mapping for small AR workspaces[C] // IEEE and ACM International Symposium on Mixed and Augmented Reality. Piscataway: IEEE, 2007: 225-234.
- [6] ENGEL J, SCHOPS T, CREMERS D. LSD-SLAM: large-scale direct monocular SLAM[C] // European Conference on Computer Vision. Berlin: Springer, 2014: 834-849.
- [7] MUR-ARTAL R, TARDOS J D. ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras[J]. IEEE Transactions on Robotics, 2017, 33(5): 1255-1262.
- [8] WALCOT-TBRYANT A, KAESSE M, JOHANSSON H, et al. Dynamic pose graph SLAM: long-term mapping in low dynamic environments[C] // IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway: IEEE, 2012: 1871-1878.
- [9] DAI W C, ZHANG Y, LI P, et al. RGB-D SLAM in dynamic environments using point correlations[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(1): 373-389.
- [10] LI S, LEE D. RGB-D SLAM in dynamic environments using static point weighting[J]. IEEE Robotics and Automation Letters, 2017, 2(4): 2263-2270.
- [11] ZOU D, TAN P. CoSLAM: collaborative visual SLAM in dynamic environments[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(2): 354-366.
- [12] SAPUTRA M R U, MARKHAM A, TRIGONI N. Visual SLAM and structure from motion in dynamic environments: a survey[J]. ACM Computing Surveys, 2018, 51(2): 37.
- [13] XIAO L H, WANG J G, QIU X S, et al. Dynamic-SLAM: semantic monocular visual localization and mapping based on deep learning in dynamic environment[J]. Robotics and Autonomous Systems, 2019, 117: 1-16.
- [14] ZHONG F W, WANG S, ZHANG Z Q, et al. Detect-SLAM: making object detection and SLAM mutually beneficial[C] // IEEE Winter Conference on Applications of Computer Vision. Piscataway: IEEE, 2018: 1001-1010.
- [15] BESCOS B, FÁCIL J M, CIVERA J, et al. DynaSLAM: tracking, mapping, and inpainting in dynamic scenes[J]. IEEE Robotics and Automation Letters, 2018, 3(4): 4076-4083.
- [16] YU C, LIU Z X, LIU X J, et al. DS-SLAM: a semantic visual SLAM towards dynamic environments[C] // IEEE International Conference on Intelligent Robots and Systems. Piscataway: IEEE, 2018: 1168-1174.
- [17] 高兴波, 史旭华, 葛群峰, 等. 面向动态物体场景的视觉SLAM综述[J]. 机器人, 2021, 43(6): 733-750. GAO X B, SHI X H, GE Q F, et al. Overview of visual SLAM for dynamic object scene[J]. Robot, 2021, 43(6): 733-750.
- [18] RAN T, YUAN L, ZHANG J B, et al. RS-SLAM: a robust semantic SLAM in dynamic environments based on RGB-D sensor[J]. IEEE Sensors Journal, 2021, 21(18): 20657-20664.
- [19] XIE W F, LIU P X, ZHENG M H. Moving object segmentation and detection for robust RGBD-SLAM in dynamic environments[J]. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 5001008.
- [20] YUAN X, CHEN S. SaD-SLAM: a visual SLAM based on semantic and depth information[C] // IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway: IEEE, 2020: 4930-4935.
- [21] BAO S Y, BAGRA M, CHAO Y W, et al. Semantic structure from motion with points, regions, and objects[C] // IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 2703-2710.
- [22] 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. Piscataway: IEEE, 2016: 779-788.
- [23] JIA W, XU S Q, LIANG Z, et al. Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector[J]. IET Image Processing, 2021, 15(14): 3623-3637.
- [24] LIN T Y, MAIRE M, BELONGIE S, et al. Microsoft COCO: common objects in context[C] // European Conference on Computer Vision. Berlin: Springer, 2014: 740-755.
- [25] WANG H B, PAN C H, DAVOINE F, et al. Hierarchical fusion of descriptor matching and L-K optical flow[C] // IEEE International Conference on Image Processing. Piscataway: IEEE, 2011: 1893-1896.
- [26] HANDA A, WHELAN T, MCDONALD J, et al. A benchmark for RGB-D visual odometry 3D reconstruction and SLAM[C] // IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2014: 1524-1531.
- [27] AI Y B, RUI T, YANG X Q, et al. Visual SLAM in dynamic environments based on object detection[J]. Defence Technology, 2021, 17(5): 1712-1721.
- [28] JIAO J C, WANG C X, LI N, et al. An adaptive visual dynamic-SLAM method based on fusing the semantic information[J]. IEEE Sensors Journal, 2022, 22(18): 17414-17420.