



基于概率运动统计特征匹配的单目视觉 SLAM

曾维林, 刘桂华, 陈 豪

(西南科技大学 信息工程学院, 四川 绵阳 621010)

摘 要: 在单目视觉同步定位与建图(SLAM)过程中, 由于特征匹配阶段存在误匹配且耗时长, 使得机器人初始化速度慢、定位精度不高。针对此问题, 基于概率运动统计特征匹配, 提出一种单目视觉 SLAM 算法。通过设置自适应的阈值提取 ORB 特征点并使用四叉树进行保存, 根据运动的平滑性与特征匹配的一致性估计特定区域内特征匹配的概率模型, 得到正确的特征匹配点进行匹配, 完成系统自动初始化与机器人位姿跟踪。在 TUM 数据集上的实验结果表明, 该算法在特征匹配阶段耗时仅为 1.4 ms, 机器人初始化时间和定位精度分别为 1.7 s 和 0.54 cm, 且具有良好的实时性。

关键词: 单目视觉同步定位与建图; 概率运动统计特征匹配; ORB 特征; 初始化; 定位精度

开放科学(资源服务)标志码(OSID):



中文引用格式: 曾维林, 刘桂华, 陈豪. 基于概率运动统计特征匹配的单目视觉 SLAM[J]. 计算机工程, 2019, 45(12): 222-231, 236.

英文引用格式: ZENG Weilin, LIU Guihua, CHEN Hao. Monocular vision SLAM based on probabilistic motion statistics feature matching [J]. Computer Engineering, 2019, 45(12): 222-231, 236.

Monocular Vision SLAM Based on Probabilistic Motion Statistics Feature Matching

ZENG Weilin, LIU Guihua, CHEN Hao

(School of Information Engineering, Southwest University of Science and Technology, Mianyang, Sichuan 621010, China)

【Abstract】 In the current monocular vision Simultaneous Localization and Mapping (SLAM) process, the feature matching phase is time-consuming, and mismatches tend to occur, so the robot often has a slow initialization speed and low positioning accuracy. To address these problems, we propose a monocular vision SLAM algorithm based on probabilistic motion statistical feature matching. Firstly, we set the adaptive threshold to extract the ORB feature points, which are simultaneously saved by the quadtree. Then, according to the smoothness of motion and the consistency of feature matching, we estimate the probability model of feature matching in a specific region. Thereby, the correct feature matching points are obtained and the system automatically initialization and robot pose tracking are achieved subsequently. The experimental results on TUM dataset show that the algorithm only takes 1.4 ms in the feature matching phase. Besides, the robot initialization time and positioning accuracy are 1.7 s and 0.54 cm respectively, which represents a good real-time performance.

【Key words】 monocular vision Simultaneous Localization and Mapping (SLAM); probabilistic motion statistics feature matching; ORB features; initialization; positioning accuracy

DOI: 10.19678/j.issn.1000-3428.0052746

0 概述

同步定位与建图 (Simultaneous Localization and Mapping, SLAM) 是指机器人在一个陌生的环境中, 利用自身传感器获取的信息进行实时定位并

构建周围环境地图^[1-2]。视觉传感器因其体积小、价格低廉并且蕴含丰富的环境信息, 被广泛地应用于单目视觉 SLAM 系统中^[3]。根据位姿估计算法的不同, 单目视觉 SLAM 可分为特征点法和直接法。

基金项目: 国防科工局核能开发科研项目“核应急处置机器人关键技术研究”([2016]1295); 四川省科技厅重点研发项目“灵巧型激光自动导引运输装置 (AGV) 研制及其产业化”(17ZDYF0126)。

作者简介: 曾维林 (1993—) 男, 硕士研究生, 主研方向为图像处理、三维重建、SLAM 算法; 刘桂华, 教授、博士; 陈 豪, 本科生。

收稿日期: 2018-09-25 **修回日期:** 2019-01-01 **E-mail:** 2937496455@qq.com

PTAM-SLAM^[4] 算法是特征点法单目视觉 SLAM 的代表算法。该算法采用关键帧构建地图,并使用 BA(Bundle Adjustment)优化^[5] 相机位姿,替代了传统的基于滤波算法的特征点位置和相机位姿估计,提高了位姿估计精度。但是 PTAM-SLAM 算法没有大型的闭环检测,不能消除位姿估计中的累计误差,同时视角不变性较差,易丢失跟踪目标。文献[6-7]提出 ORB-SLAM2 算法,其采用全局 BA 优化与闭环检测算法,相比 PTAM-SLAM 算法具有更高的定位精度。

直接法单目视觉 SLAM 的代表算法有 LSD-SLAM^[8] 和 DSO^[9] 算法,LSD-SLAM 算法可以在不使用 GPU 的情况下大范围构建半稠密地图,同时直接优化位姿,但是该算法定位精度明显低于 PTAM-SLAM 和 ORB-SLAM2 算法。DSO 算法定位精度高于 LSD-SLAM 算法,但是没有闭环检测。直接法虽然可以在不计算特征点和描述子的基础上构建半稠密地图,但是其对光照变化较敏感,限制了系统的使用环境。相比直接法单目视觉 SLAM,基于特征点的 SLAM 算法虽然需要进行两帧图像之间的特征匹配,但是可以在一定程度上克服光照影响。

目前主流的特征匹配算法有暴力匹配、FLANN^[10-12]、RANSAC^[13-14]、CODE^[15] 以及 RepMatch^[16] 等。其中:暴力匹配、FLANN 和 RANSAC 是经典的匹配算法,其匹配速度较快,然而误匹配较多使得 SLAM 系统初始化速度慢,定位精度不高;CODE 对于具有重复结构的环境误匹配较多,RepMatch 虽然解决了该问题,匹配效果好,但是这 2 种算法匹配速度慢,主要应用于对实时性要求不高的三维重建系统中,不适用于对实时性要求较高的 SLAM 系统。

为消除 SLAM 特征匹配过程中的误匹配,保持系统的实时性,同时获得较好的初始化位姿估计结果,本文基于概率运动统计特征匹配设计一种单目视觉 SLAM 算法。

1 算法框架

基于概率运动统计特征匹配的单目视觉 SLAM 算法流程如图 1 所示,该算法主要分为特征提取、概率运动匹配、初始化以及位姿跟踪 4 个阶段。特征提取采用改进的 ORB 特征提取算法,首先将当前帧图像通过图像金字塔生成 8 个不同尺度的图像,然后将不同尺度的图像同时网格化,利用自适应 FAST 角点提取算法在各个尺度上提取 FAST^[17] 特征点,确保每个区域都有一定的特征点,以提高鲁棒性,最后计算特征点的 BRIEF^[18] 描述

子,生成 ORB 特征点,保证旋转不变性,同时以四叉树的形式保存特征点。概率运动特征匹配的核心思想是根据贝叶斯规则与运动的平滑性寻找图像中正确的特征匹配点。首先根据四叉树的结构特点,计算四叉树同一层的 ORB 特征点之间的 Hamming 距离,然后寻找每个匹配的特征点周围是否存在其他匹配点,通过特征点周围匹配点的数量,得到该特征点的匹配正确的概率,最后使用 fisher^[19] 判别准则思想,通过设定阈值,判断匹配的正确性。特征匹配点数量达到设定值后,使用匹配的特征点进行初始化。针对平面结构和非平面结构,根据匹配的特征点分别计算出单应性矩阵 H 和基本矩阵 F ,通过重投影误差计算出 H 矩阵和 F 矩阵的得分,再经过三角化恢复出初始的地图点,并据此进行位姿跟踪。

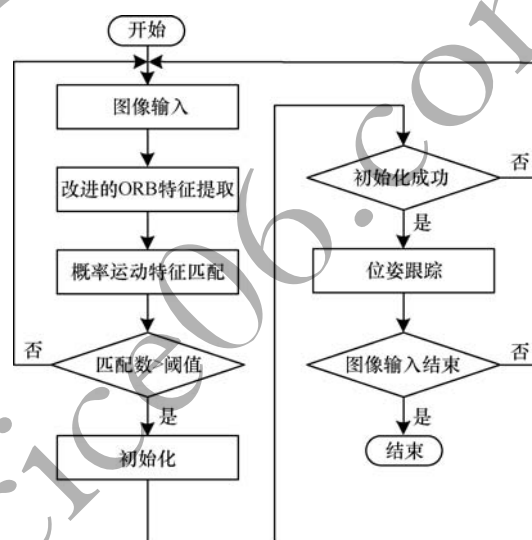


图 1 概率运动特征匹配算法流程

2 概率运动统计特征匹配 SLAM

特征提取与匹配是单目视觉 SLAM 的关键环节。目前主要的特征提取算法有 SIFT^[20]、SURF^[21] 和 ORB^[22] 等。经典特征提取算法 SIFT 由于建立 128 维特征描述子,因此运行速度较慢。SURF 在 SIFT 上进行了改进,虽然提高了算法运行速度,但是其实时性依然不足。

ORB 特征点由于其提取与匹配速度快,是目前视觉 SLAM 主要使用的特征点之一。但是仅采用 ORB 特征点会导致较多误匹配,同时针对于不同的场景 ORB 特征点分布可能集中于图像的一个区域,使得位姿估计存在随机误差。采用改进的 ORB 特征提取和概率运动统计特征匹配相结合的算法,可以得到分布均匀且正确的特征匹配,以避免产生随机误差,提高位姿跟踪精度。

2.1 改进的 ORB 特征提取

由于单目视觉 SLAM 中特征点的提取速度影响算法实时性,同时图像中均匀分布的特征点能减少随机误差,因此本文提出一种改进的 ORB 特征点提取算法,其具体步骤如下:

1) 构建高斯图像金字塔,设置金字塔尺度为 1.2,层数为 8,具有尺度不变性。将金字塔每层图像进行网格划分,每个区域大小设置为 30×30 ,实现特征点均匀分配。采用自适应阈值的方法提取 FAST 特征点,首先设定 FAST 的初始阈值为 12,最小阈值为 6,然后使用初始阈值在每层图像特定区域内提取 FAST 特征点,若没有提取到特征点,减小阈值再次提取,保证特征点的均匀分布。

2) 在特征点半径为 3 的圆形区域内,计算该区域图像质心,得到特征点的主方向,保证旋转不变性,同时使用 BRIEF 算法计算该特征点的描述子。

3) 如图 2 所示,将图像中的特征点进行四叉树划分,根据区域划分的大小与顺序依次将区域内的特征点放入四叉树的节点与叶子节点。由于只有在同一节点的特征点才可能是匹配点,因此可以提高匹配精度和速率。

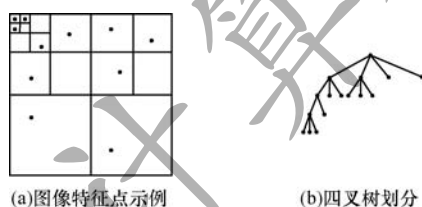


图 2 四叉树的划分

2.2 概率运动统计特征匹配

特征匹配是单目视觉 SLAM 关键环节之一,将得到的特征点通过描述子进行匹配能有效地解决 SLAM 信息关联问题,确定相机前后 2 帧之间的运动关系。基于概率运动统计的特征匹配算法可以提高前后 2 帧图像之间特征匹配的正确性,算法具体步骤如下:

1) 特征点预匹配

对于前后 2 帧图像,根据四叉树结构的特点,将属于同一节点的特征点取出,通过特征点的 BRIEF 描述子计算 2 个特征点之间的 Hamming 距离,设定最小距离阈值为 70,对 2 幅图像的 ORB 特征点进行预匹配。

2) 概率模型估计

如图 3 所示,假设图像对 $\{I_a, I_b\}$ 中特征点数分别为 $\{M, N\}$,所有的特征匹配点集 $X = \{x_1, x_2, \dots, x_N\}$, x_i 代表第 i 个特征匹配点对。图像对 $\{I_a, I_b\}$ 中区域 $\{a, b\}$ 的特征点数为 $\{n, m\}$,特征匹配 $X_i \subseteq X$,

记 S_i 为区域 $\{a, b\}$ 中支持 x_i 的特征匹配的个数。

$$S_i = |X_i| - 1 \quad (1)$$

其中, $|X_i|$ 表示区域内特征匹配个数的总和。

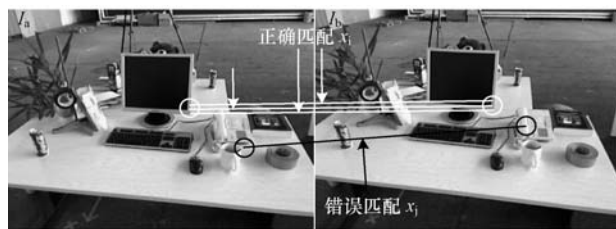


图 3 概率模型估计图

定义 f_a^b 为区域 a 中一个特征点的特征匹配点在区域 b 中, f_a^f 表示区域 a 中一个特征点匹配正确的概率 $p(f_a^f) = t$, 则 f_a^f 表示区域 a 中一个特征点匹配错误的概率 $p(f_a^f) = 1 - t$ 。假设区域 a 中特征点匹配错误,它的匹配点可以落在图像 I_b 中的任意位置,则在匹配错误的条件下,匹配点在区域 b 中的概率为:

$$p(f_a^b | f_a^f) = \beta m / M \quad (2)$$

其中, β 为容错因子,用来容纳个别违反该定义的特征点以提高准确性。

如图 3 所示,白色圆圈区域 T^{ab} 为区域 $\{a, b\}$ 在前后 2 帧场景中相同位置,黑色圆圈区域 F^{ab} 为区域 $\{a, b\}$ 在前后 2 帧场景中的不同位置。根据贝叶斯准则,在 T^{ab} 的条件下,区域 a 中的特征匹配点在 b 中的概率 p_t 为:

$$\begin{aligned} p_t &= p(f_a^b | T^{ab}) = p(f_a^f | T^{ab}) + p(f_a^f, f_a^b | T^{ab}) = \\ &= p(f_a^f | T^{ab}) + p(f_a^f | T^{ab}) p(f_a^b | f_a^f, T^{ab}) = \\ &= p(f_a^f) + p(f_a^f) p(f_a^b | f_a^f) = t + (1 - t) \beta m / M \end{aligned} \quad (3)$$

在 F^{ab} 的条件下,区域 a 中的特征匹配点在 b 中的概率 p_f 为:

$$\begin{aligned} p_f &= p(f_a^b | F^{ab}) = p(f_a^f, f_a^b | F^{ab}) = \\ &= p(f_a^f | F^{ab}) p(f_a^b | f_a^f, F^{ab}) = \\ &= p(f_a^f) p(f_a^b | f_a^f) = (1 - t) \beta m / M \end{aligned} \quad (4)$$

p_t 的物理含义是指区域 a 中的特征点匹配在区域 b 中的概率,即实际匹配正确的概率, p_f 则是指实际匹配错误的概率。由于每个特征点是相互独立的,因此区域 a 中的特征点匹配满足二项分布:

$$S_i \sim \begin{cases} B(n, p_t) \\ B(n, p_f) \end{cases} \quad (5)$$

其中, n 表示区域 a 中所有的特征点数。

3) 概率运动统计

正确匹配点周围领域中的特征点是匹配的。当

运动尺度过大时,为保证算法的鲁棒性,根据运动的平滑性,可以将区域 a 扩展到其领域范围,如图 4 所示。

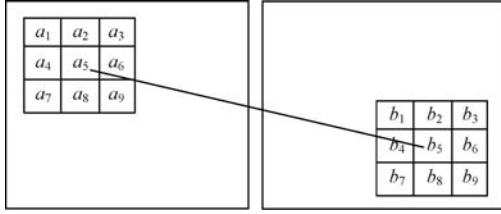


图 4 区域扩展示意图

统计 K 个区域范围内的特征匹配对 S_i 有:

$$S_i = \sum_{k=1}^K |X_{a_k b_k}| - 1 \quad (6)$$

其中, K 表示区域 a 扩展区域的个数, 根据 S_i 满足二项分布, 如式(7)所示。

$$S_i \sim \begin{cases} B(Kn, p_t) \\ B(Kn, p_f) \end{cases} \quad (7)$$

该区域内正确匹配和错误匹配的均值和标准差为:

$$\begin{cases} m_t = Knp_t \\ s_t = \sqrt{Knp_t(1-p_t)} \end{cases} \quad (8)$$

$$\begin{cases} m_f = Knp_f \\ s_f = \sqrt{Knp_f(1-p_f)} \end{cases} \quad (9)$$

其中, m_t, s_t 是正确匹配的均值和方差, m_f, s_f 是错误匹配的均值和方差。利用 fisher 判别思想, 定义区别正确匹配和错误匹配可分性度量 P 为:

$$P = \frac{m_t - m_f}{s_t - s_f} = \frac{Knp_t - Knp_f}{\sqrt{Knp_t(1-p_t)} + \sqrt{Knp_f(1-p_f)}} \quad (10)$$

最大化 P 值就是使得正确匹配和错误匹配的均值之差足够大, 标准差之和足够小, 式(10)可以等价:

$$P \propto \sqrt{Kn} \quad (11)$$

由式(11)可以得到, 区域 a 中特征点数量越多或者 K 值越大, 则正确匹配和错误匹配可分性就越大。但是图像中的特征点个数和领域数 K 都与系统运行时间成正比, 通常适当增大 K 值提高区分度 P 值的大小。

4) 运动网格加速

由于对于每一个特征匹配对需要估计其周围领域得分 S_i , 不满足实时性要求, 因此本文采用运动网格统计模式。将图像对划分成 $N \times N$ 网格, 每个网格仅计算一次领域得分 S_i , 同时网格移动步长为半

个格子的宽度, 保证格子边缘特征点的正确统计。如图 5 所示, 每一对格子 $\{a, b\}$ 的得分 S_{ab} 为:

$$S_{ij} = \sum_{k=1}^{K=9} |X_{i_k j_k}| \quad (12)$$

其中, $|X_{i_k j_k}|$ 表示图像中网格 $\{i_k, j_k\}$ 的特征匹配数量。

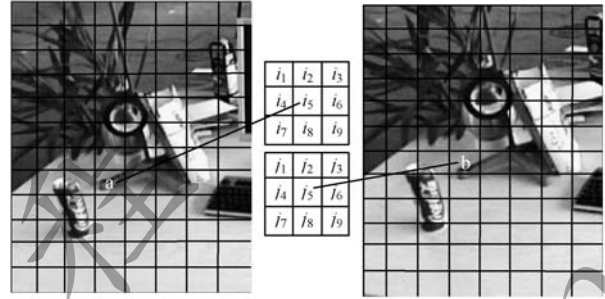


图 5 图像网格划分

网格数量越多匹配的准确性越高, 但此时区域特征数量 n 值减少, 区分度 P 值降低。如果通过提高 K 值进行补偿, 算法运行时间将增多。因此, 本文实验采用 10×10 网格进行特征匹配的筛选, 能同时满足实时性与精度要求。

如图 6 所示, 需要选择一个阈值区分网格对匹配的正确性。

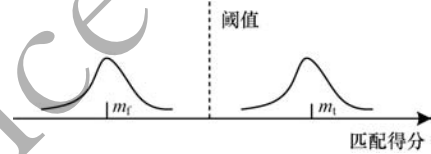


图 6 阈值判别图

阈值 t 的计算公式为:

$$t = m_t + \alpha s_t \approx \alpha \sqrt{n} \quad (13)$$

当网格对的匹配得分 $S_{ij} > t$ 时, 认为是正确匹配, 否则是错误匹配。

2.3 自动初始化

由于单目相机存在深度不确定性, 单目视觉 SLAM 需要进行初始化位姿估计, 构建初始的三维地图点。初始化算法的具体步骤如下:

1) 设定初始化特征提取数量为 2 000, 采用改进的 ORB 特征提取算法对连续两帧图像进行特征提取, 只有当连续两帧图像提取的特征点数量均大于 200 时才进行初始化操作。

2) 假设参考帧 F_r 和当前帧 F_c 在图像中的特征匹配点为 $\{x_r, x_c\}$, 根据式(14)采用 8 点法和 SVD^[23] 相结合的方式求解单应性矩阵 H_{cr} 和基本矩阵 F_{cr} 。

$$\begin{cases} \mathbf{x}_c = \mathbf{H}_{cr} \mathbf{x}_r \\ \mathbf{x}_c^T \mathbf{F}_{cr} \mathbf{x}_r = 0 \end{cases} \quad (14)$$

其中,参考帧和当前帧图像特征点坐标分别为 $\mathbf{x}_c = (u_c, v_c, 1)^T$, $\mathbf{x}_r = (u_r, v_r, 1)^T$ 。单应性矩阵 \mathbf{H}_{cr} 表示处于同一平面上的匹配点在 2 帧图像之间的映射关系,基本矩阵 \mathbf{F}_{cr} 表示 2 个匹配点之间的空间位置关系,两者均为 3×3 矩阵。

3) 利用重投影误差对 \mathbf{H}_{cr} 和 \mathbf{F}_{cr} 进行得分统计,根据其得分 S_H 和 S_F 采用式(15)判断使用哪种矩阵进行相机当前位姿估计。

$$R_H = \frac{S_H}{S_H + S_F} \quad (15)$$

其中, R_H 表示评估参数。当 $R_H > 0.45$ 时,选用单应性矩阵 \mathbf{H}_{cr} , 否则选择基本矩阵 \mathbf{F}_{cr} 恢复当前相机位姿。

4) 判断参考帧和当前帧之间的视差角是否大于 5° , 大于 5° 则使用当前相机位姿和三角化算法恢复出当前初始三维地图点, 同时判断地图点数量是否大于 90, 小于 90 则重新初始化。

2.4 位姿跟踪

在获得正确的初始化位姿和点云地图之后, 采用参考帧模式和恒速模式对相机位姿进行跟踪。如图 7 所示, 将初始化的最后一帧设置为当前帧的参考帧。



图 7 参考帧模型

参考帧模式为：

$$SE3_k \approx SE3_{RF} \quad (16)$$

其中, $SE3_k$ 表示当前帧位姿矩阵的李群, $SE3_{RF}$ 表示参考帧位姿矩阵的李群, 矩阵大小均为 4×4 。采用式(16)将参考帧的位姿设置为当前帧位姿, 同时根据概率统计模型对当前帧和参考帧进行特征匹配。将匹配部分的参考帧的地图点设置为当前帧特征点对应的地图点, 通过 BA 优化当前帧位姿。

在参考帧模式设置成功之后, 采用恒速模式对相机位姿进行跟踪, 如图 8 所示, 恒速模式认为相机近似匀速运动。

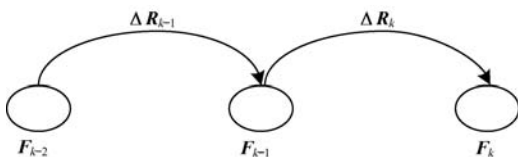


图 8 恒速模型

恒速模式为：

$$\Delta R_k \approx \Delta R_{k-1} \quad (17)$$

其中, ΔR_k 和 ΔR_{k-1} 表示前后 2 帧之间的相对旋转, 均为 3×3 的矩阵。采用式(17)得到当前帧的旋转矩阵, 通过近似匀速运动的特点得到平移向量, 进而得到当前帧的位姿。再通过上一帧特征点对应的地图点重投影到当前帧, 搜索与当前帧的匹配点, 构建当前帧的特征点对应的地图点, 同时使用 BA 优化得到最终当前帧的位姿矩阵。

在跟踪相机位姿过程中, 将获得的地图点存储在局部地图中构建稀疏地图, 同时采用 DBow2^[24-25] 进行图像相似度的判断。当检测到闭环时, 采用全局 BA 对相机位姿进行优化, 得到最后的相机运动轨迹。

3 实验验证与结果分析

本文在 ubuntu14.04 操作系统上采用 C++ 编程, 系统 CPU 为 Intel i7 2.20 GHz, 内存为 16 GB。

3.1 特征提取

本文使用 TUM^[26] 中的 fr2/desk 数据集对 ORB 和改进的 ORB 特征点提取算法进行测试, 实验提取 1 000 个特征点, 结果如图 9 所示。从图 9 可以看出, 改进后的 ORB 特征点提取算法可以在图像上获得均匀分布的特征点。



(a) 原始 ORB 特征点



(b) 改进的 ORB 特征点

图 9 ORB 特征点分布比较

本文同时将改进的 ORB 特征点提取算法与 SIFT、SURF、ORB 特征点提取算法进行实时性对比。各对比算法分别对图 3 场景提取 1 000 个的特征点,为了消除随机误差,计算 10 次特征提取的平均值,结果如表 1 所示。

特征点提取算法	时间	ms
改进的 ORB	21.89	
原始的 ORB	24.95	
SIFT	376.06	
SURF	121.56	

从表 1 可以看出,改进后的 ORB 特征点提取算法运行速度快于原始的 ORB 特征点提取算法,实时性远高于 SIFT 和 SURF 算法,满足单目视觉 SLAM 的实时性要求。

3.2 特征匹配

基于 TUM 中的 fr2/desk 数据集,使用改进的 ORB 特征点进行概率运动统计匹配,并且与 BF (Brute Force) 匹配和 FLANN 匹配进行对比,实验结果如图 10 所示。

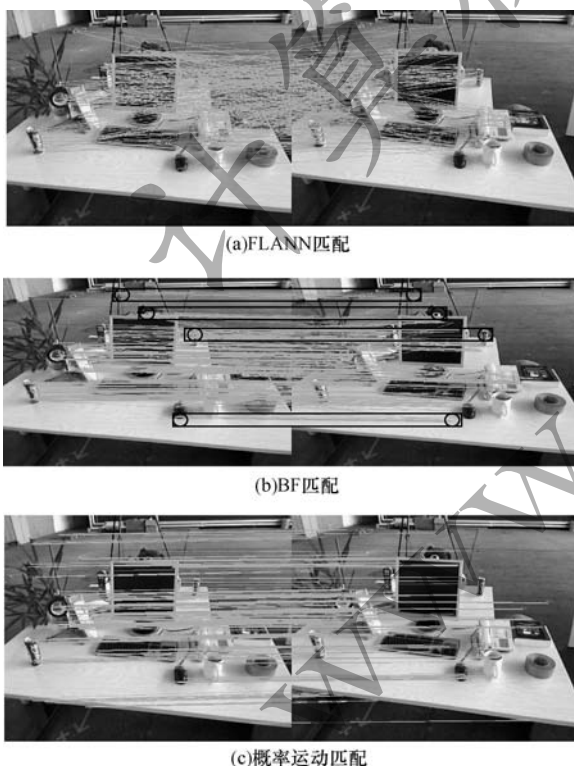


图 10 特征匹配比较

如图 10(a) 所示,FLANN 匹配完全是混乱的,前后 2 帧图像存在大量误匹配。图 10(b) 中黑色矩形框内的圆圈区域表示前后两帧图像部分错误的特征匹配,可以看出 BF 匹配依然存在较多的误匹配。如图 10(c) 所示,概率运动匹配可以获得正确的特征匹配,确保位姿估计的正确性。

分别对前后 2 帧图像提取 1 000 和 1 500 个改进的 ORB 特征点,进行概率运动匹配、BF 匹配和 FLANN 匹配 3 种算法运行时间的对比,如表 2 所示。

匹配算法	1 000 个特征点	1 500 个特征点	ms
概率运动匹配	1.441	2.464	
BF 匹配	10.769	15.476	
FLANN 匹配	98.569	143.014	

从表 2 可以看出,FLANN 匹配耗时长,BF 匹配在特征点较少时耗时较短,但是随着特征点的增多,BF 匹配时间明显上升。概率运动统计匹配相对于 BF 匹配和 FLANN 匹配耗时最少,完全满足视觉 SLAM 中实时性要求。

3.3 初始化

由于采用概率运动统计特征匹配能得到正确的特征匹配,因此当相机运行到具有足够的视差(实验设置为 5°)时,根据式(15)的得分选择平面结构或是非平面结构,恢复出足够的地图点,完成初始化过程。在 TUM 中的 fr2/desk 数据集上进行初始化测试,实验结果如图 11 所示。

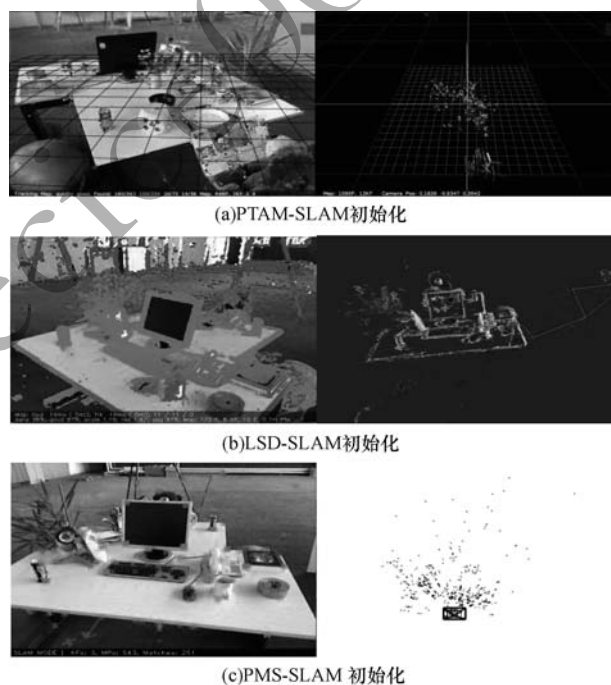


图 11 SLAM 系统初始化比较

从图 11 可以看出,PTAM-SLAM 提取的特征点和 LSD-SLAM 提取的关键点均位于场景的同一平面上,初始化结构为平面型,而基于概率运动统计 SLAM (PMS-SLAM) 提取的改进的 ORB 特征分布于场景各个位置。这是因为 PTAM-SLAM 和 LSD-SLAM 均以平面结构初始化,限制了该类算法的使

用场景,当环境为非平面结构时,容易初始化不成功或是跟踪丢失。

在 ORB-SLAM2 系统中当相机运动获得足够的视差时,由于特征匹配存在误匹配,使得位姿估计存在较大误差或估计错误,进而重复进入初始化过程。本文提出的基于概率运动统计 SLAM 只要获得足够的视差,系统就能完成初始化,获得较好的初始位姿估计,在 TUM 数据集上进行测试,初始化时间、图像帧数、地图点构建数对比如表 3 所示,其中,

表 3 SLAM 系统自动初始化对比

数据集	ORB-SLAM2(单目)			PMS-SLAM		
	时间/s	帧数	地图点	时间/s	帧数	地图点
fr1/xyz	13.325 70	177	106	2.033 17	20	231
fr1/floor	6.707 75(- ∞)	98	98	6.138 72	65	300
fr2/desk	10.325 70	153	124	1.751 10	17	305
fr2/360	67.120 10	846	135	6.342 50	70	160
fr2/kidnap	6.850 11	87	94	5.143 82	32	373
fr2/xyz	3.385 50	47	111	1.769 81	14	466
fr3/xyz	16.475 70	213	98	6.975 98	50	368

3.4 定位精度与实时性分析

3.4.1 离线定位精度分析

由于单目相机存在深度不确定性,因此通过刚体变换将相机轨迹与真实轨迹对齐后采用均方根误差(Root Mean Square Error, RMSE)进行定位精度分析。选取 TUM 数据集集中的 fr1/desk、fr1/floor、

“6.707 75(- ∞)”表示如果初始化成功,其时间为 6.707 75 s,也可能初始化不成功,时间为无穷大。从表 3 可以看出,由于获得足够的正确匹配,PMS-SLAM 能够快速初始化。在 fr2/desk 数据集上,PMS-SLAM 最优的初始化的时间为 1.751 s,帧数为 17,创建地图点数为 305,明显优于单目 ORB-SLAM2,在其他数据集上也有相同的结果,并且其初始化的稳定性也具有明显的优势,避免了单目 ORB-SLAM2 中初始化不成功的风险。

fr2/desk 等 10 个数据集进行测试,将 PMS-SLAM 与 ORB-SLAM2(单目)、LSD-SLAM、PTAM、RGBD-SLAM^[27]进行对比分析,实验结果如表 4 所示。其中,RGBD-SLAM 运动轨迹从 TUM 数据集官方发布数据获取,“—”表示未对此数据集进行测试,“X”表示跟踪丢失。

表 4 离线定位精度 RMSE 对比

数据集	PMS-SLAM	ORB-SLAM2	PTAM-SLAM	LSD-SLAM	RGBD-SLAM
fr1/desk	1.56	1.69	X	10.65	2.52
fr1/floor	2.32	2.99	X	37.03	3.51
fr2/desk	0.81	0.90	X	4.57	9.50
fr2/peron	0.58	0.64	X	31.70	6.97
fr2/360_kidnap	2.57	3.81	2.63	X	100.50
fr2/xyz	0.25	0.30	0.20	2.15	2.61
fr3/long_office	1.29	3.45	X	38.33	—
fr3/nstr_tex_near	1.21	1.39	2.74	7.54	—
fr3/sit_xyz	0.71	0.79	0.83	7.93	—
fr3/str_tex_near	1.41	1.58	1.04	X	—

在使用 LSD-SLAM 估计位姿时,由于采用随机的深度进行初始化,因此去掉前 20 帧数据进行误差估计。从表 4 可以看出,PTAM-SLAM 和 LSD-SLAM 对部分数据集跟踪丢失,而 PMS-SLAM 在所有数据集上均能成功跟踪,表明本文算法具有较强的鲁棒性。在 fr2/xyz 数据集上

本文算法定位精度略低于 PTAM-SLAM,这是因为 PTAM-SLAM 是在人为干预下拥有高精度的平面结构初始化。此外,本文算法定位精度均高于 ORB-SLAM2。在上述数据集中,将本文算法计算出的相机运动轨迹与真实运动轨迹对比,结果如图 12 所示。

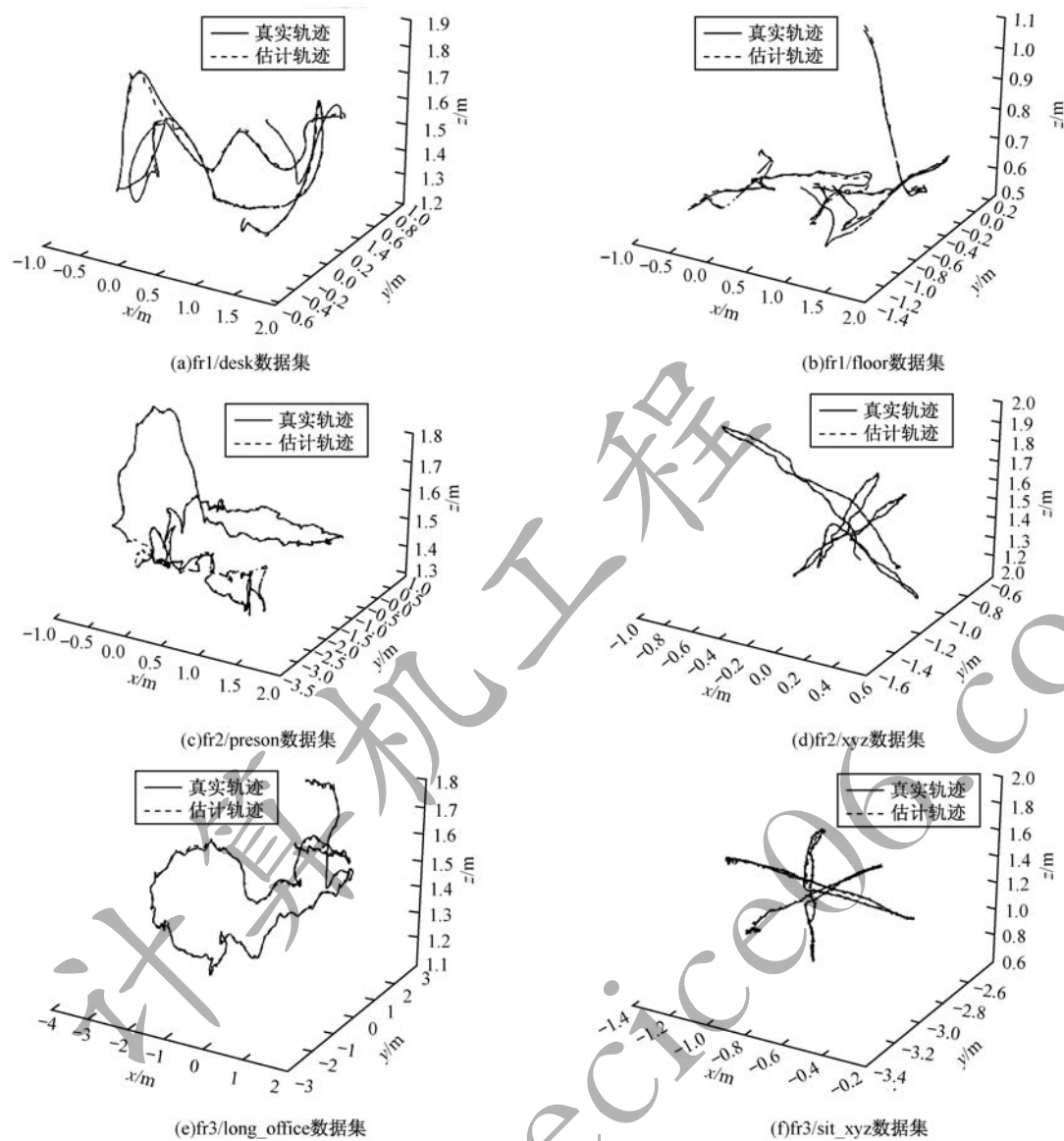


图 12 相机运动轨迹

从图 12 可以看出,相机运动轨迹与真实的轨迹十分相近,验证了本文算法的正确性和稳定性。

3.4.2 离线实时性分析

PMS-SLAM 系统通过多线程进行设计,将追踪、建图、位姿优化、显示使用 4 个不同线程并行执行,以提高系统的实时性。使用 TUM 数据集对 PMS-SLAM 的实时性进行分析,在系统初始化完成之后,计算出系统对后续的单帧图像进行特征提取,与上一帧图像特征匹配,估计出当前帧相机位姿和恢复出当前帧观测到的三维空间点的总时间,即系统对单帧图像的跟踪时间。系统跟踪单帧图像所耗时间的均值如表 5 所示。

表 5 离线跟踪时间统计

数据集	时间/s
fr1/xyz	0.040 3
fr1/floor	0.036 6
fr2/desk	0.043 2
fr2/xyz	0.042 7
fr3/sit_xyz	0.041 1
fr3/str_tex_near	0.041 7

根据表 5 中的数据,计算得到跟踪的平均时间为 0.040 9 s,即系统运行的帧率为 25 Hz,满足单目视觉 SLAM 的实时性要求。

3.4.3 在线定位精度与实时性分析

为了进一步验证系统的定位精度和实时性,本文利用 Turtlebot2 移动机器人进行实际环境的实验测试。设定 FAST 算法的初始阈值为 12,最小阈值

为 6, 金字塔尺度为 1.2, 层数为 8, 在每一层均提取 FAST 特征点。每帧图像提取的改进的 ORB 特征点数最多为 1 000 个, 采用 10×10 网格进行特征匹配筛选。初始化的最小视差角为 5° , 最少的地图点为 90 个。实验系统如图 13 所示。



图 13 实验系统

首先对 Turtlebot2 移动机器人上的相机进行标定, 然后通过摄像头获取图像, 最后进行特征点提取和位姿跟踪。实验环境和特征点提取如图 14 所示。



(a) 实验环境



(b) 特征点提取

图 14 实验环境与特征点提取

使用 Turtlebot2 上的激光雷达 (思岚 RPLIDAR A2), 将激光雷达 SLAM^[28-29] 算法作为定位精度分析的真实轨迹, 统计 6 次实验下 PMS-SLAM 和 ORB-SLAM 的定位精度, 实验结果如表 6 所示。

表 6 在线定位精度 RMSE 对比

实验次数	PMS-SLAM	ORB-SLAM2	cm
1	0.73	0.89	
2	0.98	1.21	
3	1.12	1.93	
4	1.35	2.12	
5	0.54	0.96	
6	1.20	1.33	

从表 6 可以看出, 在以激光数据作为真实轨迹的情况下, PMS-SLAM 算法在每次实验的定位精度均高于 ORB-SLAM2 算法的定位精度。绘制出 PMS-SLAM 算法其中一次的轨迹图, 如图 15 所示。

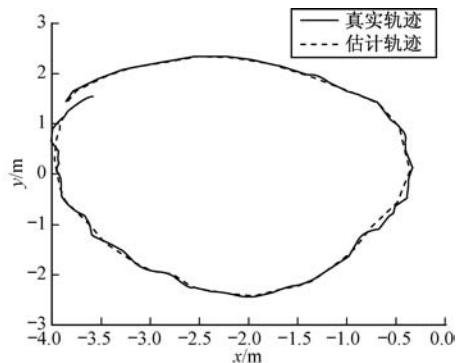


图 15 机器人运行轨迹

从图 15 可以看出, 采用 PMS-SLAM 算法得到的运动轨迹和激光雷达得到的轨迹十分接近, 从而表明本文算法具有较高的定位精度。

在上述 6 次实验中, 本文算法的跟踪时间如表 7 所示。

表 7 在线跟踪时间统计

实验次数	时间/s
1	0.039 2
2	0.036 2
3	0.040 5
4	0.039 6
5	0.041 3
6	0.037 4

根据表 7 中的数据, 计算得到跟踪的平均时间为 0.039 0 s, 即在线实测运行帧率为 25.6 Hz, 满足单目视觉 SLAM 的实时性要求。

图 16 为一次实验生成的实际运动轨迹。

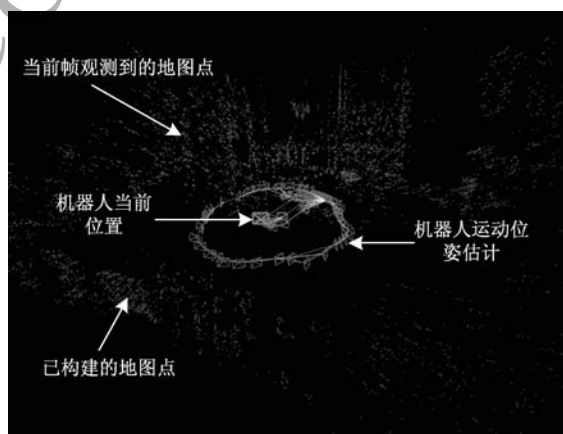


图 16 机器人运动轨迹

4 结束语


本文提出 PMS-SLAM 算法, 采用基于概率统计模型的特征匹配方式, 根据运动的平滑性和正确的特征匹配在图像中的一致性, 对特征匹配进行筛选, 快速得到正确的初始化位姿, 同时恢复出足够的地

图点以进行位姿跟踪。实验结果表明,相比于传统的 SLAM 算法,PMS-SLAM 具有较高的定位精度,并且满足机器人定位的实时性需求。由于本文算法采用的是特征点法,因此其在低纹理特征的情况下容易出现跟踪丢失的现象,而在这种环境下,直接法的视觉 SLAM 就具有明显的优势,下一步将结合直接法并基于深度学习实现复杂环境下机器人的高精度定位,同时提高系统的稳定性和实时性。

参考文献

- [1] CADENA C, CARLONE L, CARRILLO H, et al. Past, present, and future of simultaneous localization and mapping: toward the robust-perception age [J]. IEEE Transactions on Robotics, 2016, 32(6): 1309-1332.
- [2] DUAN Huaxu, YAN Fei, ZHUANG Yan, et al. Simultaneous localization and mapping for UGVs with binocular camera [J]. Journal of Huazhong University of Science and Technology (Nature Science Edition), 2015, 43 (z1): 319-323. (in Chinese)
段华旭, 闫飞, 庄严, 等. 无人车基于双目视觉的同时定位与地图构建 [J]. 华中科技大学学报(自然科学版), 2015, 43(增刊): 319-323.
- [3] LIU Haomin, ZHANG Guofeng, BAO Hujun. A survey of monocular simultaneous localization and mapping [J]. Journal of Computer-Aided Design & Computer Graphics, 2016, 28(6): 855-868. (in Chinese)
刘浩敏, 章国锋, 鲍虎军. 基于单目视觉的同时定位与地图构建方法综述 [J]. 计算机辅助设计与图形学学报, 2016, 28(6): 855-868.
- [4] KLEIN G, MURRAY D. Parallel tracking and mapping for small AR workspaces [C]//Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality. Washington D. C., USA: IEEE Press, 2007: 1-10.
- [5] TRIGGS B, MCLAUCHLAN P F, HARTLEY R I, et al. Bundle adjustment a modern synthesis [C]//Proceedings of International Workshop on Vision Algorithms. Berlin, Germany: Springer, 1999: 298-372.
- [6] MUR-ARTAL R, MONTIEL J M M, TARDOS J D. ORB-SLAM: a versatile and accurate monocular SLAM system [J]. IEEE Transactions on Robotics, 2015, 31(5): 1147-1163.
- [7] MUR-ARTAL R, TARDÓS 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] ENGEL J, SCHÖPS T, CREMERS D. LSD-SLAM: large-scale direct monocular SLAM [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2014: 834-849.
- [9] ENGEL J, KOLTUN V, CREMERS D. Direct sparse odometry [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(3): 611-625.
- [10] INDYK P, MOTWANI R. Approximate nearest neighbors: towards removing the curse of dimensionality [C]//Proceedings of the 30th Annual ACM Symposium on Theory of Computing. New York, USA: ACM Press, 1998: 604-613.
- [11] MUJA M, LOWE D G. Fast approximate nearest neighbors with automatic algorithm configuration [C]//Proceedings of VISAPP'09. Lisboa, Portugal: [s. n.], 2009: 1-10.
- [12] MUJA M, LOWE D G. Scalable nearest neighbor algorithms for high dimensional data [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(11): 2227-2240.
- [13] FISCHLER M A, BOLLES R C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography [J]. Communications of the ACM, 1981, 24(6): 381-395.
- [14] RAGURAM R, FRAHM J M, POLLEFEYS M. A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2008: 500-513.
- [15] LIN Wenyan, WANG Fan, CHENG Mingming, et al. Code coherence based decision boundaries for feature correspondence [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(1): 34-47.
- [16] LIN Wenyan, LIU Siying, JIANG Nianjuan, et al. Repmatch: robust feature matching and pose for reconstructing modern cities [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2016: 562-579.
- [17] ROSTEN E, DRUMMOND T. Machine learning for high-speed corner detection [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2006: 430-443.
- [18] CALONDER M, LEPETIT V, STRECHA C, et al. Brief: binary robust independent elementary features [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2010: 778-792.
- [19] ZHAO Liha. Research and improvement of Fisher discriminant analysis method [D]. Harbin: Northeast Forestry University, 2013. (in Chinese)
赵丽娜. Fisher 判别法的研究及应用 [D]. 哈尔滨: 东北林业大学, 2013.
- [20] LOWE D G. Distinctive image features from scale-invariant keypoints [J]. International Journal of Computer Vision, 2004, 60(2): 91-110.
- [21] BAY H, TUYTELAARS T, VAN GOOL L. Surf: speeded up robust features [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2006: 404-417.
- [22] RUBLEE E, RABAU D V, KONOLIGE K, et al. ORB: an efficient alternative to SIFT or SURF [C]//Proceedings of IEEE International Conference on Computer Vision. Washington D. C., USA: IEEE Press, 2011: 1-8.
- [23] HARTLEY R, ZISSERMAN A. Multiple view geometry in computer vision [M]. Cambridge, UK: Cambridge University Press, 2003.
- [24] GÁLVEZ-LÓPEZ D, TARDOS J D. Bags of binary words for fast place recognition in image sequences [J]. IEEE Transactions on Robotics, 2012, 28(5): 1188-1197.
- [25] STRASDAT H, MONTIEL J, DAVISON A J. Scale drift-aware large scale monocular SLAM [J]. Robotics: Science and Systems, 2010, 2(3): 73-80.

(上接第 231 页)

- 
- [26] STURM J, ENGELHARD N, ENDRES F, et al. A benchmark for the evaluation of RGB-D SLAM systems [C]// Proceedings of 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. Washington D. C., USA: IEEE Press, 2012: 573-580.
- [27] ENDRES F, HESS J, ENGELHARD N, et al. An evaluation of the RGB-D SLAM system[J]. Proceedings of IEEE International Conference on Robotics and Automation. Washington D. C., USA: IEEE Press, 2012: 1691-1696.
- [28] KOHLBRECHER S, VON STRYK O, MEYER J, et al. A flexible and scalable SLAM system with full 3D motion estimation[C]// Proceedings of 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics. Washington D. C., USA: IEEE Press, 2011: 155-160.
- [29] SANTOS J M, PORTUGAL D, ROCHA R P. An evaluation of 2D SLAM techniques available in robot operating system[C]// Proceedings of 2013 IEEE International Symposium on Safety, Security, and Rescue Robotics. Washington D. C., USA: IEEE Press, 2013: 1-6.

编辑 刘盛龄