

引言

机器人实现应用的关键可概括为3个核心问题:我在哪、我要去哪和如何去^[1]的问题,其中“我在哪”的定位问题是首要问题^[2],而且随着机器人在矿井^[3]、医疗^[4]、服务^[5]等领域的应用,定位精准性的需求逐渐成为人们研究和关注的重点^[6,7].由于单一传感器的定位精度和可适用性有限,目前,机器人定位普遍采用将多种传感器数据进行融合的方式实现^[8-11],其中惯性测量单元(IMU)和相机(camera)存在互补性^[12],因此IMU和Camera传感器的数据融合方式受到了更多青睐.

IMU和Camera数据融合问题又称为VIO(Visual-Inertial Odometry)问题.目前,基于滤波的主流VIO算法有:Multi-Sensor Fusion(MSF)、Multi-State Constraint Kalman Filter(MSCKF)、Robust Visual Inertial Odometry(ROVIO)等.其中MSCKF^[13]是2007年由Mourikis等提出的算法,该算法采用扩展卡尔曼(EKF)^[14]将IMU和单目Camera传感器数据进行EKF融合,融合方法是用IMU运动模型构建EKF状态方程,用单目Camera的重投影误差建立EKF量测模型.

近年来,随着机器人行业的迅速发展,关于MSCKF算法的研究有许多进展.文献[15]从计算代价角度,集成了EKF-SLAM和MSCKF两种算法.首先预测每个特征分配策略的计算成本,然后制定目标函数,最后用目标函数的最小化决定定位系统所使用的融合策略.该算法对状态向量也进行了增广,使得当系统进行更新时,长时间跟踪的特征点能及时得到利用.文献[16]是将原来的单目Camera改为立体Camera,提高定位的稳健性,并且边缘化采用Two-way Marginalization策略,考虑了无人机应用中悬停等特性.文献[17]利用了全方位摄像机获得球形图像,从更宽的视眼内获得更多的视觉特征与IMU测量相结合,避免因环境纹理不足造成定位效果不佳的问题.文献[18]在初始化阶段应用sigma点滤波器,使得当机器人初始化阶段遭受冲击时定位与导航性能仍然可靠.

本文针对传统MSCKF算法使用的IMU中加速度计传感器的自身缺点:测量信息中包含重力信息,且算法在执行时无法很好的将重力加速度有效去除^[19];速度和位置状态方程经积分解算得到,存在偏置漂移累计和距离测量精度随时间恶化^[20]等问题,提出了改进MSCKF算法.改进的MSCKF算法用轮式里程计(Wheeled Odometer,后续使用W-Odom表示)取代加

速度计传感器,利用W-Odom对机器人运动的平移数据测量较为稳定和准确的优点^[19],改进传统MSCKF算法的EKF状态方程.改进的EKF状态方程使用陀螺仪构建姿态方程,使用轮式里程计构建速度和位置方程,避免状态方程受重力干扰和由于积分产生累计误差的问题,使预测更有效,从而提升定位精度.

1 MSCKF 算法

MSCKF是基于EKF滤波的IMU和Camera的数据融合算法.在介绍MSCKF算法之前,首先需要对相关的坐标系进行定义和约定.在本文中, I 表示IMU坐标系,又称为载体坐标系,与载体固连,随载体运动而变化; C 表示相机坐标系; G 表示全局坐标系,是固定不变的坐标系,本文定义 G 系在算法的初始位置,即是以机器人中心点位置为原点, z 轴为垂直载体向上, x 轴指向载体的前部, y 轴指向载体的左侧.上述坐标系均满足右手定则.

MSCKF算法的系统状态向量 X 定义为:

$$X = [X_{\text{IMU}}, {}^I_C q^1, {}^I_{P_C} q^1, {}^I_C q^2, {}^I_{P_C} q^2, \dots, {}^I_C q^N, {}^I_{P_C} q^N]^T \quad (1)$$

其中, $X_{\text{IMU}} = [{}^I_G q, b_\omega, v_I, b_f, {}^G p_I]^T$ 为IMU状态,依次表示IMU的姿态、陀螺仪零偏、机体IMU速度、加速度计零偏和机体IMU在全局坐标系下的位置.剩余部分是Camera位姿状态,为滑窗内 N 帧图像对应的相机姿态 ${}^I_C q$ 和位置 ${}^I_{P_C}$.MSCKF算法的主体是EKF的预测和更新过程,其中预测过程对应EKF状态方程,更新过程对应EKF量测方程.本文提出的改进MSCKF算法是对传统MSCKF算法的EKF状态方程中速度方程和位置方程的替换,因此本文只论述MSCKF算法预测部分的EKF状态方程,并在此基础上论述改进MSCKF算法.MSCKF算法的EKF量测方程参考文献[21].

MSCKF算法的EKF状态方程使用IMU数据对系统状态向量进行预测.IMU测量输出的数据是角速度 ω 和线加速度 f ,将其经坐标转换可得到角速度 $\hat{\omega}$ 和线加速度 \hat{f} 在IMU坐标系下的表示,这其中包含了陀螺仪和加速度计的零偏 b_ω 、 b_f 以及噪声 w_ω 、 w_f ,因此真实的角速度 $\hat{\omega}$ 和加速度 \hat{f} 定义为:

$$\hat{\omega} = \tilde{\omega} - b_\omega - w_\omega \quad (2)$$

$$\hat{f} = \tilde{f} - b_f - w_f \quad (3)$$

MSCKF算法的状态方程推导较为复杂,本文直接

给出连续状态方程,具体推导过程见文献[21]:

$${}^I_G \dot{\hat{q}} = \frac{1}{2} \Omega(\hat{\omega}) {}^I_G \hat{q} \quad (4)$$

$$\hat{b}_\omega = 0_{3 \times 1} \quad (5)$$

$${}^G \dot{\hat{v}} = C_G^T \hat{f} + {}^G g \quad (6)$$

$$\hat{b}_f = 0_{3 \times 1} \quad (7)$$

$${}^G \dot{\hat{p}} = {}^G \hat{v} \quad (8)$$

$${}^I_C \dot{\hat{q}} = 0_{3 \times 1} \quad (9)$$

$${}^I \dot{\hat{q}}_C = 0_{3 \times 1} \quad (10)$$

其中, $\Omega(\hat{\omega}) = \begin{pmatrix} -[\hat{\omega}_\times] & \hat{\omega} \\ \hat{\omega}^T & 0 \end{pmatrix}$, $[\hat{\omega}_\times]$ 是 $\hat{\omega}$ 的反对称矩阵.

式(4)为姿态状态方程,式(6)为速度状态方程,式(8)为位置状态方程.通过对上述方程采用积分方法即可实现对系统状态向量的预测.

2 MSCKF 算法的改进与实现

2.1 改进 MSCKF 算法

由传统 MSCKF 算法的 EKF 状态方程可以看出,速度方程受重力影响,且是对加速度计测量的线加速度数据和重力加速度进行积分得到,位置状态方程是对速度再次积分,即对线加速度数据进行两次积分得到,这使得速度和位置方程受重力干扰,且位置漂移随时间二次增长.针对此问题,本文利用 W-Odom 传感器以及 IMU 中陀螺仪的互补性:陀螺仪测量的角速度短时间内精度较高, W-Odom 对平移测量较为准确,以及室内环境特点:机器人在二维平面内运动,替换 EKF 状态方程中的式(6)和式(8),改进 MSCKF 算法.

首先分析 W-Odom. 已知本文实验使用的 Turtlebot2 机器人采用两轮差速底盘控制,轮子半径为 r ,左右轮之间距离为 L ,轮子行走一圈编码器的总脉冲数为 sum ,设 Δt 时间内左右编码器输出的脉冲数为 N_L 、 N_R ,左右轮的线速度为 V_L 、 V_R ,机器人的速度 v 与角速度 ω_0 ,由式(11)–式(17)可计算出机器人的速度 v 以及位置 x 、 y . 其中式(13)为速度方程,式(16)和式(17)是 W-Odom 位置方程.

$$V_L = \frac{N_L}{sum} \times \frac{\pi \cdot r}{\Delta t} \quad (11)$$

$$V_R = \frac{N_R}{sum} \times \frac{\pi \cdot r}{\Delta t} \quad (12)$$

$$v = \frac{N_L + N_R}{2} \quad (13)$$

$$\omega_0 = \frac{2(N_L - N_R)}{L} \quad (14)$$

$$\theta_{k+1} = \theta_k + \omega_0 \Delta T \quad (15)$$

$$x_{k+1} = x_k + v \cos(\theta_{k+1}) \Delta t \quad (16)$$

$$y_{k+1} = y_k + v \sin(\theta_{k+1}) \Delta t \quad (17)$$

改进 MSCKF 算法首先用 W-Odom 的速度方程式(13)替换 EKF 速度状态方程式(6);其次用 EKF 姿态状态方程式(4)求得姿态角中的航向角 θ ,并用 θ 代替 W-Odom 的位置方程式(16)和式(17)中的 θ ,改进位置方程,最后用改进后的位置方程替换 EKF 位置状态方程式(8),即得到改进的 MSCKF 算法.改进 MSCKF 算法舍弃 IMU 中加速度计的使用,综合利用 IMU 中陀螺仪和 W-Odom 的各自特点,将两种传感器数据进行融合,改进传统 MSCKF 算法的 EKF 状态方程,适用于机器人在室内的运动.

2.2 MSCKF 与改进 MSCKF 算法的实现

2.2.1 MSCKF 与改进 MSCKF 算法的实现流程

MSCKF 算法通过 IMU 数据预测状态向量,通过单目 Camera 对环境进行观测,提取自然路标,并运用路标信息的路标误差约束对预测的状态信息进行修正更新.如图1是传统 MSCKF 算法的流程图,图中的英文对应的是程序里不同的函数.

判断 10 s 内是否有新一帧图像.若在 10 s 内无图像帧到来,则 MSCKF 算法停止状态量更新,结束;若有新一帧图像到来,则进行下一步.

(1) 判断是否初始化.若没有完成初始化,则进行初始化,并构造 MSCKF 状态向量,进行下一步;若已经完成初始化,则直接进行下一步.

(2) EKF 预测.传统 MSCKF 算法输入为 IMU 数据,调用根据第 1 节 EKF 状态方程编写的函数,输出系统状态向量的预测值和对应的协方差矩阵.

(3) 对新图像进行光流跟踪、匹配,并提取新的特征点.

(4) 将新一帧图像的位姿 pose 加入到状态向量和协方差中(增广).

(5) 特征点处理.

① 若之前视图中的特征点在当前帧中观测不到(看不见的特征点), 则将该特征的跟踪列表加入到 measurementUpdate 中, 用于更新 MSCKF 状态向量。

② 若当前帧观测到的特征是之前在视图中已经观测到的特征点(成熟特征点), 则将该特征点加入到跟踪列表。

③ 对新提取的特征点分配新的 featureID, 加入到跟踪列表。

(6) 循环遍历所有加入到 measurementUpdate 中的特征点, 进行 EKF 更新, 得到 MSCKF 系统状态向量的最优解。

① 对特征点进行三角化, 精确得到当前路标点在全局坐标系的位置。

② 特征点的边缘化: 将重投影误差中的关于特征点误差的约束进行边缘化(左零空间投影), 将特征点误差约束转化为系统状态向量的约束(待优化变量)。

③ 利用重投影误差, 采用视觉更新, 进行一次线性化, 求解当前帧时刻系统状态向量的最优解。

(7) 将滑动窗口滑图像帧进行边缘化, 使其维持在固定的长度。

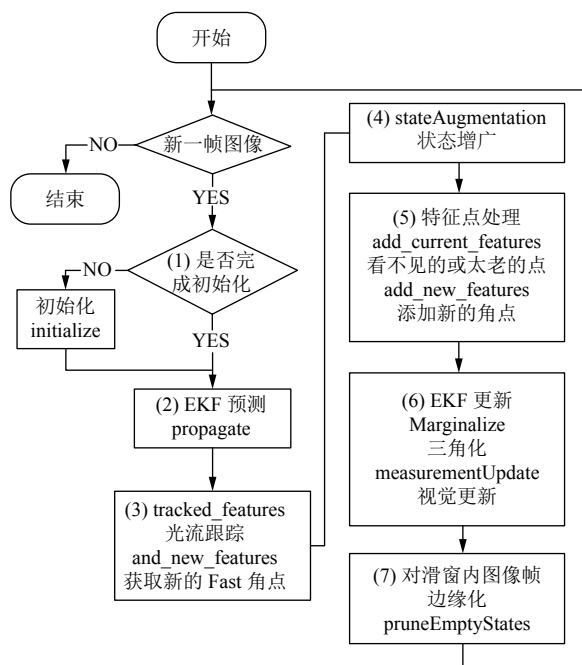


图1 传统 MSCKF 算法流程图

以上是 MSCKF 算法的流程图. 改进 MSCKF 算法是在原算法基础上, 对第二步 EKF 预测部分的状态方程进行改进. 改进算法的输入为 W-Odom 数据和

IMU 中陀螺仪的数据, 且执行的函数(方程)不同, 执行的函数为 im_propagate.

由此, 可以完成一次对当前时刻机器人位置的估计, 当新一帧图像到来之后, 循环执行此步骤, 即实现 MSCKF 算法及其改进算法。

2.2.2 MSCKF 与改进 MSCKF 算法的实现

本文使用 Turtlebot2 机器人, 在装有 Ubuntu 16.04 机器人操作系统(ROS)的 Lenovo G50 笔记本电脑上进行实验测试. 如图 2 所示, 是带有 kokuki 底盘的 Turtlebot2 机器人, 内置了 IMU 和轮式里程计传感器, 在此基础上外加单目相机和核心处理器 Intel i5, 就组成了实验硬件. ROS 是实现机器人定位的软件部分. ROS 可提供用户、计算机操作系统以及外部设备间的通信功能, 并集成了一系列定位、导航等行为的工具和库, 能够在不同的机器人平台(如 Turtlebot)上辅助研究人员进行机器人系统相关的研究和开发, 创建定位、导航等复杂的机器人行为。



图2 Turtlebot2 机器人

本文使用的 MSCKF 及其改进算法以 msckf 和 im_msckf 包封装在 ROS 内. MSCKF 及其改进算法首先订阅陀螺仪、轮式里程计、单目 Camera 发布的话题, 然后执行根据 2.1.1 节编写的算法处理数据, 并以 msg 格式发布 msckf 和 im_msckf 话题. 最后在 Ubuntu 16.04 系统上编写 msckf.py, 该文件订阅 MSCKF 及其改进算法发布的话题, 并以 txt 形式输出 MSCKF 及其改进算法计算的机器人位置坐标。

MSCKF 及改进算法有两个比较重要的文件: src/ros_interface.cpp 和 src/cornet_detector.cpp. 其中 src/ros_interface.cpp 是算法主体, 包括状态预测、状态增广和测量更新等步骤. src/cornet_detector.cpp 是视觉前端处理部分, 主要进行光流跟踪及特征点处理等操作.

3 实验结果分析

本文采用两种策略验证改进 MSCKF 算法的有效性: 一种是使机器人做规定轨迹运动, 用 Matlab 读取 TXT 数据, 绘制运动轨迹图, 对比真实轨迹与算法轨迹, 直观的观察机器人运动过程中的定位精度; 第二种策略不关注机器人的真实运动轨迹, 关注机器人运动时间长短与闭环定位精度的关系, 因此使机器人做闭环轨迹运动, 并将 (0.00, 0.00) 与算法计算的终点位置的距离作为衡量定位准确度的指标 (闭环误差), 绘制闭环误差表, 统计平均闭环误差. 其中机器人运动的实验环境为实验室和走廊两种环境, 如图 3 所示.



图 3 实验环境

3.1 规定轨迹运动

遥控使机器人做封闭规定轨迹运动, 实验运动环境为图 3(a). 实验前预先在路径上设置 4 个坐标点 (0.00, 0.00)、(3.80, 0.00)、(3.80, 2.70)、(2.70, 0.00), 规定机器人路线为矩形, 运动距离为 13 m, 且各坐标点之间的运动为直线, 运动一圈后回到起点位置 (0.00, 0.00).

图 4 为一次实验 Matlab 绘制的运动轨迹图. 图中五角星为机器人起点及终点位置, 直线轨迹为规定的机器人运动轨迹, x 号轨迹为使用传统 MSCKF 算法的 IMU/Camera 运动轨迹, 圆圈轨迹为改进 MSCKF 算法的 IMU/W-Odom/Camera 运动轨迹. 图中可直观的看出, 改进 MSCKF 算法的整体运动轨迹更接近真实轨迹, 计算的终点位置也更接近起点 (0.00, 0.00).

3.2 封闭轨迹运动

遥控使机器人在走廊内从起点 (0.00, 0.00) 开始运

动, 最后回到起点 (0.00, 0.00), 即做闭环轨迹运动, 且实验中机器人的运动为直线和转弯的结合. 为更好的分析闭环定位误差, 减小由于运动时间、环境、操作误差和偶然性对实验结果的干扰, 本文在两种环境下进行了运动时间不同的 5 次实验, 其中, 实验 2 和实验 4 实验环境为图 3(a), 其余实验环境为图 3(b), 统计的运动闭环误差如表 1. 表中 x、y 表示算法计算得到的终点坐标, 闭环误差是该终点坐标与起点 (0.00, 0.00) 的距离.

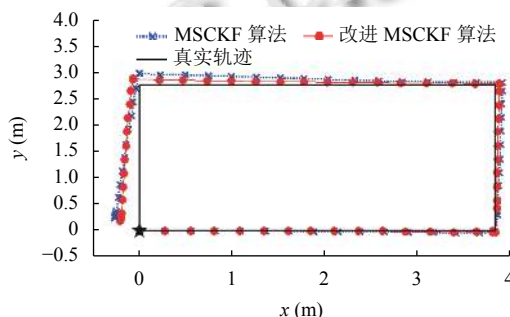


图 4 规定路线运动轨迹

表 1 闭环误差

实验	时长 (s)	MSCKF 算法			改进 MSCKF 算法		
		x(m)	y(m)	闭环误差 (m)	x(m)	y(m)	闭环误差 (m)
1	168	-0.295	0.222	0.369	-0.242	0.172	0.297
2	198	-0.269	0.241	0.361	-0.208	0.182	0.276
3	236	-0.345	0.298	0.456	-0.286	0.243	0.375
4	262	-0.332	0.289	0.440	-0.274	0.237	0.362
5	298	-0.403	0.325	0.518	-0.311	0.296	0.429
平均闭环误差 (m)		0.429			0.348		

对比实验 1-实验 4 的结果可以看出, 实验 2 和实验 4 的运动时间较长, 但其定位精度优于实验 1 和实验 3, 说明在特征纹理丰富环境, 单目 Camera 在更新时可选取的有效特征点较多, 使得 MSCKF 算法定位效果优于特征贫乏环境. 由表 1 中可以看出, 改进 MSCKF 算法定位精度优于传统 MSCKF 算法, 平均闭环误差提升了 0.081 m, 定位精度提升, 说明改进 MSCKF 算法使用轮式里程计对平移运动的约束比加速度计好, 改进 MSCKF 算法的定位结果更优.

4 结束语

针对传统 MSCKF 算法实现机器人室内定位时加速度计传感器的固有缺点: 受重力干扰和需积分解算得到速度和位置方程的问题, 本文提出了改进 MSCKF

算法. 改进算法使用轮式里程计取代 IMU 中加速度计传感器, 替换传统 MSCKF 算法中 EKF 速度和位置状态方程, 使 EKF 预测更加准确. 同时将 MSCKF 及其改进算法封装在 ROS 内, 结合 Turtlebot2 机器人实现算法并进行实验验证. 通过对实验结果分析得到, MSCKF 算法及改进 MSCKF 算法在特征纹理丰富环境中的定位效果要优于特征贫乏环境, 且改进 MSCKF 算法的整体运动轨迹更接近真实轨迹, 闭环误差也优于传统的 MSCKF 算法, 这为后续室内机器人进行自主导航和路径规划提供了更好的前提.

参考文献

- 1 Durrant-Whyte H. Where am I? A tutorial on mobile vehicle localization. *Industrial Robot: An International Journal*, 1994, 21(2): 11–16. [doi: [10.1108/EUM000000004145](https://doi.org/10.1108/EUM000000004145)]
- 2 夏凌楠, 张波, 王营冠, 等. 基于惯性传感器和视觉里程计的机器人定位. *仪器仪表学报*, 2013, 34(1): 166–172. [doi: [10.3969/j.issn.0254-3087.2013.01.024](https://doi.org/10.3969/j.issn.0254-3087.2013.01.024)]
- 3 刘园园, 葛世荣, 朱华, 等. 煤矿救灾机器人定位技术研究. *煤矿机械*, 2011, 32(1): 49–52. [doi: [10.3969/j.issn.1003-0794.2011.01.021](https://doi.org/10.3969/j.issn.1003-0794.2011.01.021)]
- 4 Masamune K, Kurima I, Kuwana K, *et al.* HIFU positioning robot for less-invasive fetal treatment. *Procedia CIRP*, 2013, 5: 286–289. [doi: [10.1016/j.procir.2013.01.056](https://doi.org/10.1016/j.procir.2013.01.056)]
- 5 褚辉, 李长勇, 杨凯, 等. 多信息融合的物流机器人定位与导航算法的研究. *机械设计与制造*, 2019, (4): 240–243. [doi: [10.3969/j.issn.1001-3997.2019.04.059](https://doi.org/10.3969/j.issn.1001-3997.2019.04.059)]
- 6 文邹韬, 冯穗力. 基于 VIO 和 Wi-Fi 指纹技术的室内定位系统设计. *电讯技术*, 2019, 59(4): 449–454. [doi: [10.3969/j.issn.1001-893x.2019.04.014](https://doi.org/10.3969/j.issn.1001-893x.2019.04.014)]
- 7 徐则中, 庄燕滨. 移动机器人定位方法对比研究. *系统仿真学报*, 2009, 21(7): 1891–1896.
- 8 陈小宁, 黄玉清, 杨佳. 多传感器信息融合在移动机器人定位中的应用. *传感器与微系统*, 2008, 27(6): 110–113. [doi: [10.3969/j.issn.1000-9787.2008.06.035](https://doi.org/10.3969/j.issn.1000-9787.2008.06.035)]
- 9 Yuan K, Wang H, Zhang H. Robot position realization based on multi-sensor information fusion algorithm. *Fourth international symposium on computational intelligence and design*. Hangzhou, China. 2011. 294–297. [doi: [10.1109/ISCID.2011.81](https://doi.org/10.1109/ISCID.2011.81)]
- 10 Marín L, Vallés M, Soriano Á, *et al.* Multi sensor fusion framework for indoor-outdoor localization of limited resource mobile robots. *Sensors*, 2013, 13(10): 14133–14160. [doi: [10.3390/s131014133](https://doi.org/10.3390/s131014133)]
- 11 何壮壮, 丁德锐, 王永雄. 基于多传感器融合的移动机器人定位. *计算机与数字工程*, 2019, 47(2): 325–329, 343. [doi: [10.3969/j.issn.1672-9722.2019.02.014](https://doi.org/10.3969/j.issn.1672-9722.2019.02.014)]
- 12 Corke P, Lobo J, Dias J. An introduction to inertial and visual sensing. *The International Journal of Robotics Research*, 2007, 26(6): 519–535. [doi: [10.1177/0278364907079279](https://doi.org/10.1177/0278364907079279)]
- 13 Mourikis AI, Roumeliotis SI. A multi-state constraint kalman filter for vision-aided inertial navigation. *Proceedings 2007 IEEE International Conference on Robotics and Automation*. Roma, Italy. 2007. 3565–3572. [doi: [10.1109/ROBOT.2007.364024](https://doi.org/10.1109/ROBOT.2007.364024)]
- 14 Groves PD. GNSS 与惯性及多传感器组合导航系统原理. 练军想, 曹聚亮, 吴文启, 等, 译. 北京: 国防工业出版社, 2011. 66–68.
- 15 Li MY, Mourikis AI. Optimization-based estimator design for vision-aided inertial navigation. *Robotics: Science and Systems*. 2013. 241–248.
- 16 Sun K, Mohta K, Pfrommer B, *et al.* Robust stereo visual inertial odometry for fast autonomous flight. *IEEE Robotics and Automation Letters*, 2018, 3(2): 965–972. [doi: [10.1109/LRA.2018.2793349](https://doi.org/10.1109/LRA.2018.2793349)]
- 17 Ramezani M, Khoshelham K, Fraser C. Pose estimation by omnidirectional visual-inertial odometry. *Robotics and Autonomous Systems*, 2018, 105: 26–37. [doi: [10.1016/j.robot.2018.03.007](https://doi.org/10.1016/j.robot.2018.03.007)]
- 18 Feng W, Zheng L. Rapid and robust initialization for monocular visual inertial navigation within multi-state Kalman filter. *Chinese Journal of Aeronautics*, 2018, 31(1): 148–160. [doi: [10.1016/j.cja.2017.10.011](https://doi.org/10.1016/j.cja.2017.10.011)]
- 19 李东轩. 多传感器融合的室内移动机器人定位[硕士学位论文]. 杭州: 浙江大学, 2018.
- 20 Liu HHS, Pang GKH. Accelerometer for mobile robot positioning. *IEEE Transactions on Industry Applications*, 2001, 37(3): 812–819. [doi: [10.1109/28.924763](https://doi.org/10.1109/28.924763)]
- 21 文坤. 多状态多视图约束视觉/惯性组合导航算法研究[硕士学位论文]. 长沙: 国防科学技术大学, 2016.