同行专家业内评价意见书编号：<u>20250854407</u>

附件1

浙江工程师学院（浙江大学工程师学院）
同行专家业内评价意见书

姓名： <u>苏昱竹</u>

学号： <u>22260006</u>

申报工程师职称专业类别（领域）： <u>电子信息</u>

浙江工程师学院（浙江大学工程师学院）制

2025年03月20日

填表说明

一、本报告中相关的技术或数据如涉及知识产权保护、军工项目保密等内容，请作脱密处理。

二、请用宋体小四字号撰写本报告，可另行附页或增加页数，A4纸双面打印。

三、表中所涉及的签名都必须用蓝、黑色墨水笔，亲笔签名或签字章，不可以打印代替。

四、同行专家业内评价意见书编号由工程师学院填写，编号规则为：年份4位＋申报工程师职称专业类别(领域)4位+流水号3位，共11位。

一、个人申报

（一）基本情况【围绕《浙江工程师学院（浙江大学工程师学院）工程类专业学位研究生工程师职称评审参考指标》，结合该专业类别(领域)工程师职称评审相关标准，举例说明】

1.对本专业基础理论知识和专业技术知识掌握情况(不少于200字)

本人系统掌握了机器人定位与建图（SLAM）领域的理论知识，包括传感器原理、多传感器融合技术、优化算法及建图方法，并深入研究了隧道场景下的鲁棒建图与定位算法。在基础理论层面，熟练掌握基于滤波和图优化SLAM框架的数学模型和原理。同时，深入理解多传感器数据融合SLAM算法中的IMU（惯性测量单元）的预积分模型、激光雷达点云配准算法、视觉里程计的特征匹配与光流法等，并能够通过图优化（Graph Optimization）方法融合多源传感器观测信息，提升系统鲁棒性，能够结合实际场景需求设计针对性解决方案。

2.工程实践的经历(不少于200字)

本人在中控技术股份有限公司进行工程实践，该公司成立于1999年，是国内领先、全球化布局的智能制造整体解决方案供应商，并致力于“AI+数据”核心能力的构建及落地应用。在该工程实践项目中，主要以四足巡检机器人在隧道场景实现建图和定位为目标，结合激光雷达、相机、惯导器件、GNSS进行多传感器数据融合，并针对隧道场景的结构特点，实现四足机器人地图构建以及基于先验地图的全局定位。在这个过程中，极大地提升了我综合运用所学知识解决实际问题的能力，同时也培养了我的沟通协作能力，受益匪浅。

3.在实际工作中综合运用所学知识解决复杂工程问题的案例（不少于1000字）

在参与工程实践项目的实际工作中，针对由于隧道内结构单一，视觉纹理重复，且传统的卫星定位方法在天空视野有遮挡情况下定位精度差，常用的建图与定位算法在隧道场景中容易出现退化甚至失效的问题，结合激光雷达、相机、惯导器件、GNSS进行多传感器数据融合，并针对长隧道场景的结构特点，实现四足机器人环境感知建图以及基于先验地图的全局定位。首先设计了隧道场景的建图和定位系统，系统分别接收来自激光雷达、IMU以及相机传感器的数据，并且对高频的IMU数据进行预积分处理，并经过线性插值的处理方式对激光雷达由于运动畸变产生的误差进行校正。其次，针对隧道场景中由于动态物体干扰而造成的建图定位算法退化加剧的问题，提出了一种动态隧道场景下的地图构建与定位算法。在激光点云数据预处理部分加入了动态物体检测器，通过构建范围图像并基于遮挡原理识别并去除动态点云。之后，结合视觉2D目标检测，并利用相机投影关系，将图像中目标物的二维边界框转化为3D点云空间上的视锥。随后利用基于密度的点云聚类算法，快速得到隧道语义特征点云。结合隧道语义特征点云，线、面特征点云，进行帧-局部地图匹配，得到激光雷达的匹配误差函数，同时联合IMU预积分过程中的误差函数，构建同时包含激光雷达和IMU的联合误差函数，并进行迭代优化求解，实现前端的激光里程计部分。系统的后端根据激光里程计数据进行关键帧的选择，每一个关键帧都会作为因子图中的一个顶点，并将之前得到的里程计数据和基于GNSS计算得到的全局位置的相对变换添加到图中作为边，在机器人移动的过程中整个图也在不断的扩展，并进行全局优化，得到每个关键帧最终的精确位姿，并基于每个关键帧的位姿信息，构建全局的三维点云地图。最后基于全局先验地图，里程计，以及帧全局地图匹配结果，实现基于先验地图的实时定位。在该工作中，充分利用了我对于SLAM领域的知识，结合了我的多传感器融合技术、非线性优化算法设计经验以及SLAM算法在隧道等退化场景中面对的挑战的深刻理解，最终在实际的隧道环境中验证了该算法的有效性。这一研究成果的应用可以显著降低人工巡检成本，为隧道安全监测提供了智能化解决方案。整个过程中，我主导了算法设计、实现以及系统集成调试工作，

通过不断优化算法逻辑，最终实现了理论研究与工程实践的深度结合，在这个过程中，极大地提升了我综合运用所学知识解决实际问题的能力，也充分体现了我对于复杂工程问题解决的能力。

（二）取得的业绩（代表作）【限填3项，须提交证明原件（包括发表的论文、出版的著作、专利证书、获奖证书、科技项目立项文件或合同、企业证明等）供核实，并提供复印件一份】

1.

公开成果代表作【论文发表、专利成果、软件著作权、标准规范与行业工法制定、著作编写、科技成果获奖、学位论文等】

成果名称	成果类别 [含论文、授权专利（含发明专利申请）、软件著作权、标准、工法、著作、获奖、学位论文等]	发表时间/授权或申请时间等	刊物名称/专利授权或申请号等	本人排名/总人数	备注
一种长隧道环境下的实时定位和建图方法	发明专利申请	2024年03月25日	申请号：CN 202410344081.0	2/5	已进入实质性审查阶段
Lidar-inertial 3d slam with plane constraint for multi-story building	会议论文	2024年05月21日	2024 IEEE International Conference on Advanced Robotics and Its Social Impacts (ARSO)	2/7	EI会议收录

2.其他代表作【主持或参与的课题研究项目、科技成果应用转化推广、企业技术难题解决方案、自主研发设计的产品或样机、技术报告、设计图纸、软课题研究报告、可行性研究报告、规划设计方案、施工或调试报告、工程实验、技术培训教材、推动行业发展中发挥的作用及取得的经济社会效益等】

（三）在校期间课程、专业实践训练及学位论文相关情况	
课程成绩情况	按课程学分核算的平均成绩： 86 分
专业实践训练时间及考核情况（具有三年及以上工作经历的不作要求）	累计时间： 1.1 年（要求1年及以上） 考核成绩： 82 分
本人承诺	
个人声明：本人上述所填资料均为真实有效，如有虚假，愿承担一切责任，特此声明！ **申报人签名：** 苏昱行	

22260006

二、日常表现考核评价及申报材料审核公示结果

日常表现考核评价	非定向生由德育导师考核评价、定向生由所在工作单位考核评价： ☑优秀　　☐良好　　☐合格　　☐不合格 德育导师/定向生所在工作单位分管领导签字（公章）：徐钢　2025年3月21日
申报材料审核公示	根据评审条件，工程师学院已对申报人员进行材料审核（学位课程成绩、专业实践训练时间及考核、学位论文、代表作等情况），并将符合要求的申报材料在学院网站公示不少于5个工作日，具体公示结果如下： ☐通过　　☐不通过（具体原因：　　　　　　　　） 工程师学院教学管理办公室审核签字（公章）：　　　　年　月　日

浙 江 大 学 研 究 生 院

攻读硕士学位研究生成绩表

学号：22260006	姓名：苏昱竹	性别：女	学院：工程师学院	专业：电子信息	学制：2.5年
毕业时最低应获：26.0学分		已获得：28.0学分		入学年月：2022-09	毕业年月：
学位证书号：			毕业证书号：		授予学位：

学习时间	课程名称	备注	学分	成绩	课程性质	学习时间	课程名称	备注	学分	成绩	课程性质
2022-2023学年秋季学期	新时代中国特色社会主义理论与实践		2.0	92	专业学位课	2022-2023学年冬季学期	产业技术发展前沿		1.5	83	专业学位课
2022-2023学年秋季学期	工程技术创新前沿		1.5	85	专业学位课	2022-2023学年春季学期	研究生英语基础技能		1.0	81	公共学位课
2022-2023学年秋冬学期	工程伦理		2.0	82	专业学位课	2022-2023学年春季学期	自然辩证法概论		1.0	84	专业学位课
2022-2023学年秋冬学期	高阶工程认知实践		3.0	82	专业学位课	2022-2023学年春夏学期	人工智能制造技术		3.0	91	专业学位课
2022-2023学年秋冬学期	智能工业机器人及其应用		3.0	84	专业选修课	2022-2023学年春夏学期	标准与知识产权		2.0	89	跨专业课
2022-2023学年秋冬学期	数据分析的概率统计基础		3.0	91	专业选修课	2022-2023学年夏季学期	研究生英语		2.0	84	专业学位课
2022-2023学年秋冬学期	研究生论文写作指导		1.0	87	专业选修课		硕士生读书报告		2.0	通过	

说明：1.研究生课程按三种方法计分：百分制，两级制（通过、不通过），五级制（优、良、中、及格、不及格）。

2.备注中“*”表示重修课程。

学院成绩校核章：

成绩校核人：张梦依

打印日期：2025-03-20

(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 118274815 A
(43)申请公布日 2024.07.02

(21)申请号 202410344081.0

(22)申请日 2024.03.25

(71)申请人 浙江大学
地址 310058 浙江省杭州市西湖区余杭塘路866号

(72)发明人 朱秋国 苏昱竹 张家齐 胡鄢伟 吴俊

(74)专利代理机构 杭州求是专利事务所有限公司 33200
专利代理师 邱启旺

(51)Int.Cl.
G01C 21/00(2006.01)
G01C 21/16(2006.01)
G01S 17/86(2020.01)
G01S 17/89(2020.01)
G01S 19/42(2010.01)

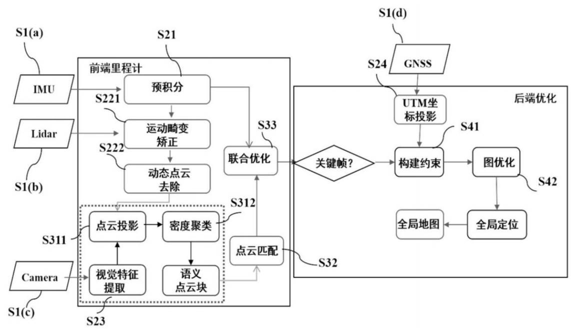
权利要求书2页 说明书9页 附图4页

(54)发明名称

一种长隧道环境下的实时定位和建图方法

(57)摘要

本发明公开了一种长隧道环境下的实时定位和建图方法,通过获取惯性测量信息、长隧道环境下的原始点云信息、长隧道顶部的图像信息、卫星定位信息;并对获取的信息进行预处理;对预处理后的点云信息和图片信息进行点云特征提取,得到隧道语义特征点云,同时结合线面特征点云进行点云配准,构建帧-局部地图的残差,并结合IMU预积分处理的结果,构建统一的代价函数,优化得到前端里程计;再将里程计结合隧道前后的GNSS数据,在后端中通过全局因子图优化的方式进行全局优化,得到每个关键帧最终的精确位姿,并基于每个关键帧的位姿信息,构建全局的三维点云地图。本发明在长隧道场景中有较高的鲁棒性,能够在长隧道场景下保持较高的定位和建图精度。



2024 IEEE International Conference on
Advanced Robotics and Its Social Impacts (ARSO)
Hong Kong, China, May 20-22, 2024

LiDAR-Inertial 3D SLAM with Plane Constraint for Multi-story Building

Jiashi Zhang[1*], Yuzhu Su[2*], Chengyang Zhang[1], Jianxiang Jin[1,3], Jun Wu[1,3], Rong Xiong[1,3], Qiuguo Zhu[1,2,3†]

2024 IEEE International Conference on Advanced Robotics and Its Social Impacts (ARSO) | 979-8-3503-4463-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ARSO60199.2024.10557752

Abstract— **The ubiquitous planes and structural consistency are the most apparent features of indoor multi-story buildings compared with outdoor environments. In this paper, we propose a tightly coupled LiDAR-Inertial 3D SLAM framework with plane features for the multi-story building. The framework we proposed is mainly composed of three parts: tightly coupled LiDAR-Inertial odometry, extraction of structural representative planes, and factor graph optimization. By building a local map and inertial measurement unit (IMU) pre-integration, we get LiDAR scan-to-local-map matching and IMU measurements, respectively. Minimize the joint cost function to obtain the LiDAR-Inertial odometry information. Once a new keyframe is added to the graph, all the planes of this keyframe that can represent structural features are extracted to find the constraint between different poses and stories. A keyframe-based factor graph is conducted with the constraint of planes, and LiDAR-Inertial odometry for keyframe poses refinement. The experimental results show that our algorithm has outstanding performance in accuracy compared with the state-of-the-art algorithms.**

I. INTRODUCTION

With the development of environmental perception capabilities, the scenarios can be explored are expanding from 2D to 3D by drone and legged robot. Accurate state estimation and mapping are the basic premises for applying robots toward to the real world. Facing indoor environments, especially multi-story buildings, the robot must obtain a globally consistent pose estimation on different floors. Otherwise the point clouds of different floors will overlap or be deflected, which cannot be used for autonomous navigation of robots. How to make the robot obtain globally consistent pose estimation on different floors is the focus and difficulty of SLAM in multi-story buildings.

A 3D LiDAR based on scanning mechanism has the advantages of textureless, invariant to the illumination, and broad horizontal of view (FOV) of 360°, which is generally used in indoor environments[1], [2]. Under normal circumstances, LiDAR-aided SLAM mainly uses extracting corner points and surf points method [3], [4], Normal Distributions Transform (NDT)[5] scan matching, or floor extraction[6] methods to achieve SLAM for a single floor. Although many algorithms implement SLAM by extracting planes in indoor environments, most only use plane constraints in the odometry part and achieve accurate SLAM algorithms by finding the scan-to-scan plane correspondence. These algorithms can achieve good results in scenes with a single indoor floor or a relatively small number of floors. However, when the robot explores from bottom to top in a multi-story building, the existing algorithms cannot achieve accurate state estimation on the robot's 6-DOF, due to long-distance and loop closure does not work. In the multi-story SLAM, due to the consistency of structure between different floors, some planes on different floors can represent the same building structure. When the robot observes the same plane on different stories, it can correct the current pose. Here we call these planes structural representative planes (SRP). Fig. 1 shows an example of SRP, where the same SRP is displayed in the same color on different stories. Finding the correspondence between SRP within the scope is the key to achieving low-drift SLAM in multi-story scenes.

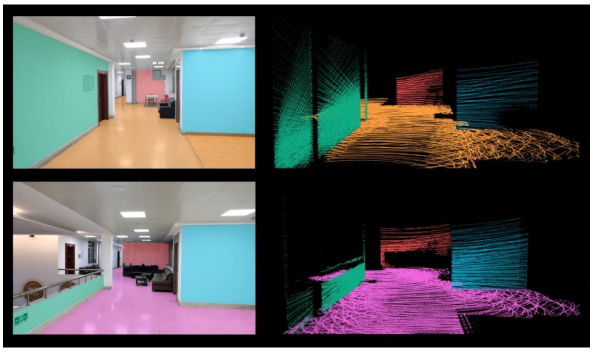


Fig. 1. Schematic diagram of SRP. On the left are actual scenes from different stories. Different colors represent different SRPs, and the same color represents the same plane. On the right is the SRP extracted from the LiDAR point cloud. Using the same SRP to construct constraints on different stories can eliminate accumulated errors.

This paper uses SRPs to build global constraints in different stories. Our framework has three parts: (1) tightly coupled LiDAR-Inertial odometry, (2) extraction of representative planes of the structure, and (3) factor graph optimization. The odometry is obtained by jointly optimizing the relative pose of the scan-to-local-map and the IMU pre-integration measurements. According to the odometry information, all

*This work was supported by the National Key R&D Program of China (Grant No. 2022YFB4701502), the "Leading Goose"R&D Program of Zhejiang(Grant No. 2023C01177), the Key Research Project of Zhejiang Lab (Grant No. 2021NB0AL03), and the Key R&D Project on Agriculture and Social Development in Hangzhou City(Asian Games) (Grant No. 20230701A05).

*Jiashi Zhang and Yuzhu Su are co-first authors of the article, Qiuguo Zhu is the corresponding author (e-mail: qgzhu@zju.edu.cn).
[1] College of Control Science and Engineering, Zhejiang University, Hangzhou, China.
[2] Polytechnic Institute, Zhejiang University, Hangzhou, China.
[3] State Key Laboratory of Industrial Control and Technology, Zhejiang University, Hangzhou, China.

the SRP will be extracted as candidates for the global plane constraint once a new keyframe is selected. Transform the global SRP to the keyframe coordinate system, and construct the global constraint relationship between keyframes according to the direction of planes' normals and the distance to the coordinate origin. Add odometry information and constraint information from planes to the factor graph, perform global optimization, and get the accurate pose of each keyframe.

The main contributions of this paper is summarized as follows:

- We propose the method of finding and constructing the global constraints of SRP in the multi-story blocks to achieve accurate 6-DOF state estimation of the robot when the loop closure is not possible.
- We propose a tightly coupled LiDAR-Inertial, keyframe-based SLAM framework to get the dense 3D point cloud maps of multi-story blocks.
- We validate the algorithm using the data collected from Velodyne VLP-16 and Xsens Mti-300 mounted on a real quadruped robot (Jueying Robot). Compared with the method without SRP, better results are obtained.

II. RELATED WORK

LiDAR Inertial odometry 3D LiDAR and IMU have been widely used in SLAM, both indoors and outdoors. The fusion methods of LiDAR and IMU are mainly divided into two categories: loosely coupled and tightly coupled. In the field of loosely coupled, LOAM [3] is a classic loosely coupled framework. It uses the orientation calculated by the IMU de-skew the point cloud and as prior information in the optimization process. The same method is also applied to its variants LeGO-LOAM [7]. LIO-Mapping [8] implemented LiDAR-Inertial tightly coupled algorithm by optimizing the cost function that includes both LiDAR and inertial measurements. However, the optimization process is carried out in a sliding window, so the time-consuming calculations make it impossible to maintain real-time performance. In their follow-up work, R-LINS [9], they use iterated-ESKF for the first time to achieve LiDAR-Inertial tightly coupled fusion and propose an iterated Kalman filter [10] to reduce wrong matchings in each iteration. A tightly coupled framework based on iterated Kalman filter is presented in [11], similar to R-LINS. An incremental kd-tree data structure is adopted to ensure cumulative updates and dynamic balance to ensure fast and robust LiDAR mapping. LIO-SAM [4] proposed by Shan T optimizes the measurements of LiDAR and IMU by factor graph, and at the same time, estimates the bias of the IMU.

SLAM related to plane features Whether in vision-based SLAM or LiDAR-based SLAM, plane-related features are widely used to improve state estimation accuracy. In LiDAR-based SLAM, LOAM [3] proposed extracting feature points from planar surface patches and sharp edges based on curvature calculation and improved the iterative closest point (ICP) [12] method based on the extracted feature points demonstrating the superb LiDAR odometry effect. Koide K [6] realize SLAM in a large-scale environment by detecting the ground, assuming that the indoor environment is a single flat floor. But this assumption is not applicable in all scenes and can only limit the height on the z-axis. LIPS [13] extract the plane in the three-axis direction of the point cloud, not only the ground plane, and combine the plane and IMU measurements in a graph-based framework. At the same time, the closets point (CP) is used to represent the plane to solve the singularity. π-LSAM, an indoor environment SLAM system using planes as landmarks, is proposed by Zhou L [14]. They adopt plane adjustment (PA) as the back-end to optimize plane parameters and poses of keyframes, similar to bundle adjustment (BA) in visual SLAM. Their subsequent work [15] extended this by using first-order Taylor expansion to replace the Levenberg Marquardt (LM) [16] method. To achieve faster computational speed, they define the integrated cost matrix (ICM) for each plane and achieve outstanding SLAM effects in a single-layer indoor environment. All of the above frameworks use a single LiDAR or a loosely coupled method of LiDAR and IMU as the front-end. On the contrary, we use a tightly coupled LiDAR-Inertial method as the front-end, which can obtain a more accurate prior pose of the keyframe, making it more precise when looking for the corresponding between the planes.

III. LIDAR-INERTIAL ODOMETRY

The Lidar-Inertial odometry, which is adapted from [8], maintains two sliding windows for building local map and optimizing states. Although it cannot run in real time, it can calculate an accurate pose transformation between two keyframes.

A. IMU Pre-integration

The LiDAR and IMU reference frames at time t are noted L_t and I_t, respectively. The state $\mathbf{X}_{I_t}^W$ of IMU to be estimated in the world frame W and the extrinsic matrix $\mathbf{T}_I^L$ from IMU to LiDAR are written as:

$$
\begin{aligned}
\mathbf{X}_{I_t}^W &= \begin{bmatrix} \mathbf{p}_{I_t}^{W\,T} & \mathbf{v}_{I_t}^{W\,T} & \mathbf{q}_{I_t}^{W\,T} & \mathbf{b}_{a_t}{}^T & \mathbf{b}_{g_t}{}^T \end{bmatrix}^T \\
\mathbf{T}_I^L &= \begin{bmatrix} \mathbf{p}_I^{L\,T} & \mathbf{q}_I^{L\,T} \end{bmatrix}^T
\end{aligned} \tag{1}
$$

where $\mathbf{p}_{I_t}^W$, $\mathbf{v}_{I_t}^W$, and $\mathbf{q}_{I_t}^W$ are the position, velocity, and orientation of IMU in the world frame W at time t. $\mathbf{b}_{a_t}$ and $\mathbf{b}_{g_t}$ are the bias of accelerometer and gyroscope of IMU.

Let t_i and t_j be the starting time and ending time of a raw LiDAR scan $\tilde{\mathscr{S}}_i$, respectively, so the pre-integration measurements $\Delta\mathbf{p}_{ij}$, $\Delta\mathbf{v}_{ij}$, $\Delta\mathbf{q}_{ij}$ of IMU from time t_i to t_j are computed as:

$$
\begin{aligned}
\Delta\mathbf{p}_{ij} &= \sum_{k=i}^{j-1} \left[\Delta\mathbf{v}_{ik}\Delta t + \frac{1}{2}\Delta\mathbf{R}_{ik}\left(\hat{\mathbf{a}}_k - \mathbf{b}_{a_k} - \mathbf{n}_a\right)\Delta t^2 \right] \\
\Delta\mathbf{v}_{ij} &= \sum_{k=i}^{j-1} \Delta\mathbf{R}_{ik}\left(\hat{\mathbf{a}}_k - \mathbf{b}_{a_k} - \mathbf{n}_a\right)\Delta t \\
\Delta\mathbf{q}_{ij} &= \prod_{k=i}^{j-1} \delta\mathbf{q}_k = \prod_{k=i}^{j-1} \begin{bmatrix} \frac{1}{2}\Delta t\left(\hat{\omega}_k - \mathbf{b}_{w_k} - \mathbf{n}_w\right) \\ 1 \end{bmatrix}
\end{aligned} \tag{2}
$$

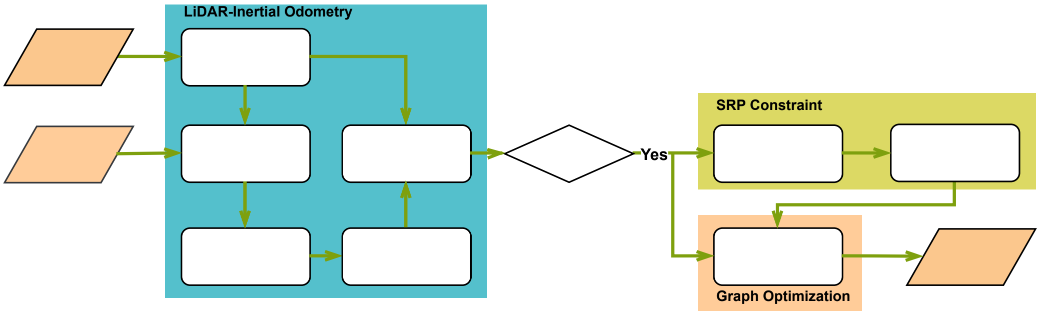


Fig. 2. System overview of our algorithm.

Readers can refer to [17] for the detailed derivation of Eq. (2).

B. Scan Deskewing and Feature Extraction

Due to the relative movement between the laser and the robot, there will be motion distortion for the raw LiDAR output $\tilde{\mathscr{S}}_i$, where $\tilde{\mathscr{S}}_i$ represents the point cloud starting from time t_i to time t_j. Every point $\mathbf{x}(t) \in \tilde{\mathscr{S}}_i$ is transformed to the correct position by linear interpolation to $\mathbf{T}_{ij}^{L}$ according to its timestamp, where $t \in [t_i, t_j)$. $\mathbf{T}_{ij}^{L}$ is obtained by IMU pre-integration and extrinsic matrix $\mathbf{T}_I^L$, and the undistorted scan is represented by $\mathscr{S}_i$.

To improve the efficiency of calculation, only the feature points that can reflect the characteristics of the surrounding environment are selected to find the relative pose of the LiDAR. Here we use the method of extracting feature points located on sharp edges and planar surfaces proposed by LOAM. The extracted edge and planar feature points from $\mathscr{S}_i$ are denoted as $\mathscr{F}_e^{L_i}$ and $\mathscr{F}_p^{L_i}$, respectively.

C. LiDAR Relative Measurements

When the new feature points $\mathscr{F}_e^{L_i}$ and $\mathscr{F}_p^{L_i}$ are extracted, the measurements of LiDAR need to be found to jointly perform the optimization with IMU.

1) Building Local Map: Since the points of a single scan are not dense enough, to obtain more accurate LiDAR measurements, we use a sliding window to construct a local map. The sliding window contains n LiDAR frames from time t_{i-1} to time t_{i-n}. Since we have extracted planar points and edge points separately, we transform $\left\{\mathscr{F}_e^{L_{i-n}}, \ldots, \mathscr{F}_e^{L_{i-2}}, \mathscr{F}_e^{L_{i-1}}\right\}$ and $\left\{\mathscr{F}_p^{L_{i-n}}, \ldots, \mathscr{F}_p^{L_{i-2}}, \mathscr{F}_p^{L_{i-1}}\right\}$ to frame L_{i-1} respectively with $\left\{\mathbf{T}_{i-n}^{i-1}, \ldots, \mathbf{T}_{i-2}^{i-1}, \mathbf{T}_{i-1}^{i-1}\right\}$ to obtain two feature local maps, $\mathscr{M}_e^{L_{i-1}}$ and $\mathscr{M}_p^{L_{i-1}}$.

2) Scan Matching: The relationship between the feature points and the local maps at time t_i are calculated by the point-line and the point-plane distances. First, transform the feature points $\mathscr{F}_e^{L_i}$ and $\mathscr{F}_p^{L_i}$ to frame L_{i-1}. The prediction transformation $\mathbf{T}_{i-1}^{i}$ used here is obtained through IMU pre-integration and extrinsic matrix $\mathbf{T}_I^L$. Here we take the plane points as an example. For each transformed plane point ${}'\mathbf{x}_p^{L_i}$, find the nearest m points in $\mathscr{M}_p^{L_{i-1}}$ to fit a plane in the frame L_{i-1} and express in Hesse normal form:

$$\mathbf{x}^T \mathbf{n}_p - d_p = 0 \tag{3}$$

where $\mathbf{n}_p$ is the unit normal vector of plane, and d_p is the distance from plane to the origin of frame L_{i-1}. So for each plane point $\mathbf{x}_p^{L_i} \in \mathscr{F}_p^{L_i}$, the residual is expressed as the point-plane distance:

$$\begin{aligned} \mathbf{T}_{L_i}^{L_{i-1}} &= \begin{bmatrix} \mathbf{R}_{L_i}^{L_{i-1}} & \mathbf{p}_{L_i}^{L_{i-1}} \\ \mathbf{0} & 1 \end{bmatrix} \\ r_{\mathscr{P}}(\mathbf{T}_{L_i}^{L_{i-1}}) &= \left(\mathbf{R}_{L_i}^{L_{i-1}} \mathbf{x}_{L_i}^{p} + \mathbf{p}_{L_i}^{L_{i-1}}\right)^T \mathbf{n}_p - d_p \end{aligned} \tag{4}$$

Similar to the calculation method of the plane point, the Hesse normal form can also describe the line in $\mathbb{R}^2$. For each edge point, the residual is represented as the point-line distance:

$$r_{\mathscr{E}}(\mathbf{T}_{L_i}^{L_{i-1}}) = \left(\mathbf{R}_{L_i}^{L_{i-1}} \mathbf{x}_{L_i}^{e} + \mathbf{p}_{L_i}^{L_{i-1}}\right)^T \mathbf{n}_l - d_l \tag{5}$$

D. Front-End Optimization

We build a cost function including IMU measurements and LiDAR measurements jointly and optimize all the states in the sliding window iteratively. For a sliding window of size n at time t_i, the states need to be optimized is $\mathbf{X}_i = \left[\mathbf{T}_i^{i-n}, \ldots, \mathbf{T}_{i-(n-1)}^{i-n}\right]$, and the final cost function is described as:

$$\begin{aligned} \min_{\mathbf{X}_i} \frac{1}{2} \Bigg\{ & \sum_{\alpha \in \{i-n, \ldots, i-1\}} \left\| r_{\mathscr{I}}(z_{\alpha+1}^{\alpha}, \mathbf{X}_i) \right\|_{\mathbf{C}_{I_{\alpha+1}}^{I_\alpha}}^2 \\ & + \sum_{\substack{\mathbf{x}_{L_i}^{p} \in \mathscr{F}_{L_i}^{p} \\ \beta \in \{i-n, \ldots, i-1\}}} \left\| r_{\mathscr{P}}(\mathbf{X}_i) \right\|_{{}^{p}\mathbf{C}_{L_{\beta+1}}^{L_{i-n}}}^2 \\ & + \sum_{\substack{\mathbf{x}_{L_i}^{e} \in \mathscr{F}_{L_i}^{e} \\ \gamma \in \{i-n, \ldots, i-1\}}} \left\| r_{\mathscr{E}}(\mathbf{X}_i) \right\|_{{}^{e}\mathbf{C}_{L_{\gamma+1}}^{L_{i-n}}}^2 \Bigg\} \end{aligned} \tag{6}$$

where $\|\mathbf{X}\|_{\mathbf{C}}^2 = \mathbf{X}^T \mathbf{C} \mathbf{X}$ and $r_{\mathscr{I}}(\mathbf{X}_i)$ is the residual of IMU measurements, which is defined in[8]. $r_{\mathscr{P}}(\mathbf{X}_i)$ and $r_{\mathscr{E}}(\mathbf{X}_i)$ are the residuals of planar points matching and edge points matching. $\mathbf{C}_{I_{\alpha+1}}^{I_\alpha}, \mathbf{C}_{L_{\beta+1}}^{L_{i-n}}, \mathbf{C}_{L_{\gamma+1}}^{L_{i-n}}$ represent the covariance matrix. This non-linear least squares problem is solved using the Levenberg–Marquardt algorithm[16].

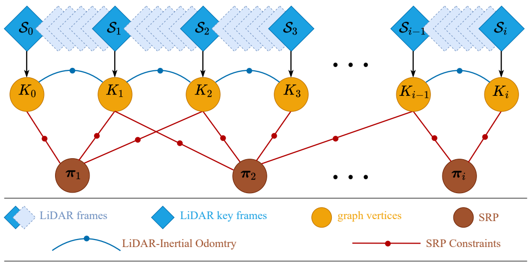


Fig. 3. The structure of the factor graph. The system selects keyframes based on the odometry as the vertices of the factor graph. The edges between the vertices are formed by LiDAR-Inertial odometry (blue curve) and SRP constraints (red line).

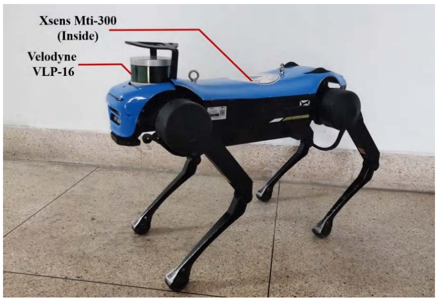


Fig. 4. The Jueying Mini quadruped robot equipped with Velodyne VLP-16 and Xsens Mti-300. The LiDAR is fixed on the head of Jueying, and the IMU is assembled at the center of mass.

IV. SRP CONSTRAINT AND GRAPH OPTIMIZATION

In this part, we extract keyframes based on the LiDAR-Inertial odometry and extract all SRP from the LiDAR scan in the keyframe coordinate system, find the correspondence in the entire graph and construct constraints as demonstrated in Fig. 3.

A. SRP Extraction

For the calculation efficiency, we select keyframes as vertices of the factor graph according to the odometry of the front-end. Since we are using a LiDAR based on scanning mechanism, the change of the yaw angle does not affect the selection of keyframes. The new keyframe will be selected only when the distance between the new frame and the previous keyframe exceeds 1m or the pitch angle or roll angle exceeds 10°.

We extract all SRP from the corrected LiDAR scan $\mathscr{S}_i$ for each newly added keyframe K_i. Here we define the plane as $\pi(\mathbf{n}, d)$ through the Hesse normal form described by Eq. (3). $\mathbf{n} = [n_x, n_y, n_z]^T$ represents the unit normal vector of the plane, and d represents the distance from the coordinate origin of K_i to the plane. Next, apply RANSAC[18] to extract planes for $\mathscr{S}_i$, but not all planes are reserved for building constraints, but only those planes that can represent the structure of the building (e.g., ground, walls, etc.) are selected. Here we adopt the following strategies for the extraction of SRP:

- Keep all the planes with more than $\mathscr{N}$ points (Here, we set $\mathscr{N}$ to 400).
- According to the normal vector of the extracted plane, three planes containing the most points and almost orthogonal are retained.
- Use 80% of the points in $\mathscr{S}_i$ to extract the plane, and the remaining points default to the unextractable points.

In a multi-story building, the walls between different floors are likely to be on the same plane in space, but small planes such as doors and cabinets are usually not associated between different floors. Therefore, we use the RANSAC algorithm to extract planes according to the number of inliers from large to small, and extract the most obvious planes first. The plane of the ground can be used to constrain the change of the Z-axis of the robot within the same floor and during stair climbing. If the first 80% cannot find three orthogonal SRPs, it is considered that there are no SRPs in the remaining 20%. We build constraints using already found SPRs (maybe 1 or 2). Too many planes are extracted will increase the uncertainty of the RANSAC process and cause mismatches in the plane matching process. Here we only use three orthogonal planes to obtain the precise pose of the LiDAR with 6-DOF. At the same time, fewer edges will be constructed in the factor graph to reduce the calculation time.

B. SRP Global Constraint

To construct the global constraint, all SRP extracted from keyframe K_i will be checked whether they have appeared in the previous keyframes. Here we denote all the planes added to the graph as $\Pi = \left\{\pi_1^{K_0}, \cdots, \pi_{k_0}^{K_0}, \cdots, \pi_1^{K_{i-1}}, \cdots, \pi_{k_{i-1}}^{K_{i-1}}\right\}$, and the SRP under the K_i frame as $\Pi^{K_i} = \left\{\pi_1^{K_i}, \cdots, \pi_{k_i}^{K_i}\right\}$. First, according to the optimized results $\mathbf{T}_{K_m}^{W}, m \in \{1, \cdots, i-1\}$ and the front-end odometry $\mathbf{T}_{K_{i-1}}^{K_i}$, the planes in Π are transformed to the frame of keyframe K_i.

$$\begin{aligned} \mathbf{T}_{K_m}^{K_i} &= \mathbf{T}_{K_{i-1}}^{K_i} \mathbf{T}_{W}^{K_{i-1}} \mathbf{T}_{K_m}^{W} = \begin{bmatrix} \mathbf{R}_{K_m}^{K_i} & \mathbf{p}_{K_m}^{K_i} \\ \mathbf{0} & 1 \end{bmatrix} \\ \begin{bmatrix} {}'\mathbf{n}^{K_i} \\ {}'d^{K_i} \end{bmatrix} &= \begin{bmatrix} \mathbf{R}_{K_m}^{K_i} & 0 \\ -\mathbf{p}_{K_m}^{K_i\,T} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{n}^{K_m} \\ d^{K_m} \end{bmatrix} \end{aligned} \tag{7}$$

For all $\pi_m^{K_i}(\mathbf{n}^{K_i}, d^{K_i}) \in \Pi^{K_i}, m \in \{1, \cdots, k_i\}$, calculate the angle $\delta\theta$ between its normal vector $\mathbf{n}^{K_i}$ and ${}'\mathbf{n}^{K_i}$ and the distance δd between d^{K_i} and ${}'d^{K_i}$. Once $\delta\theta$ and δd are lower than the preset threshold, add a plane edge to the factor graph. Otherwise, it's considered a new plane and added to Π.

C. Graph optimization

When the LiDAR-Inertial odometry and SRP construct the constraints between keyframes, the SLAM problem is expressed in a factor graph. The vertices of the graph represent states of being optimized, and the edges represent the constraints formed by the sensors' measurements, as

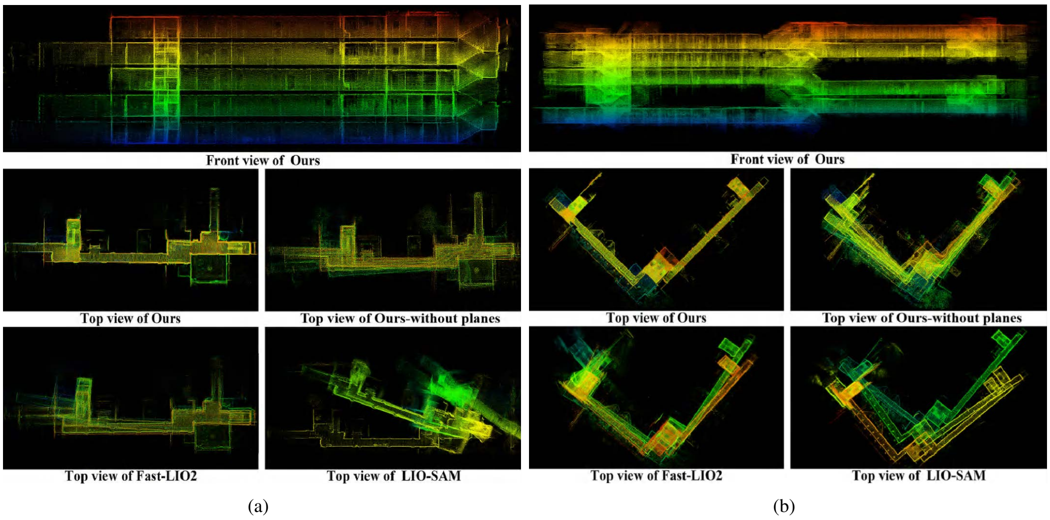


Fig. 5. Maps generated by Ours, Ours-without planes, Fast-LIO2, and LIO-SAM. Fast-LIO2, LIO-SAM and Ours-without planes drift on different stories. SRP constraints allow the robot to obtain accurate pose estimation at different stories. (a) Maps of Building A. (b) Maps of Building B.

shown in Fig. 3. Following[19], [20], the maximum likelihood estimation problem is expressed as this nonlinear least-squares problem:

$$F(\mathbf{x}) = \sum_{\langle i,j\rangle \in \mathscr{C}} \mathbf{e}\left(x_i, x_j, z_{ij}\right)^T \Omega_{ij} \mathbf{e}\left(x_i, x_j, z_{ij}\right) \tag{8}$$

where $\mathbf{x}$ represents all states to be optimized and $x_i, x_j \in \mathbf{x}$, z_{ij} and Ω_{ij} represent the mean and the information matrix of a constraint between x_i and x_j, $\mathscr{C}$ is the set of pairs of indices for which the constraint exist, and $\mathbf{e}\left(x_i, x_j, z_{ij}\right)$ is the error function between x_i, x_j and z_{ij}. Eq. (8) is minimized by Gauss-Newton or Levenberg-Marquardt algorithm.

V. EXPERIMENTS

A. Experimental Settings

To verify the versatility of the algorithm, we conduct experiments in different buildings. We use the Jueying Mini robot (Fig. 4) equipped with Velodyne VLP-16 and Xsens Mti-300 to collect data from multiple sets of multi-story scenes. The LiDAR is fixed on the head of Jueying, and the IMU is assembled at the center of mass. Since there are currently no publicly available datasets of LiDAR and IMU for indoor multi-story scenes, we used Jueying to collect actual data in two buildings and named them *Building A* and *Building B*, respectively. *Building A* is a five-story building in the shape of long corridor, and *Building B* is a six-story building with two long corridor-shaped scattered on the left and right. Our algorithm is tested on a PC with Intel Core i7-7567U, 16G memory.

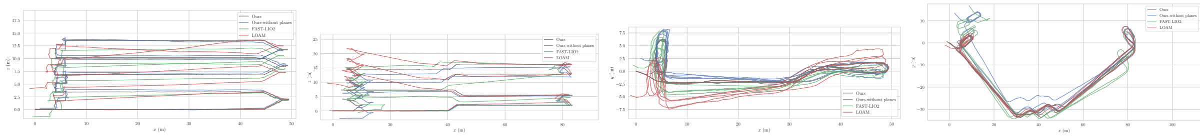
B. Results and Analysis

We compared the state-of-the-art SLAM algorithms based on multi-sensor fusion, including Fast-LIO2[11], LIO-SAM[4] and LOAM[3]. Due to the unique experimental scene, we cannot obtain the ground truth of the robot motion. At the same time, we set the robot's starting point and ending point to be the same when collecting data to calculate the relative position and orientation deviation.

Overview The performance of Ours, Fast-LIO2 and LIO-SAM on the *Building A* and *Building B* datasets are shown in Fig. 5. We can see that Ours with SRP constraint is better than the others on both datasets because of plane constraints. When the 16-line LiDAR moves horizontally, the height estimation will produce more significant deviations, especially in degraded scenarios such as corridors. Despite the aid of IMU, there will still be cumulative errors, which is seen more clearly in Fast-LIO2 and LIO-SAM. Ours-without SRP optimizes each state in the sliding window, which consumes more time, so the effect of height estimation is better, but in the end, it does not return to the starting point as well. The other two algorithms did not return to the starting point in the end due to the lack of performing loop closure. **Trajectory** Fig. 6 shows the trajectories of two datasets. LIO-SAM fails in *Building A* and *Building B*, so we did not plot its trajectory. Although we do not have global ground truth, we can see in Fig. 6(b) and Fig. 6(d) that in the staircase on the left of *Building A* and *Building B*, the other three algorithms drift a lot. Still, after adding plane constraints, ours can maintain the consistency of different floors. Table I provides the relative deviations of translation and rotation. Since accurate state estimation is achieved on other stories, our algorithm can return to the starting point without loop closure. Because of the same planes used to construct constraints, both translation and rotation are almost consistent with the starting point.

VI. CONCLUSION AND FUTURE WORK

This paper proposes a SLAM algorithm for indoor multi-story scenes with a plane as the main feature. To reduce the possibility of plane mis-matching, we use the tightly coupled LiDAR and IMU as the front-end. By plane matching and constraints building, the robot can eliminate the cumulative



(a) (b) (c) (d)

Fig. 6. Comparison of trajectories estimated by different algorithms (LIO-SAM fails on both datasets, so we don't plot its trajectory.). (a) Front view of trajectories in Building A. (b) Front view of trajectories in Building B. (c) Top View of trajectories in Building A. (d) Top View of trajectories in Building B.

TABLE I

THE ABSOLUTE VALUE OF RELATIVE DEVIATION OF DIFFERENT SLAM ALGORITHMS UNDER THE SAME STARTING POINT AND ENDING POINT.

Dataset	**Distance (m)**	**System**	**Translation (m)**				**Rotation (rad)**			
			ΔX	**ΔY**	**ΔZ**	**ΔXYZ**	**ΔYaw**	**ΔPitch**	**ΔRoll**	**ΔAngle**
Building A	396	Ours	**0.018**	**0.023**	**0.015**	**0.033**	**0.021**	**0.002**	**0.018**	**0.028**
		Ours-without planes	0.085	3.204	0.625	3.266	0.151	0.020	0.037	0.157
		LOAM	1.190	4.620	4.061	6.265	0.143	0.081	0.048	0.171
		LIO-SAM with loop cl	8.576	35.110	25.802	44.407	1.629	0.629	1.438	2.262
		FAST-LIO2	1.529	1.424	1.442	2.539	0.078	0.023	0.086	0.118
Building B	613	Ours	**0.021**	**0.023**	**0.002**	**0.031**	**0.006**	**0.007**	**0.006**	**0.011**
		Ours-without planes	0.571	13.491	3.139	13.863	0.214	0.048	0.051	0.225
		LOAM	1.109	1.606	9.999	10.188	0.293	0.010	0.011	0.293
		LIO-SAM with loop cl	4.296	12.792	12.109	18.131	2.409	0.030	0.180	2.416
		FAST-LIO2	1.529	1.424	1.442	2.539	0.078	0.023	0.086	0.118

error in different stories, and achieve an effect similar to "dimensionality reduction." Experiments show that our algorithm can significantly improve the state estimation and increase the accuracy of both localization and mapping. This improvement will boost robot performance in tasks such as indoor autonomous navigation and detection, which is extremely important for applications in indoor service and rescue robotics. However, the current plane matching process relies heavily on front-end odometry. In the future, it may be necessary to combine features unique to the plane to make the process more robust and fast.

REFERENCES

[1] J. A. Hesch, F. M. Mirzaei, G. L. Mariottini, and S. I. Roumeliotis, "A Laser-Aided Inertial Navigation System (L-INS) for human localization in unknown indoor environments," in *2010 IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 5376–5382.

[2] Q. Zou, Q. Sun, L. Chen, B. Nie, and Q. Li, "A Comparative Analysis of LiDAR SLAM-Based Indoor Navigation for Autonomous Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–15, 2021, early access.

[3] J. Zhang and S. Singh, "LOAM: Lidar Odometry and Mapping in Real-time." in *Robotics: Science and Systems*, vol. 2, no. 9, 2014.

[4] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 5135–5142.

[5] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3D-NDT," *Journal of Field Robotics*, vol. 24, no. 10, pp. 803–827, 2007.

[6] K. Koide, J. Miura, and E. Menegatti, "A portable three-dimensional LIDAR-based system for long-term and wide-area people behavior measurement," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, p. 1729881419841532, 2019.

[7] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4758–4765.

[8] H. Ye, Y. Chen, and M. Liu, "Tightly Coupled 3D Lidar Inertial Odometry and Mapping," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 3144–3150.

[9] C. Qin, H. Ye, C. E. Pranata, J. Han, S. Zhang, and M. Liu, "LINS: A Lidar-Inertial State Estimator for Robust and Efficient Navigation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 8899–8906.

[10] B. Bell and F. Cathey, "The iterated Kalman filter update as a Gauss-Newton method," *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 294–297, 1993.

[11] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast Direct LiDAR-Inertial Odometry," *IEEE Transactions on Robotics*, pp. 1–21, 2022, early access.

[12] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, 2001, pp. 145–152.

[13] P. Geneva, K. Eckenhoff, Y. Yang, and G. Huang, "LIPS: LiDAR-Inertial 3D Plane SLAM," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 123–130.

[14] L. Zhou, S. Wang, and M. Kaess, "π-LSAM: LiDAR Smoothing and Mapping With Planes," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5751–5757.

[15] L. Zhou, D. Koppel, and M. Kaess, "LiDAR SLAM With Plane Adjustment for Indoor Environment," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7073–7080, 2021.

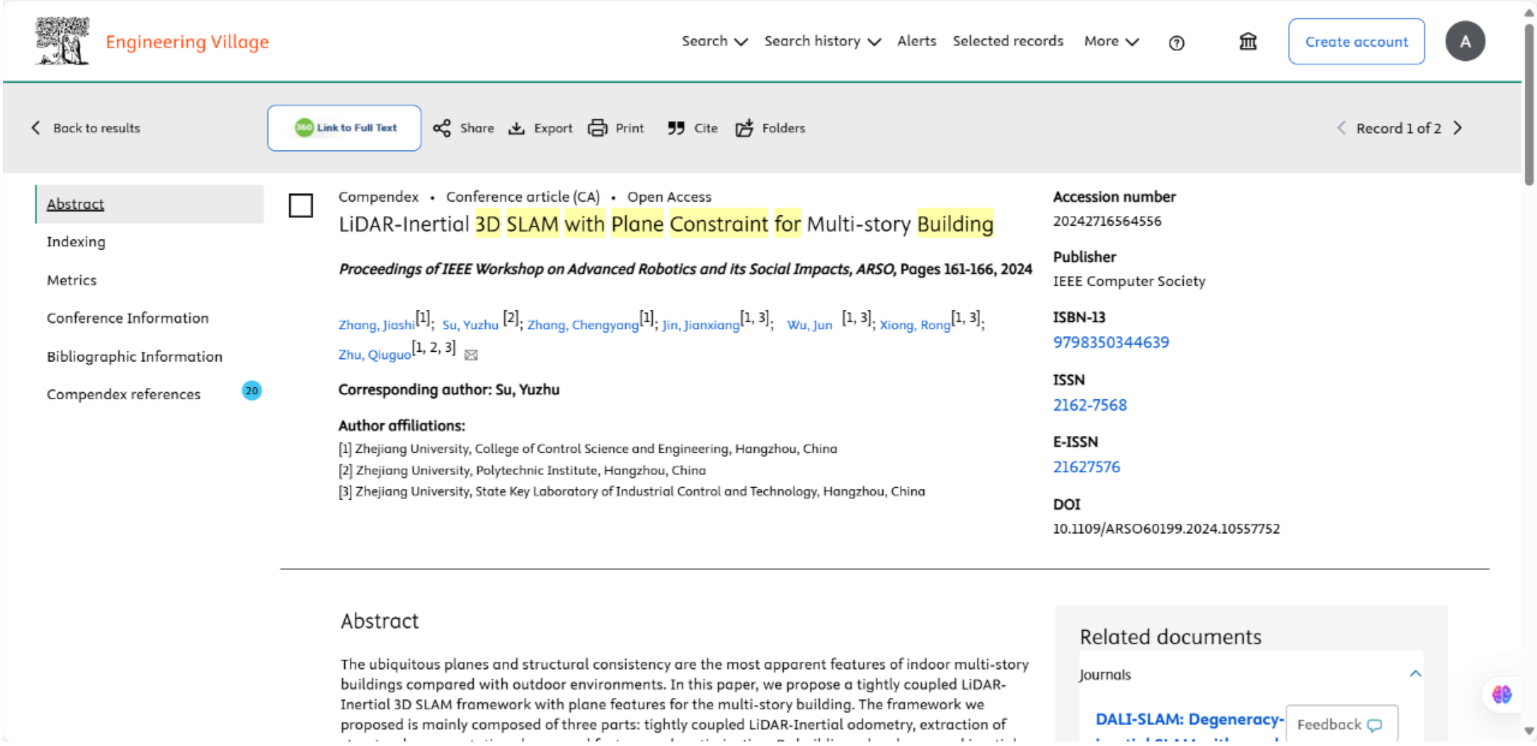
[16] J. J. Moré, "The Levenberg-Marquardt algorithm: implementation and theory," in *Numerical analysis*. Springer, 1978, pp. 105–116.

[17] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-Manifold Preintegration for Real-Time Visual–Inertial Odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2017.

[18] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[19] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G^2o: A general framework for graph optimization," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 3607–3613.

[20] G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard, "A Tutorial on Graph-Based SLAM," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, 2010.



Engineering Village

Search | Search history | Alerts | Selected records | More | Create account

Back to results | Link to Full Text | Share | Export | Print | Cite | Folders | Record 1 of 2

Abstract
Indexing
Metrics
Conference Information
Bibliographic Information
Compendex references 20

Compendex • Conference article (CA) • Open Access

LiDAR-Inertial 3D SLAM with Plane Constraint for Multi-story Building

Proceedings of IEEE Workshop on Advanced Robotics and its Social Impacts, ARSO, Pages 161-166, 2024

Zhang, Jiashi[1]; Su, Yuzhu[2]; Zhang, Chengyang[1]; Jin, Jianxiang[1, 3]; Wu, Jun[1, 3]; Xiong, Rong[1, 3]; Zhu, Qiuguo[1, 2, 3]

Corresponding author: Su, Yuzhu

Author affiliations:

[1] Zhejiang University, College of Control Science and Engineering, Hangzhou, China

[2] Zhejiang University, Polytechnic Institute, Hangzhou, China

[3] Zhejiang University, State Key Laboratory of Industrial Control and Technology, Hangzhou, China

Accession number
20242716564556

Publisher
IEEE Computer Society

ISBN-13
9798350344639

ISSN
2162-7568

E-ISSN
21627576

DOI
10.1109/ARSO60199.2024.10557752

Abstract

The ubiquitous planes and structural consistency are the most apparent features of indoor multi-story buildings compared with outdoor environments. In this paper, we propose a tightly coupled LiDAR-Inertial 3D SLAM framework with plane features for the multi-story building. The framework we proposed is mainly composed of three parts: tightly coupled LiDAR-Inertial odometry, extraction of

Related documents

Journals

DALI-SLAM: Degeneracy-

Feedback

链接地址：

https://www.engineeringvillage.com/app/doc/?docid=cpx_M5c9f106219091b85c5fM738110178165134&pageSize=25&index=1&searchId=ea3a3f5578334b6aaace72e07ea9cbb0&resultsCount=2&usageZone=resultslist&usageOrigin=searchresults&searchType=Quick