

分类号 _____

密 级 _____

U D C _____

编 号 10486 _____

武汉大学

硕 士 学 位 论 文

LiDAR/INS 组合的室内定位与制图
(SLAM) 算法改进

研 究 生 姓 名 : 余 彤

学 号 : 2014206180005

指导教师姓名、职称 : 牛小骥 教授
唐健 副教授

专 业 名 称 : 导航制导与控制

研 究 方 向 : 室内 SLAM 技术

二〇一七年五月

Algorithm Improvement of Indoor
Simultaneous Location and Mapping (SLAM)
Based on LiDAR/INS Integration

By

Yu, Tong

Supervised by

Prof. Niu, Xiaoji
Prof. Tang, Jian

Wuhan University
May, 2017

论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的研究成果。除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

学位论文作者（签名）：

年 月 日

摘要

近年来,随着面对公众的位置服务(Location-based Service, LBS)的市场不断扩大,亟需一种高效、高精度且成本低的室内定位与制图技术。由于单一导航定位技术自身的局限性,多传感器(LiDAR/惯导/视觉等)组合的室内同步定位与制图技术(simultaneous location and mapping, SLAM)成为了该领域研究的热点。本文搭建了基于 LiDAR/INS(Inertial Navigation System)组合的室内移动测图平台,分析了原有 LiDAR 匹配算法和组合导航算法的问题和不足,针对这些问题提出了一套改进 LiDAR/INS 组合定位定姿算法,并将原先的后处理工作改进为准实时处理解决方案,大大减小了结果输出的时间延迟。最后通过实验验证了本文提出的改进方案的精度、有效性和可行性。本文所做的主要工作和贡献为:

(1)详细阐述了多分辨率栅格地图的创建过程和 LiDAR 匹配算法,并比较分析了两种 LiDAR 匹配算法—IMLE 算法和高斯-牛顿迭代匹配算法的耗时。实验结果表明,IMLE 算法的平均耗时为 58.5ms,最长耗时为 88.2ms;高斯-牛顿算法的平均耗时为 10.9ms,最长为 14.4ms,远远小于 IMLE 算法的耗时。因此本文选用基于高斯-牛顿迭代法进行 LiDAR 匹配搜索,用以节省总制图时间,并为实时性改进工作作铺垫。

(2)虽然高斯-牛顿迭代算法能快速迭代出最优解,但是该算法很容易陷入局部最优中。针对这种现象,本文在总结了 LiDAR/INS 组合的室内 SLAM 相关理论和方法后,提出了一套改进的 LiDAR/INS 组合导航算法,使得组合后的结果能为高斯-牛顿算法提供一个更准确的先验初值。通过实验对比分析了 LiDAR/INS 组合的定位制图结果、仅靠 INS 姿态辅助的 LiDAR 定位制图结果以及单一 LiDAR 匹配的定位制图结果。对比结果表明,与高精度的参考地图相比较,本文所提出的 LiDAR/INS 组合导航算法的制图效果最好,精度可以达到 0.056m,考虑到人工选取特征点的操作误差,这个制图精度是可以接受的;仅靠 INS 姿态辅助的定姿结果在动态较大的情况下容易变得不准确,会导致所绘制的地图偏差越来越大;而仅靠 LiDAR 单独匹配时,即使在动态信息不太丰富的情况下也容易陷入局部最优中,所绘制的地图完全混乱。所以实验表明,LiDAR/INS 的组合算法能够有效地帮助高斯-牛顿迭代避免陷入局部最优中;而且动态信息

越丰富，INS 发挥的作用越大。

(3)不同传感器的不同采样率和不同算法的处理时间一直是实时 SLAM 算法中的难点。针对 LiDAR 匹配耗时远远大于 IMU 解算耗时而导致标准 Kalman 组合结果无法实时输出的问题，本文采用了一种分步完成观测更新再将误差状态及其协方差一步转移到当前时刻的 Kalman 滤波数学模型，改进了原先的后处理算法，与传统的后处理算法相比，实时性改进后的算法在保证制图精度的前提下，能够大大减少导航结果的输出延迟。

综上所述，本文的贡献可归结为：改进了原有的 LiDAR/INS 的组合算法，可以有效地避免 LiDAR 匹配算法陷入局部最优解中；并提出了一种实时性改进算法，使得整个系统变成实时定位与制图系统。此外，本文的工作对消费级的 MEMS 惯性器件应用于高精度的室内测图技术起到一定的推动作用。

关键词：室内 SLAM；移动测图平台；LiDAR/INS 组合导航；实时导航

ABSTRACT

The establishment of highly efficient, accurate and low-cost indoor mapping technology is becoming more and more necessary and urgent, due to the growing interest and market of indoor Location Based Services (LBSs). Multi-sensors (LiDAR/IMU/CAMERA) integrated indoor Simultaneous Location and Mapping (SLAM) technology becomes a promising solution, for no single navigation technology is robust enough to meet the requirements on its own. This paper builds a LiDAR/INS integrated based indoor mobile mapping platform. Based on the problems and shortcomings of the previous work, an improved LiDAR/INS integrated method is proposed, and the previous post-processed work was improved to a real-time solution in this paper. Finally, the methods are validated by a set of experiments. The main research work are summarized as follows:

(1) The multi-resolution occupancy grid map and LiDAR scan matching algorithm are described in detail, and the time-consuming of two kinds of LiDAR scan matching algorithm—IMLE algorithm and Guass-Newton algorithm are compared. The experiment result shows that the mean time-consuming of the former is 64.2ms, the maximum is 105.5ms; while the latter is 2.0ms and 9.5ms respectively. Therefore, the Guass-Newton is chose for LiDAR scan matching is due to the low time-consuming, which not only can help save the total mapping time, but also provide convenience for the real-time performance improvement.

(2) Although the positioning results can be achieved by using Guass-Newton scan matching in very short time, the shortcoming of this method is that it can easily get stuck in local minima if the initial searching value is not accurate. Aimed to the issue, this paper proposed an improved LiDAR/INS integrated algorithm to mitigate the problem. The positioning and mapping results of LiDAR/INS integrated method, only INS attitude based LiDAR scan matching and standalone LiDAR scan matching are compared and analyzed in experiments. The results shows that, compared with the high precision reference map, the mapping results of LiDAR/INS integrated method proposed in this paper was best. The RMS errors was about 0.0342m. Considering the

errors brought by manual operation, the accuracy is reasonable; In the rich dynamic conditions, only INS attitude based LiDAR positioning results became inaccurate, and the errors of mapping results became larger and larger; However, even in the unambiguous situation, the standalone LiDAR scan matching was also easy to get stuck in local optimum solution. Therefore, the LiDAR/INS integrated solution can help mitigate the problem, and the more abundant dynamic information, the better performance of INS.

(3) The real-time response issue of multi-sensors is a big challenge for a real-time SLAM system, due to the different sampling frequencies and processing time of different algorithms. In this paper, an online Extended Kalman Filter (EKF) integrated algorithm of LiDAR scan matching and IMU mechanization for indoor mobile navigation system is introduced. Since LiDAR scan matching is considerably more time consuming than the IMU mechanism, the real-time synchronous issue is solved via a one-step-error-state-transition method in EKF. When the mobile mapping system is wandering, mapping results can be shown on the computer in the meantime. Compared to the traditional sequential post-processed EKF algorithm, the proposed method can significantly mitigate the time-delay of navigation outputs under the premise of guaranteeing the positioning accuracy, which can be used as an online navigation solution for indoor mobile mapping

In summary, this paper focuses on improving the LiDAR/INS integrated navigation algorithm to help Gauss-Newton approach avoid get stuck in local minima; And improving the traditional post-processed method to a real-time solution. In addition, this research is also a push to use the MEMS inertial sensors in the indoor precision navigation and mapping.

Key words: indoor SLAM; mobile mapping platform; LiDAR/INS integrated; real-time navigation.

目录

摘要.....	I
ABSTRACT	III
1 绪论.....	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	2
1.3 论文的主要研究内容和章节安排.....	6
1.3.1 主要研究内容.....	6
1.3.2 章节安排.....	7
2 LiDAR 点云匹配算法与地图创建.....	8
2.1 引言.....	8
2.2 创建地图.....	9
2.2.1 常用地图简介.....	9
2.2.2 多分辨率栅格地图的构建.....	10
2.3 LiDAR 点云匹配算法研究.....	12
2.3.1 LiDAR 传感器数学模型.....	12
2.3.2 基于 IMLE 的 LiDAR 匹配算法.....	12
2.3.3 高斯-牛顿迭代算法.....	15
3 LiDAR/INS 组合定位定姿.....	20
3.1 引言.....	20
3.2 常用坐标系统.....	21
3.3 仅靠 INS 姿态辅助的算法.....	22
3.4 惯性导航机械编排算法.....	23
3.4.1 传感器误差补偿.....	23
3.4.2 速度更新.....	24
3.4.3 位置更新.....	25
3.4.4 姿态更新.....	25
3.5 LiDAR/INS 组合导航算法.....	27
3.5.1 INS 误差状态模型.....	27
3.5.2 基于 15 维状态向量的卡尔曼滤波算法.....	29
4 组合导航算法近实时性改进.....	33
4.1 引言.....	33
4.2 OSEST 算法详细过程.....	33
4.3 建立 OSEST 算法的数学模型.....	34
5 实测实验与分析.....	36
5.1 引言.....	36
5.2 室内移动测图的 UGV 平台设计.....	36
5.3 高斯-牛顿算法与 IMLE 算法的耗时性分析.....	39

5.4	INS 辅助对高斯-牛顿匹配的影响分析	41
5.6	近实时算法的实验验证	49
5.6.1	精度验证	47
5.6.2	实时性验证	49
6	总结与展望	53
6.1	论文工作总结	53
6.2	工作展望	54
	参考文献	55
	致谢	58

1 绪论

1.1 研究背景及意义

近年来,随着面对公众的位置服务的市场越来越大,如何研究出一种高效、精度高且消费低的室内制图技术变得极为重要,因此 SLAM 技术逐渐成为机器人领域的研究热点,SLAM 问题是指把机器人放在未知的环境中,机器人增量式地创建未知环境的连续地图,同时确定它在地图中的位置,它在机器人领域已经被研究了数十年(陈卫东等, 2005)。

在当今社会和经济发展中,室内地图的应用十分广泛,在军事、日常生活、交通、通信等领域都起到极其重要的作用,如室内救援、室内侦察、室内扫地机器人、全景看房、AR 等。但现阶段采集建筑物的数据并绘制室内地图的主要方法仍然是人工用激光卷尺等方法采集环境中特征点的数据,再用电脑软件如 Computer-aided design(CAD)等绘制出来。这种人工的方法不仅效率低下,在绘图的时候还容易加上人为的主观判断,不能十分准确地反映出整个空间的真实情况,并且在绘图完成后,再进行更改也是一项费时费力的工作。而 SLAM 技术不仅可以快速且准确地测量出环境中每个建筑物的大小、形状、复杂度等等,形成所需场景中的地图,还可以随着室内环境的改变实时更新地图,大大提高了所需的测量效率(Hess et al., 2016)。

随着测绘学、导航制导与控制学、遥感科学等的交叉融合,催生了以不同传感器组合定位制图的方式,来代替单一导航系统,多传感器组合导航系统取长补短,优势互补,能够比任意单一导航系统获得更高的导航性能。在室外环境下,最常用的技术是利用全球卫星导航系统(Global Navigation Satellite System, GNSS)和惯性导航系统(Inertial Navigation system)组合定位,当 GNSS 信号条件好时,可以提供厘米级的定位精度(Fotopoulos et al., 2001)。

然而在室内环境下,GNSS 信号高度衰减,以至于不能满足室内导航定位的需求。在室内 SLAM 中,所用的传感器一般分为两类:环境感知传感器和运动感知传感器(杨景阳, 2014)。环境感知传感器一般包括激光测距仪(Light Detection and Ranging, LiDAR)、视觉传感器、声呐、光流传感器等,主要根据周围环境特征来实现定位。运动感知传感器有 INS、里程计等,可以不依靠外界环境实现自

主导航。

LiDAR 以其定位精度高, 定位精度高, 不像视觉传感器受环境光照等影响, 比较稳定等优点, 广泛应用于室内 SLAM 中, 但由于 LiDAR 匹配高度依赖周围环境的特征, 在低特征的环境下表现很差, 例如长廊等; 而惯性导航虽然不依靠外部环境信息便可实现自主导航, 但其中包含的传感器误差会随着工作时间的变长而逐渐累积, 误差累积到一定程度后便会失效(李荣冰, 2012; 高钟毓, 2012)。因此, 将两种传感器有效结合, 取长补短, 高效率地绘制出高精度的室内地图变成了当前国内外学者研究的一个重要方向(Bailey, 2006)。

1.2 国内外研究现状

随着人工智能科学的迅速发展, 国内外的 SLAM 研究都取得了突破性的进展。2016 年 10 月 6 日, Google 公司将实时 SLAM 算法 Cartographer 开源, 这在 SLAM 领域是非常具有重要性的一个举动, Google 的这项技术能同时应用于二维三维的定位及制图(Hess et al., 2016), 该 SLAM 算法结合了来自 LiDAR、IMU 和多个摄像头的的数据, 主要侧重于 LiDAR SLAM。早在 2014 年 9 月, Google 公司就曾推出一款同名背包, 这款背包配备了两个多回波激光扫描仪和一个惯性测量组件, 用 SLAM 技术实时绘制室内平面图。还可以在室内地图上添加信息点, 比如标记酒店的房间号、博物馆的展览品等。Google 公司声称“只用几个小时就完成了 39 层的旧金山马奎斯万豪酒店室内地图的制作”。还有 Google 公司和特斯拉公司共同研发的基于 SLAM 的无人驾驶车(Czapla, 2013), 在车顶处配备 3D 激光雷达, 勇于实时构建周围的 3D 环境地图, 最大辐射范围 200m, 然后将采集的环境信息传输至车载中央处理系统, 主要用于避障。

国内的 SLAM 研究进两年发展的也十分迅猛, 有各大高校进行的理论和实验结合的研究, 如武汉大学的测绘遥感等研究单位开始利用 LiDAR 对车载导航系统进行辅助(李德仁等, 2006), 国防科技大学利用惯导和超声波等辅助 LiDAR 进行测图(章大勇等, 2010), 南京航空航天大学以微小型四旋翼导航控制为应用背景, 研究基于 LiDAR、视觉、光流的 SLAM 实现(王平等, 2013)。随着国内市场需求越来越大, 以思岚科技公司为代表的一批国内创业公司也在迅速崛起, 致力为消费者提供消费级产品领域的高性能机器人定位导航解决方案及相关核心传感器, 如激光雷达、机器人自主定位导航系统、通用型商用机器等。

现阶段,几乎所有的基于 LiDAR 的 SLAM 算法都有一个共同特点:基于概率估计。概率估计算法的一个优势在于它能稳定地测量出环境中的噪声,并且能够表示出在测量和估计过程中的不确定性。大部分的概率模型都会用到贝叶斯法则来解决制图问题(Thrun et al., 2005)。

卡尔曼滤波(Kalman Filters, KF)是现阶段最常用的贝叶斯估计之一,卡尔曼滤波分为两个阶段:预测和更新。预测阶段是从迭代的过程中估计出状态向量,而更新过程是将估计出的状态量与其他传感器得到的观测量进行融合,得到更准确的结果。由于它便于计算机编程实现,并能够对现场采集的数据进行实时的更新和处理,所以 Kalman 滤波是目前应用最为广泛的滤波方法之一,在通信,导航,制导与控制等多领域得到了较好的应用。但机器人的姿态模型并不总是线性的,于是 Sunahara 等人(2007)提出了扩展卡尔曼滤波(Extended Kalman Filter, EKF),并将它应用在线性系统中。Huang 等人(2007)也对基于 EKF 的 2D 非线性 SLAM 问题的收敛性和一致性作出了分析。

粒子滤波(Particle Filter, PF)是另一种贝叶斯估计。它是用粒子集来表达概率,其核心思想是通过从后验概率中抽取的随机状态例子来表达其分布,每一个粒子赋予一个权重。它假定下一时刻的状态仅仅只依靠当前的状态(Thrun, 2001)。粒子滤波技术的优势在于处理非线性、非高斯系统以及多模态的处理能力,这些决定了它的应用范围非常广泛。Montemerlo 等人(2002)将 Rao-Blackwellised 粒子滤波器应用到机器人 SLAM 中,并取名为 FastSLAM 算法,它是利用改进的粒子滤波来估计出后验概率。虽然粒子滤波算法可以作为解决 SLAM 问题的有效手段,但是该算法仍然存在着一些难点。粒子滤波需要用大量的样本数量才能很好地模拟系统的后验概率密度。机器人面临的环境越复杂,描述后验概率分布所需要的样本数量就越多,算法的复杂度就越高。因此,能够有效地减少样本数量的自适应采样策略是该算法的重点。另外,重采样阶段会造成样本有效性和多样性的损失,导致样本贫化现象。如何保持粒子的有效性和多样性,克服样本贫化,也是该算法的另一个研究重点。

基于图优化(Graph-based)的 SLAM 算法也是目前应用非常广泛的一种技术,因为它克服了卡尔曼滤波和粒子滤波技术的某些劣势(Thrun, Montemerlo, 2005)。图优化 SLAM 问题主要分为两个部分:(1)构建图,机器人位姿当作顶点,位姿间关系当作边,这一步常常被成为前端(front-end),往往是传感器信息的堆积;(2)

优化图,调整机器人位姿顶点尽量满足边的约束,这一步称为后端(back-end)。Lu 和 Milios (1997)首次提出了这类算法,他们在扫描匹配的过程中来对机器人的位姿提供约束,但是由于文章中使用的优化算法的限制条件,使得该算法不能适用于大型场所。Thrun 等人(2005)在 Lu 等人(1997)的工作基础之上提出的 GraphSLAM 算法能够在大范围的城市环境中估计出机器人的位姿。

最近几年,SLAM 算法层出不穷,本文将目前比较常用的四种开源的 2D SLAM 算法作一些简单介绍:

(1) Hector SLAM

Hector SLAM 将 LiDAR 的扫描匹配与惯性导航传感器(以下简称惯导)的 3D 导航融合形成一个 2D SLAM 系统(Kohlbrecher,2011),Kohlbrecher 等人(2011)利用 LiDAR 的高更新频率和低测量噪声特点,实时得到机器人的动态信息。它通过已经获得的地图对激光束点阵进行优化,估计激光点在地图的表示和占据栅格的概率。该方法基于高斯牛顿算法来解决 LiDAR 扫描匹配问题,它的核心思想是找到激光点云集映射到已知地图的最优刚体转换。除此之外,该方法为了避免陷入局部最优而非全局最优—高斯牛顿算法中常见的问题,而采用多分辨率地图。最后利用 EKF 将 LiDAR 扫描匹配的结果与惯导得到的导航信息进行融合。该方法的优点是无需使用里程计,所以使得空中无人机及地面小车在不平坦区域建图存在运用的可行性。但该方法的缺点是需要机器人速度控制在比较低的情况下,建图效果才会比较理想,因为该方法没有用到回环(Loop Closing)。

(2) Gmapping

Gmapping 是由 Grisetti 等人(2007)提出,它是目前激光 2D SLAM 中运用的最广的一种方法,采用的是 Rao-Blackwellized Particle Filter 算法。这种算法通常需要大量的粒子来获得最好的结果,但这势必会增加计算的复杂度。同时由于粒子滤波是一个依据过程的观测逐渐更新权重与收敛的过程,重采样的过程必然会代入粒子耗散问题(depletion problem),大权重粒子显著,小权重粒子会消失,这导致有可能正确的粒子模拟由于在中间的阶段表现权重小而消失。Grisetti 等人(2007)通过自适应重采样技术有效地减少了粒子耗散的问题,同时他们在计算粒子分布的时候,不仅仅只依靠机器人的运动,也将当前观测考虑进去,减少了机器人位置在粒子滤波步骤中的不确定性。它的优点是在长廊及低特征场景中建图

效果好，但缺点是依赖里程计 (odometry)，无法适用无人机及地面小车不平坦区域，也没有用到回环 (Loop Closing)，不能有效地消除逐渐积累起来的误差。

(3) Karto SLAM

Karto SLAM 是基于图优化的方法(Vincent,2010)，它利用图的均值表示地图，每个节点表示机器人轨迹的一个位置点和传感器测量数据集，箭头的指向的连接表示连续机器人位置点的运动，每个新节点加入，地图就会依据空间中的节点箭头的约束进行计算更新。如果地标越多，内存需要就越大。但是图优化方式通常仍然比其他方法在大环境下制图更有效率，尤其是 Karto SLAM，因为它的图中仅仅只包含机器人的位姿。

(4) Cartographer

Cartographer 是 Google 的实时室内建图项目(Hess etc.al, 2016)，它将传感器安装在背包上面，主要用的传感器有 LiDAR 和惯性导航仪(Inertial Measurement Unit, IMU)，可以生成分辨率为 5cm 的 2D 栅格地图。Cartographer 的设计目的是为了解决在有限的计算资源下，能够实时地获取相对较高精度的 2D 地图，考虑到基于模拟策略的粒子滤波方法在较大环境时对内存和计算资源的高需求，Cartographer 采用基于图优化方法，具体实现是每获得的每一帧 LiDAR 扫描数据，利用扫描匹配在最佳估计位置处插入子图 (submap) 中，且扫描匹配只跟当前子图有关。在生成一个子图后，会进行一次局部的回环 (Loop closing) 来消除误差，利用分支定位和预先计算的栅格，所有子图完成后，会进行全局的回环。

另外，目前，大多数已有的高精度 SLAM 系统都是后处理工作(Thrun,2008; Kaijaluoto,2015; Jung,2014,2015)，高精度的实时制图算法仍然是一个难题，在 LiDAR/INS 组合系统的实时应用中，由于 LiDAR 扫描匹配时间过长会带来观测数据时间延迟问题，导致当前时刻观测数据无法实时得到和处理，直接影响到 LiDAR/INS 组合系统的实时性能，这是实时性改进算法中的一个主要矛盾。目前，对于这种时间延迟的解决方法有采用增广状态的方法(王光辉, 2013, Kailath, 2000)，但增广状态方法会增加状态维度，使得计算量增大；还有基于新息重组理论建立最优滤波器的方法(卢晓, 2008, 李林林, 2012)，但新息重组的方法，需要建立多个观测方程，如果有多步延迟，则不利于对 Kalman 滤波器的灵活拓展与应用；Guivant et al.(2001)提出过一种针对实时性需求的能减小计算量的优化算法，但是对地标的要求较高。Google 的 Cartographer(Hess,2016)也是一种实时

地室内地图的解决方案,他们所用的 LiDAR 匹配耗时远远小于本文中的 LiDAR 匹配耗时。但是, Cartographer 需要实时进行回环修正积累的误差,并且为了加速计算速率,导致 CPU 的占用率非常高,几乎占满了电脑的 CPU。

综上所述,大部分的室内 SLAM 算法研究都运用到了多传感器的融合技术,例如 LiDAR/视觉/超声波/里程计/惯导等传感器的组合。但所利用的传感器越多,算法也就越复杂。本文将深入研究基于仅 LiDAR/INS 组合的高精度室内 SLAM 算法,发掘 INS 更大的优势,提出一套高效的 LiDAR/INS 组合导航算法。而且,本文除了对后处理的 LiDAR/INS 组合导航算法进行改进之外,还提出了一种基于状态转移的实时性解决办法,这种方法曾经在 GNSS/INS 组合导航系统中应用过(章红平,常乐等,2016),但还未在 LiDAR/INS 组合导航系统中运用。该方法不仅计算量小,易于实现,而且可靠性高。

1.3 论文的主要研究内容和章节安排

1.3.1 主要研究内容

本文针对室内未知环境中的高精度移动测图需求,对原先工作中 LiDAR/INS 组合导航算法做出如下改进,更充分发挥 INS 的优势:

- (1) 由于本文中所用的 LiDAR 匹配搜索算法的原理是通过 INS 得到的位置和姿态,为 LiDAR 匹配提供一个较为合理的搜索初值,再利用高斯-牛顿逼近法寻找最优解,无需设定搜索范围,提高了 LiDAR 解算效率,但这种算法对搜索初值的依赖性非常高,因此本文将原先工作中的仅依靠 INS 姿态对 LiDAR 提供搜索辅助的方式,改为将 INS 进行机械编排得到位置、姿态等信息后,再与 LiDAR 匹配得到的观测信息坐卡尔曼滤波组合更好地发挥 INS 的性能,提高其定位精度。
- (2) 改进原先的后处理算法,提出一种近实时性改进算法,在不降低制图精度的前提下,大大减少导航结果的输出延迟,为以后将算法移植到嵌入式平台上,能够使得在移动机器平台行走的同时,一边采集周围环境中的信息,一边在计算机中实时显示出所绘制的地图的实时制图作铺垫。

1.3.2 章节安排

论文总共分为五章，各章节安排如下：

第一章为绪论。主要阐述了本文的研究背景及意义，综述了 SLAM 的国内外研究现状和基于 LiDAR/INS 组合的室内同步定位与制图技术的现状，最后提出了本文的研究目标，确定了研究思路，介绍了论文的主要研究内容。

第二章介绍了栅格地图的构建，建立 LiDAR 传感器数学模型，介绍了 LiDAR 点云匹配的过程以及传统的 IMLE 算法和本文所用的高斯-牛顿迭代算法。

第三章介绍了组合导航技术的基本知识和改进前仅靠 INS 姿态辅助 LiDAR 匹配的过程，详细描述了改进后的组合导航算法，包括惯性导航的机械编排算法，建立了惯导误差传播模型，和 EKF 算法。

第四章详细介绍了如何将标准的组合导航算法改进为近实时组合导航算法，并建立了算法的数学模型。

第五章为实测实验与分析。本章对改进算法后的整个移动测图系统进行了全面的测试与评估。对改进后组合导航算法的制图精度以及算法的效率进行了评估，对比分析了原先后处理算法和本文提出的近实时算法的输出延迟，验证算法的可行性。

第六章为总结与展望。总结了本文的研究工作，并对后续的工作进行了展望。

2 LiDAR 点云匹配算法与地图创建

2.1 引言

从 SLAM 的定义中可以看出, SLAM 算法包含两个方面: 定位和制图, 这两者是相辅相成的。在本文所用的 LiDAR/INS 组合系统中, LiDAR 是 SLAM 中的测量核心传感器, 它不仅可以定位自身位置, 还可以绘制周围环境地图。

目前, 主要有两种策略来解决 LiDAR 的定位问题: 利用特征匹配的绝对定位和利用扫描匹配的相对定位。利用特征匹配的绝对定位方法是通过环境感知传感器从周围环境中提取线、拐角等特征, 然后将这些特征与预设的特征地图中的已知特征点匹配, 从而进行定位 (Miller, 2010; Aghamohammadi, 2007; Aghamohammadi, 2008)。而相对定位中的扫描匹配是利用连续的两帧或多帧扫描点云直接计算出自身位置, 一般最常用的算法有 ICP (Iterative Closed Point) 算法 (Besl, 1992)、ICL (Iterative Closed Line) 算法 (Bosse, 2004; Censi, 2008)、MLE (Maximum Likelihood Estimation) 算法 (Bachrach, 2011; Steux, 2010; Pfister, 2006)。扫描匹配方法是目前为止室内定位中最可靠的方法之一。但是当 LiDAR 采样频率不够高, 连续扫描帧的信息不够时, ICP 和 ICL 算法会使得航向估计迅速发散, 随着时间逐渐积累起来的位置误差将会使得定位精度大大降低。MLE 算法是一种基于栅格地图的全局地图搜索方式, 如果地图的分辨率越高, 那么得到的定位精度也将越高; 但是如果需要高精度的大范围室内 SLAM, 那么这种方法就会占用很大的内存来存储地图信息, 并且全局的搜索方式也会大大增加时间成本。尽管基于 MLE 算法也有学者提出了一些优化的解决方法来加快搜索速率, 例如 Monte Carlo (Steux, 2010), 但是这种方法容易陷入局部最优解, 从而造成定位结果不准确。

而本文所采用的 LiDAR 点云匹配方法在没有降低原有算法定位结果的精度下, 大大加快了搜索匹配的速率。本章 2.2 节详细介绍了本文如何构建 SLAM 地图, 并分析了这种地图的优势。2.3 节根据 LiDAR 传感器数据的特点, 对 LiDAR 点云匹配算法进行了深入研究, 详细介绍了本文所运用的高斯-牛顿匹配算法的基本原理, 并对比分析了该算法的优势。

2.2 创建地图

2.2.1 常用地图简介

地图是同步定位与制图问题中的基础,机器人的定位和路径规划等都离不开地图。环境传感器可以通过感知现实环境中的信息创建、更新环境地图。目前研究领域中所用的典型的地图主要有三种,分别是:栅格地图,拓扑地图和几何特征地图。

(1) 栅格地图

栅格地图最早是由 Moravec(1985)、Elfes(1987,1989)等人提出,它是指把环境划分成一系列的栅格,其中每一个栅格被赋予一个(0,1)之间的概率值,表示这个栅格被占的概率,在地图上显示为该栅格的灰度,以此来表示这个栅格对应的实际环境中的位置是否有障碍物存在。栅格地图以其易于创建和维护的优点而被广泛应用,每个栅格中可以存储对应环境中的对应区域信息,并且容易被及时更新,特别适合处理激光或者超声测量数据。但它的缺点是所占用的内存和处理时间会随着地图规模的扩大而大大增加,所以对于大型环境或者精度要求非常高的情况下,所需的栅格数量将大大增加,运算复杂度和空间复杂度也会大大增加。

(2) 拓扑地图

拓扑地图最早是由 Brooks, Mataric 等人提出,它是指把所检测到环境表示为一张带有节点和相关连接线的拓扑结构图,其中的节点对应于环境中的特征点,连接线表示不同节点之间的连接关系(徐则中, 2004)。拓扑地图的方法很容易绘制出结构化环境的地图,它的优点是降低了空间复杂度,但同时由于节点表示的相似性,导致在构建大的非结构化的环境地图时,无法识别某个特征点。

(3) 几何特征地图

几何特征地图是指提取环境信息中的几何特征(如点、线、面)用于位置估计和目标识别(Chong, 1999; Leonard, 1992)。比如用线段来表示室内的墙面,用点来拟合墙角等。为了保证地图的一致性,几何特征地图需要各观测信息的位置是相对准确的。但在很多比较复杂的环境中,难以用可参数化的几何特征来表示环境特征,所以实现精度较高的地图是该方法的一个难点。

2.2.2 多分辨率栅格地图的构建

本文采用栅格地图来描述室内环境(Tang, 2014)。在 2.2.1 节中有提到, 栅格地图的优点是直观、易于扩展和便于维护, 特别适用于处理激光或者超声波数据; 但缺点也很明显, 为了充分获得环境的细节, 方便精确的位姿估计, 地图分辨率当然越小越好, 存储空间和计算量正比于栅格数量, 如果室内环境规模较大, 设置的分辨率过高造成栅格数量过大, 会大大增加计算负担。所以在初始化栅格地图的时候, 需要设定合理的地图的边界值和分辨率, 这些值决定了该地图能容纳的数据量和运算时间。本文将地图的分辨率设为 0.01m, 是根据所采用的 LiDAR 的相对定位精度来决定的, 如果分辨率过低则凸显不出 LiDAR 的高精度性能, 如果分辨率过高, 不仅增加了计算负荷, 对制图结果的显示也没有更显著地提升。

本文第五章的实验中所采集的数据均来自芬兰地理测量研究所(Finnish Geodetic Institute, FGI)的图书馆, 该图书馆的面积大约为 80×80m, 如果地图的分辨率设置为 0.01m, 那么地图中将会需要 8000×8000 个栅格。假如采集数据的场所面积更大, 栅格数量更多, 就会使得地图的运算超出计算机的处理能力。因此, 本文对栅格地图采用分层搜索的方式, 每一层采用不同的分辨率, 由粗到精, 这种方法可以很好地解决运算量与分辨率之间的矛盾, 使得该地图能适用于任何室内环境, 而不受环境尺寸的限制。

整个地图结构如图 2-1 所示, 类似于金字塔结构, 在地图初始化时, 需要设置地图的总层数 n 、地图的最小分辨率 d_{xy} (本文中为 0.01m)、地图的中心、地理边界值等参数。将地图自上而下地进行分块, 直至分成面积为 $d_{xy} \times d_{xy}$ 的栅格, 当进行地图搜索时, 先粗后精地进行匹配搜索, 在分辨率较低的图层匹配出精度较低的位置后, 会进一步在更精细地图层进行搜索, 得到精度更高的定位结果, 类似于 Habbecke(2006)中所用的方法。这种自上而下一层一层搜索的搜索方式大大减少了不必要的搜索区域, 提高了搜索效率。但与计算机视觉中的图像金字塔的形成过程不同的是, 本文在创建多分辨率地图时, 并不是将一个单一的高精度地图通过高斯滤波或者降采样的方式进行实现(张仲颖, 1994), 而是将不同层次的地图都保存在内存中, 在匹配的过程中通过得到的位姿估计对它们进行同步地更新, 这样可以保证不同尺度的地图在同一时刻都是一致的, 避免了因降采样等

操作带来的计算量增大等问题。这种金字塔结构还有其他的优点，将在本文的 2.3.3 节中继续描述。本文为了统一坐标，将地图的世界坐标系选用 n 系。

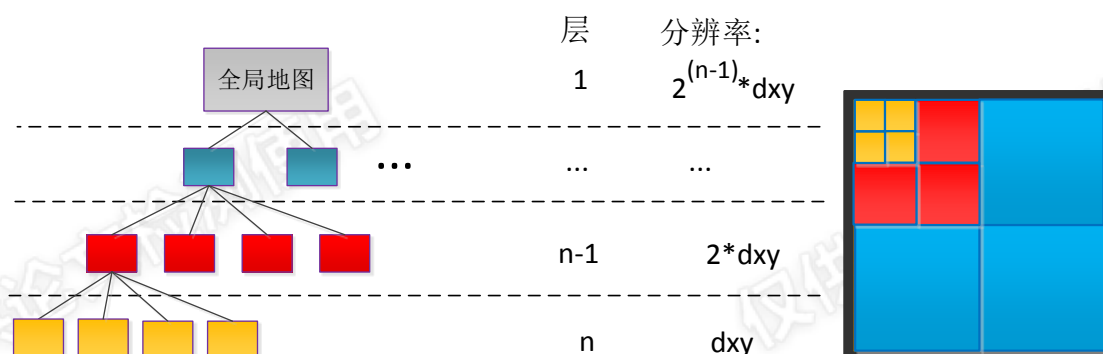


图 2-1 占据栅格地图的金字塔结构

栅格地图中存储着先前所有帧 LiDAR 点云的匹配信息，用概率值来表示。每一个栅格可赋予预先设定好的四种概率值—0.1, 0.3, 0.6 和 0.9，代表实际环境中的障碍物存在在此栅格中的概率。如图 2-2(a)所示，在初始化时，所有的栅格都被赋予 0.1 的概率值，在地图上显示为黑色，将 LiDAR 扫描的点云投影到地图上后，如果某个栅格被一个 LiDAR 点云占据，那么将这个栅格的概率值设成 0.9，在地图上显示为白色，再将围绕着该栅格的两圈栅格的概率值分别设成 0.6 和 0.3。根据概率值的大小显示栅格的灰度级。如果某个栅格的概率值被重复赋予，那么选取较大值作为该栅格的概率值，如图 2-2(b)所示。通过这种方式，可以快速形成一幅概率地图。

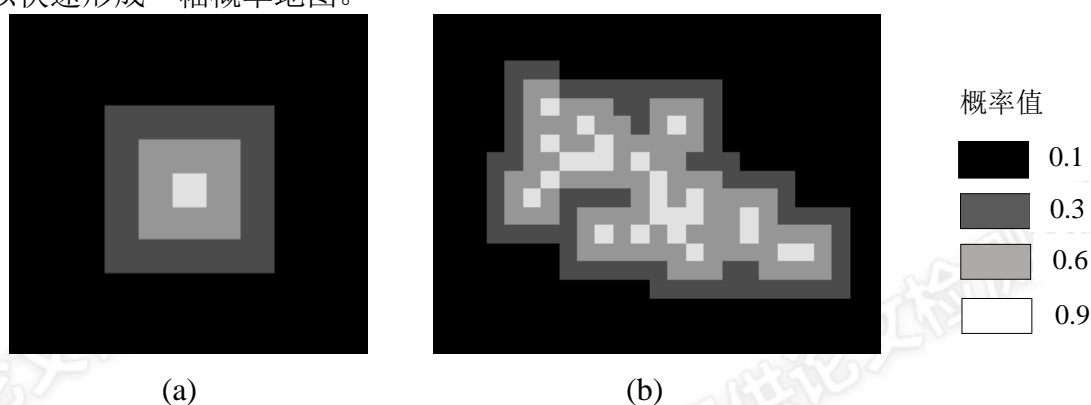


图 2-2 占据栅格地图的概率表示

2.3 LiDAR 点云匹配算法研究

2.3.1 LiDAR 传感器数学模型

LiDAR，也称作激光雷达或激光测距仪，通过发射激光来探测出与周围障碍物的距离，能够提供丰富的环境感知信息，测量范围广，精度高，不受环境光照等影响，被越来越多地应用于 SLAM 中。本文所采用的 2D LiDAR 扫描分辨率为 0.25° ，每一帧能返回 $-135^\circ \sim +135^\circ$ 范围内共 1081 个点云数据。LiDAR 采集的原始数据是以 LiDAR 为中心的局部坐标系中的极坐标点，如图 2-4 所示，自顶向下看，逆时针扫描，用极坐标 (r_i, θ_i) ($i=1, \dots, 1081$) 来表示， r_i 为第 i 个障碍物到 LiDAR 中心的距离， θ_i 为第 i 条激光束与起始扫描点之间的夹角，其中 $\theta_i = 0.25 * i$ 。

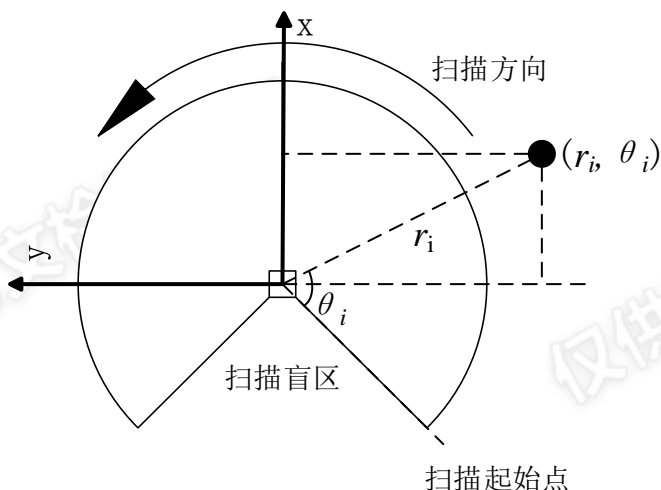


图 2-3 LiDAR 扫描示意图

在绘制地图时，需要将观测数据转换到全局坐标系下，将 LiDAR 扫描数据映射到全局坐标系下的公式为：

$$x_i = r_i * \cos\left(\frac{\theta_i - 135}{180} * \pi\right) \quad (2.1)$$

$$y_i = r_i * \sin\left(\frac{\theta_i - 135}{180} * \pi\right) \quad (2.2)$$

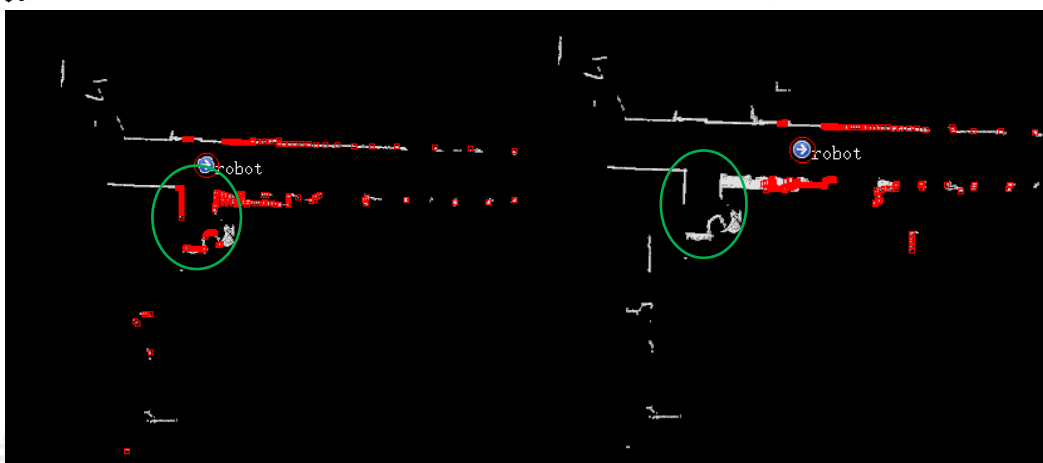
其中 (x_i, y_i) 为障碍物在全局坐标系下的坐标。

2.3.2 基于 IMLE 的 LiDAR 匹配算法

LiDAR 点云匹配算法主要分为两种：点云帧与帧之间的匹配 (Scan-to-scan matching) 和点云帧与地图的匹配 (scan-to-map matching)。比较经典的帧与帧之间

匹配的算法有：ICP(Iterative Closed Point)算法、ICL(Iterative Closed Line)算法(Bosse, 2004; Censi, 2008)等，主要是通过前后两帧 LiDAR 感知的环境中对应的点或者线信息，求出一个相应的转换关系，从而得出移动机器人在这段时间的位移和姿态信息量。而点云帧与地图的匹配算法(Biber, 2004; Bosses, 2004)，不仅仅是前后两帧的匹配，还有点云帧与已形成的地图的匹配，这种方法得到的定位结果更准确，但付出的代价是匹配的工作量大、耗时长。

2.1 节中提到，LiDAR 点云匹配的过程是一种相对定位的过程。如图 2-4 所示，假如一个移动机器人在室内的地面上从位置 x_0 移动到了 x_1 ，在位置 x_0 和 x_1 ，它会分别接受到 LiDAR 扫描到的点云 S_0 和 S_1 。扫描匹配的过程就是找到一个刚体转换 T，使得 S_t 能与 S_{t-1} 匹配。在 2D 环境中，T 由 $(\Delta x, \Delta y, \Delta \theta)$ 组成，分别表示移动平台的位置平移量和航向旋转量。扫描匹配中最重要的问题是在获得高精度的定位结果的同时，要尽量减小运算量。但 ICP 和 ICL 算法仅仅是寻找连续两帧或多帧点云中所有点的对应关系，即 scan-to-scan matching 算法。每一帧扫描数据仅仅只是反映了小部分环境，而且还夹杂着噪声，难以保证当前帧的所有点在下一帧都能找到对应的点。例如图 2-4 中，红色部分是 LiDAR 扫描到的当前帧的点云，白色部分是已经形成的地图。图(a)的绿色圆圈所圈起来的 LiDAR 点云在图(b)中并没有对应的点。在这种情况下，ICP 或者 ICL 算法会得到不准确的结果。



(a)

(b)

图 2-4 (a)机器小车平台在位置 x_0 时得到的 LiDAR 点云信息(b)机器小车平台在位置 x_1 时得到的 LiDAR 点云信息

以前的研究工作(Tang, 2014)中详细介绍了一种 san-to-map matching 的

LiDAR 匹配算法- IMLE(Improved probabilistically-motivated Maximum Likelihood Estimation), 这是基于 MLE(Maximum Likelihood Estimation)算法做出改进后的算法。与 ICP 或者 ICL 算法不同的是, MLE 和 IMLE 算法(Tang, 2014)不仅仅只是点云帧与帧之间的匹配, 还对已形成的地图中所包含的所有点云进行匹配来寻找最优结果, 如图 2-4 中的白色部分。这就意味着 IMLE 算法相对于 ICP 等算法来说, 大大增加了点云匹配的计算量。概率地图 M 中存储了由之前的 LiDAR 扫描帧得到的各个栅格的概率值, 当新接受到一帧 LiDAR 扫描数据 S_t , 就会与地图进行匹配, 当匹配的概率值 $P(S_t|M)$ 最大时, 得到关系转换 T^* 为最佳转换, 地图 M 中的某个点 p_i 的概率值 $P(p_i|M)$ 与该点到最近的环境特征 F 之间的距离 d 和测量噪声的比例因子 σ 成正态分布:

$$P(p_i|M) \propto e^{(-d(p_i,F)/\sigma)} \quad (2.3)$$

$$P(S_t|M) = \prod_{i=1}^n P(p_i|M) \quad (2.4)$$

$$T^* = \operatorname{argmax}(P(T \propto S_t|M)) \quad (2.5)$$

该方法优点是鲁棒性好而且精度高, 但是缺点也很明显, 它虽然不像 MLE 算法那样对全局的地图进行搜索, 但是仍然需要依靠人为设置的搜索范围来寻找位置的最优解, 合理的搜索范围参数能帮助过滤掉完全不需要搜索的区域, 不用再浪费大量的时间进行全局搜索; 但如果范围设置的过小, 这种搜索方式便很容易陷入局部最优解, 不能得到准确的位置结果, 从而造成地图混乱。因此, 搜索范围的设置对 LiDAR 扫描匹配的效率至关重要。但在搜索过程中每次搜索中的对应点列都是不一样的, 严格地说, 最优的搜索范围应该也会不一样, 人为给定的搜索范围更多的是经验值, 需要比每次搜索的最优范围都大, 才能保证每次搜索的结果都是最优解, 仍然不能将搜索时间减小到最低。

2.3.3 高斯-牛顿迭代算法

对于较大的室内场所，IMLE 算法所耗费的时间成本已经远远不能满足人们的需求，所以唐健老师将原有的 LiDAR 匹配算法改进后，使得 LiDAR 搜索匹配后的定位结果既能保持之前的精度，又能降低计算量，减小耗时。

(1) 双线性插值预处理

改进的 LiDAR 点云匹配算法借鉴了 1.2.1 节中所提到的 Hector SLAM 算法，利用地图的空间强度梯度信息，通过高斯-牛顿迭代算法来寻求一个 LiDAR 激光束集到已有地图的最佳映射关系，该方法最早是出现在计算机视觉研究中(Lucas, 1981)，是以梯度下降法(Gradient Descent, GD)为基础的算法。梯度下降法又称最速下降法，是沿梯度下降(或上升)的方向求解极小值的一种方法，以梯度下降(或上升)最快的方向为搜索方向，越接近目标值，步长越小，前进越慢。这种算法计算量小，能迅速找到最优解，因此在很多大规模数据集的处理上都会运用到梯度下降法或其改进算法。

在计算地图的梯度的时候，由于栅格地图是离散化的图像，每一个栅格拥有自己的概率值，类似于计算机图形学中图像上像素的灰度级，无法直接计算其偏微分，因此本文采用双线性插值的方法来对定位结果进行内插值处理，经过插值后，整个栅格地图能够近似地看作是连续的概率分布模型，在保证制图精度的情况下用以估计栅格占据概率和对结果求偏微分，双线性插值算法在计算机图形学中应用得非常普遍(王森，2008)。

具体过程为：假设一个 LiDAR 点在世界坐标系下的坐标为 P_m ，是一个非整数，而地图中每个栅格的坐标都定义成整数，所以要求点 P_m 的占据概率 $M(P_m)$ 和梯度 $\nabla M(P_m) = (\frac{\partial M}{\partial x}(P_m), \frac{\partial M}{\partial y}(P_m))$ 时，可以近似地用四个离它最近的整数坐标 $P_{00}, P_{01}, P_{10}, P_{11}$ 内插得到，如图 2-5 所示。红色的点为已知概率值点，绿色的点 P_m 为待插值。

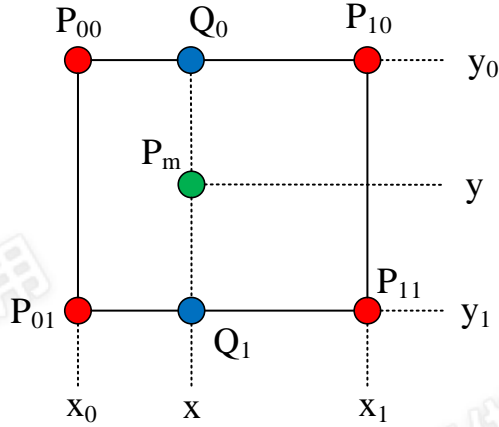


图 2-5 占据栅格地图的双线性过滤

沿着 x 轴线性插值，在 P_{00}, P_{10} 中插入蓝色的点 Q_0 ， P_{01}, P_{11} 中插入蓝色点 Q_1 ，求出 Q_0, Q_1 的概率值，即 $M(Q_0), M(Q_1)$ 后，再沿着 y 轴线性内插，计算出 Q_0 与 Q_1 之间的插入点 P_m 的占据概率，即：

$$M(P_m) \approx \frac{y-y_0}{y_1-y_0} \left(\frac{x-x_0}{x_1-x_0} M(P_{11}) + \frac{x_1-x}{x_1-x_0} M(P_{01}) \right) + \frac{y_1-y}{y_1-y_0} \left(\frac{x-x_0}{x_1-x_0} M(P_{10}) + \frac{x_1-x}{x_1-x_0} M(P_{00}) \right) \quad (2.6)$$

对 $M(P_m)$ 求偏微分可得：

$$\frac{\partial M}{\partial x}(P_m) \approx \frac{y-y_0}{y_1-y_0} (M(P_{11}) - M(P_{01})) + \frac{y_1-y}{y_1-y_0} (M(P_{10}) - M(P_{00})) \quad (2.7)$$

$$\frac{\partial M}{\partial y}(P_m) \approx \frac{x-x_0}{x_1-x_0} (M(P_{11}) - M(P_{10})) + \frac{x_1-x}{x_1-x_0} (M(P_{01}) - M(P_{00})) \quad (2.8)$$

其中， $M(P_{00})$ 、 $M(P_{01})$ 、 $M(P_{10})$ 、 $M(P_{11})$ 可根据 2.2.2 节栅格概率值的赋予方法得出。

(2) 扫描匹配算法过程

当当前帧的 LiDAR 激光点云能与已存在的地图匹配得上时，也就意味着它能够与之前的所有帧点云都能匹配得上，所以高斯-牛顿算法也属于 scan-to-map matching 的方法。但该方法与 IMLE 算法的不同之处在于它只需要设置一个合理的位姿初始值，不需要设置搜索范围，大大减少了搜索次数，因此运算速度更快，效率更高。

高斯-牛顿方法的基本思想是通过多次迭代，寻找出一个最优解 $\xi = (x, y, \varphi)^T$

使得对应的这帧 LiDAR 点云在栅格地图上表示的概率值的残差之和 ξ^* 最小。其中 ξ^* 表示为:

$$\xi^* = \sum_{i=1}^n [1 - M(S_i(\xi))]^2 \quad (2.9)$$

其中 x, y, φ 分别表示移动机器人在世界坐标系下的位置 x, y 和航向 φ ; n 为一帧 LiDAR 点云中点的总个数; 将 LiDAR 激光点的全部坐标 $s_i = (s_{i,x}, s_{i,y})^T$ 转换到世界坐标 $S_i(\xi)$, 为:

$$S_i(\xi) = \begin{pmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{pmatrix} \begin{pmatrix} s_{i,x} \\ s_{i,y} \end{pmatrix} + \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.10)$$

$M(S_i(\xi))$ 是 $S_i(\xi)$ 在地图上的占据概率。

所以在给出 ξ 的初始估计后, 要求经过多次迭代后求出初始估计与实际最优解之间的误差 $\Delta\xi$, 使得:

$$\sum_{i=1}^n [1 - M(S_i(\xi + \Delta\xi))]^2 \rightarrow 0 \quad (2.11)$$

对 $M(S_i(\xi + \Delta\xi))$ 进行一阶泰勒展开后得到:

$$\sum_{i=1}^n \left[1 - M(S_i(\xi)) - \nabla M(S_i(\xi)) \frac{\partial S_i(\xi)}{\partial \xi} \Delta\xi \right]^2 \rightarrow 0 \quad (2.12)$$

要求出 $\Delta\xi$ 取何值时函数(2.12)的值最小, 则需要函数(2.9)关于 $\Delta\xi$ 的偏导等于0, 即:

$$2 \sum_{i=1}^n \left[\nabla M(S_i(\xi)) \frac{\partial S_i(\xi)}{\partial \xi} \right]^T \left[1 - M(S_i(\xi)) - \nabla M(S_i(\xi)) \frac{\partial S_i(\xi)}{\partial \xi} \Delta\xi \right] = 0 \quad (2.13)$$

由等式(2.13)解得:

$$\Delta\xi = \mathbf{H}^{-1} \sum_{i=1}^n \left[\nabla M(S_i(\xi)) \frac{\partial S_i(\xi)}{\partial \xi} \right]^T \left[1 - M(S_i(\xi)) \right] \quad (2.14)$$

其中

$$\mathbf{H} = \left[\nabla M(S_i(\xi)) \frac{\partial S_i(\xi)}{\partial \xi} \right]^T \left[\nabla M(S_i(\xi)) \frac{\partial S_i(\xi)}{\partial \xi} \right] \quad (2.15)$$

\mathbf{H} 也叫 Hessian 矩阵, 地图梯度 $\nabla M(S_i(\xi))$ 可由等式(2.6)(2.7)近似求出。再由等式(2.10)可以得到:

$$\frac{\partial S_i(\xi)}{\partial \xi} = \begin{pmatrix} 1 & 0 & -\sin(\varphi) s_{i,x} - \cos(\varphi) s_{i,y} \\ 0 & 1 & \cos(\varphi) s_{i,x} - \sin(\varphi) s_{i,y} \end{pmatrix} \quad (2.16)$$

在求出 $\nabla M(S_i(\xi))$ 和 $\frac{\partial S_i(\xi)}{\partial \xi}$ 后, 根据高斯-牛顿方程(2.14)即可求出最优解 $\Delta\xi$ 。

整个高斯-牛顿算法可以归纳为以下四个步骤, 具体流程图见图 2-6:

- (1) $k = 0$, 首先给定一个近似的位姿初始值 ξ , 这个初始值可以由 IMU 或者里程计等给出, 本文是根据 IMU 的定位结果给出的;
- (2) 估计出地图梯度向量 $\nabla M(S_i(\xi))$ 和 Hessian 矩阵 \mathbf{H} ;
- (3) 计算出第 $k + 1$ 次的位姿估计 $\xi + \Delta\xi$, $\Delta\xi$ 由(2.14)式给出;

(4) 如果 $\Delta\xi$ 收敛，即 $\Delta\xi \rightarrow 0$ ，则算法结束， $\xi + \Delta\xi$ 为 LiDAR 扫描匹配的定
位结果；如果没有收敛，那么 $\xi = \xi + \Delta\xi$ ，再重复步骤(2)(3)(4).

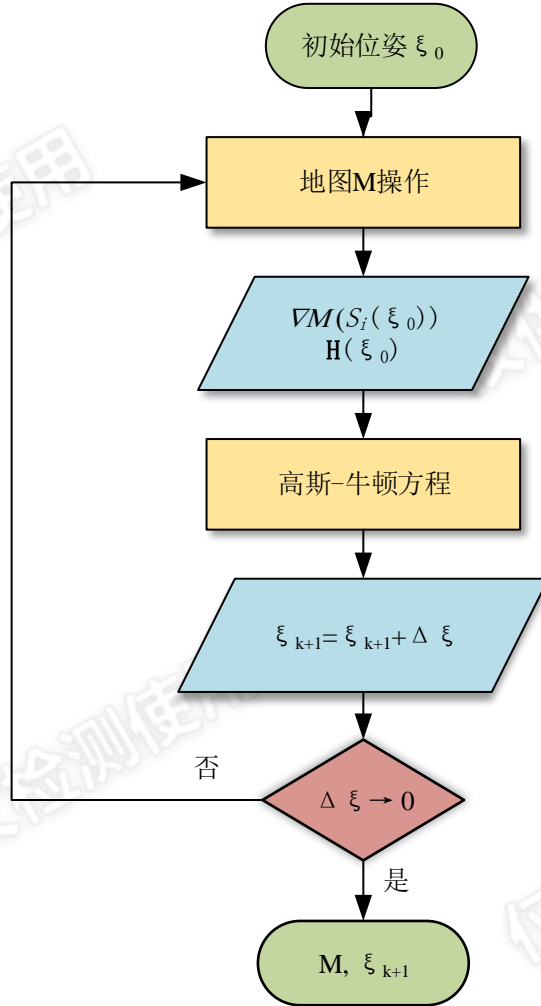


图 2-6 LiDAR 扫描匹配算法流程图

通过高斯-牛顿算法还可以估计出 LiDAR 扫描匹配的误差，即高斯-牛顿算
法的不确定度，这在 SLAM 后端的优化处理中是十分有用的。本文中所用的方
法是通过每一次迭代过程中生成 Hessian 矩阵 \mathbf{H} 来估计出协方差。当 $\Delta\xi$ 收敛时，
得到此时估计出来的位姿 $\xi + \Delta\xi$ 和 Hessian 矩阵 $\mathbf{H}(\xi + \Delta\xi)$ ，那么协方差矩阵可以
近似表示为：

$$\text{CovR} = \text{Var}\{\xi + \Delta\xi\} = \sigma^2 * \mathbf{H}^{-1} \quad (2.13)$$

其中 σ 是比例因子，由 LiDAR 的性能决定，本文给 σ 赋值为 1。公式(2.13)
的详细推导过程参见(Burguera, 2009)。改进后的高斯-牛顿算法的性能分析将在
本文的第五章的实验中详细描述。

但是任何基于梯度信息的算法都容易陷入局部最优解中，例如本节中所用的
高斯-牛顿算法。但由于 2.2.2 节中提到，本文所创建的是类似于金字塔结构的

分辨率地图，基于这种地图的搜索方式可以有效地帮助缓解 LiDAR 匹配陷入局部最优解的问题(Kohlbrecher, 2011)。当地图层的分辨率越低时，越容易找到全局最优解，将估计出来的位姿结果作为下一层地图的搜索初值，可以帮助缓解直接在高分辨率地图上进行搜索匹配而找出错误的最优方向的问题。但也并不能完全避免局部收敛。

通过本节详细描述的高斯-牛顿算法的实现过程可以看出，位姿初始值 ξ 的给定是十分重要的，它决定了高斯-牛顿算法的迭代次数和收敛情况。如果给定的 ξ 初值非常准确，已经十分接近最优解了，那么高斯-牛顿算法能很快地迭代出最优解。但如果 ξ 的初值误差很大，就需要多次迭代才能找到最优解甚至找不出最优解，这样不仅大大增加了计算量，而且很容易陷入局部最优中。所以高斯-牛顿迭代算法的正确性和精度非常依赖于初值的可靠程度。本文 5.3 节和 5.4 节中的组合导航算法和实时性算法改进的工作都是基于这种高斯-牛顿的 LiDAR 匹配算法上开展的。

3 LiDAR/INS 组合定位定姿

3.1 引言

LiDAR/INS 组合导航结合了 LiDAR 和 INS 这两种导航定位技术的优势，取长补短，可实现在室内连续工作的高采样率、高精度的组合系统。LiDAR 扫描匹配技术虽然能够提供高精度的位置、航向定位结果，但是遇到环境特征不明显的情况，LiDAR 的定位精度便大幅度下降，容易受到外界环境的干扰，而且 LiDAR 采样频率低，最高频率仅为 40Hz。惯性导航技术相对于 LiDAR 定位技术而言，它不受外界环境的影响，具备自主导航的能力，而且它的输出频率高，一般为 200Hz，能够提供高频率的位置、速度、姿态全导航参数。但是，惯性导航的导航定位误差会随着时间迅速累积，所以单独的惯性导航仅仅只能在很短的时间内提供高精度的导航定位结果。

本文所采用的 LiDAR/INS 组合导航的信息融合技术借鉴了 GNSS/INS 松组合技术(Shin, 2005)，将 LiDAR 扫描匹配算法解算得到的位置、姿态观测量作为组合卡尔曼滤波器的输入量，与 INS 机械编排得到的位置、姿态结果做组合。其优势在于：LiDAR、INS 各自数据处理具备独立性，便于互相检查核对；可靠性高，一旦 LiDAR 或者 INS 工作失灵，另外一个仍能正常工作，整个系统还可以连续地提供导航参数；另外，松组合算法中构造卡尔曼滤波量测方程相对简单，减小了计算量。

目前很多利用 IMU 传感器进行辅助的室内 SLAM 工作中(杨景阳, 2014; Hess, 2016)，仅仅只将陀螺的输出量进行积分得到姿态信息，然后将姿态信息用于辅助 LiDAR 匹配或者其他传感器的定位，当 LiDAR 或者其他传感器得到更准确的定位信息后再去修正惯导的姿态，没有用到组合的方式。所以这种方法没有完全发挥出 INS 的优势，INS 在短时间内的定位精度也较高，再加上它的高采样率，不仅可以得到高输出量的定位结果，而且如果将 INS 得到的位置和姿态信息与 LiDAR 或者其他传感器得到的位置或姿态信息组合，能够得到比单一传感器更高精度的位置和姿态信息。Tang(2015)中所用的组合导航算法中只将 IMU 的位置、速度、姿态作为状态向量，用 LiDAR 匹配得到的结果进行修正，并没有实时估计出 IMU 的陀螺零偏、加速度计零偏等误差参数，而是在初始时给定出来。事实上陀螺零偏和加速度计零偏是随着时间的变化而变化的。从本文的 3.4.3

节中将看到, 本文把陀螺零偏和加速度计零偏建模成了一阶高斯马尔可夫过程, 如果在整个组合的过程中都采用给定的零偏值, 那么 INS 机械编排估计出来的位置误差、速度误差、姿态误差绝对是不准确的。本文的第 2.3.3 节中已经指出, LiDAR 匹配算法性能非常依赖于初值的给定, 而初值是由 INS 机械编排后给出, 所以对原有的组合导航算法进行改进, 实时估计出 IMU 的陀螺零偏和加速度计零偏并进行误差补偿是十分有必要的。

本章第二节介绍了常用的坐标系统; 第三节推导了 n 系下的惯性导航机械编排算法, 第四节详细介绍了改进前后的 LiDAR/INS 组合导航卡尔曼滤波算法。

3.2 常用坐标系统

在导航系统的研究中, 坐标系的概念非常重要, 只有在相对意义下, 物体的运动和在空间的位置才有意义, 它是导航计算中的数学基石。导航计算中最常用的坐标系包括惯性坐标系、地心地固坐标系、导航坐标系、载体坐标系等。在移动制图系统中, 载体为移动机器人。

(1) 惯性坐标系(i 系)

惯性系 i 系的原点取在地球中心, z 轴与地球自转轴平行, 指向北极点, x 、 y 轴均位于赤道平面内, x 轴朝向春分点, y 轴与 x 、 z 轴构成右手坐标系。 i 系是一种理想坐标系, 它相对于惯性空间静止或做匀速直线运动, 不存在旋转或者加速度。惯性仪表的测量值都是以 i 系为参考基准。

(2) 地心地固坐标系(e 系)

常用的 e 系定义是 1984 世界大地坐标系(WGS-84), 原点取在地球中心, x 轴在赤道平面内指向平均格林尼治子午圈, z 轴平行于地球平均自转轴指向北极, y 轴与 x 轴和 z 轴构成右手坐标系。

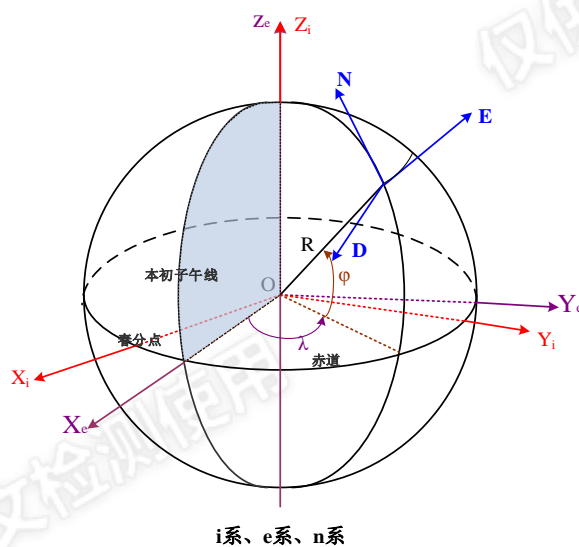
(3) 导航坐标系(n 系)

导航坐标系即当地水平坐标系, 是载体姿态表达的参照坐标系。原点在载体的中心, x 轴沿经线切向方向指向北, y 轴沿纬线切向方向指向东, z 轴与 x 、 y 轴构成右手坐标系, 又称北东地(North-East-Down, NED)。本文选用的参考坐标系是即为 n 系。

(4) 传感器坐标系(b 系)

传感器坐标系是指捷联惯性传感器的坐标系，定义了运动物体的位置参考点以及姿态参考方向。坐标原点位于 IMU 的测量中心，x 轴指向载体的正前方，y 轴指向载体的右方，z 轴垂直于 x 轴和 y 轴向下，构成右手坐标系。在本文中，认为传感器坐标系与车体坐标系(vehicle frame, v 系)的坐标轴指向一致，也可统称为载体坐标系。

图 3-1 展示了 i 系、e 系、n 系的坐标系。其中 X、Y 和 Z 表示轴线，其上标代表相应的坐标系 φ 和 λ 为载体所在的位置的维度和经度。



i系、e系、n系

图 3-1 参考坐标系

3.3 仅靠 INS 姿态辅助的算法

由于 INS 对载体动态的感知能力比 LiDAR 匹配要好，因此很多室内 SLAM 的研究工作将 IMU 的陀螺仪的输出量进行积分得到姿态信息，为 LiDAR 匹配提供姿态搜索初值，再将 LiDAR 匹配后得到的准确姿态结果去修正惯导的误差。具体结构图如图 3-3 所示。姿态推算的过程是：利用前一时刻和当前时刻的 IMU 陀螺测量值计算出中间时刻的姿态四元数，再通过前一时刻的姿态四元数递推出当前时刻的姿态四元数，从而求出当前时刻的姿态转移矩阵 C_b^n 。这种做法省去了卡尔曼滤波算法的耗时，所以计算速度更快。

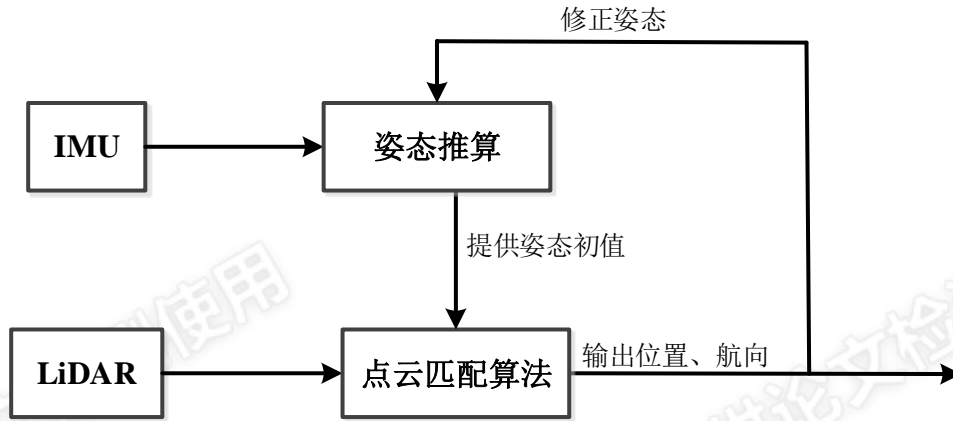


图 3-2 仅靠 INS 姿态辅助 LiDAR 匹配算法结构图

从理论上讲,这种方法存在着得到定位结果不够准确的风险,不能仅仅依靠 LiDAR 和 INS 进行定位,一般还需要再搭载一个其他的辅助传感器,如超声波等(杨景阳, 2014),这样反而会加大计算量。

3.4 惯性导航机械编排算法

惯性导航系统由惯性传感器陀螺仪和加速度计组成,分别用来测量载体相对于惯性空间的转动角速度和比力(定义为载体相对于惯性空间的全加速度和引力加速度的矢量差)信息,机械编排的主要任务是利用陀螺仪和加速度计的输出值,通过位置、速度、姿态更新算法来实现导航参数的求解,它是一个递推计算的过程。

3.4.1 传感器误差补偿

由于陀螺仪和加速度计的输出值不可避免地存在误差,例如零偏误差、交轴耦合误差、比例因子误差和随机噪声等。所以在导航解算时必须先对惯性器件的输出值进行误差补偿。本论文中不考虑交轴耦合和比例因子的影响。

陀螺的角增量输出和加速度计的速度增量输出可以通过式(3.1)进行补偿(Shin,2005):

$$\Delta\theta_k = \Delta\tilde{\theta}_k - \mathbf{b}_g^0\Delta t_k \quad (3.1)$$

$$\Delta\mathbf{v}_{f,k}^b = \Delta\tilde{\mathbf{v}}_{f,k}^b - \mathbf{b}_a^0\Delta t_k \quad (3.2)$$

其中 $\Delta\tilde{\theta}_k$ 和 $\Delta\tilde{\mathbf{v}}_{f,k}^b$ 分别为陀螺仪和加速度计在 t_k 时刻的速度增量和角速度增量输出, $\Delta t_k = t_k - t_{k-1}$,三维矢量 \mathbf{b}_g^0 和 \mathbf{b}_a^0 分别表示已知的陀螺零偏和加速度计

零偏。

由于改进前的组合导航算法中陀螺零偏和加速度计零偏是给定的，所以式(3.1)和式(3.2)中的 \mathbf{b}_g^0 和 \mathbf{b}_a^0 为常值。但改进后的组合导航算法将陀螺零偏和加速度计零偏建模成一阶高斯马尔可夫过程(Shin,2005)，并在计算过程中实时估计出零偏的变化，进行补偿，详细过程将在 3.4.3 节给出。

3.4.2 速度更新

在连续时间下，根据哥氏定理可以推导出惯导在 n 系下的速度微分方程为：

$$\dot{\mathbf{V}}_{en}^n = \mathbf{C}_b^n \mathbf{f}^b + \mathbf{g}^n - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \mathbf{V}_{en}^n \quad (3.3)$$

其中 \mathbf{V}_{en}^n 为载体在 e 系下的速度(即地速)相对于 n 系并投影在 n 系下的速度矢量， \mathbf{f}^b 是加速度计输出的比力， $2\boldsymbol{\omega}_{ie}^n \times \mathbf{V}_{en}^n$ 为哥氏加速度， $\boldsymbol{\omega}_{en}^n \times \mathbf{V}_{en}^n$ 为载体对地向心加速度， \mathbf{g}^n 为重力加速度。

对速度微分方程(3.3)进行离散化后得到速度更新方程：

$$\mathbf{V}_k^n = \mathbf{V}_{k-1}^n + \Delta \mathbf{V}_{f,k}^n + \Delta \mathbf{V}_{g/cor,k}^n \quad (3.4)$$

式中 \mathbf{V}_{k-1}^n 为上一时刻 t_{k-1} 的惯导速度， \mathbf{V}_k^n 为待求的当前惯导速度。

$$\Delta \mathbf{V}_{f,k}^n = \int_{t_{k-1}}^{t_k} \mathbf{C}_b^n \mathbf{f}^b dt \quad (3.5)$$

$$\Delta \mathbf{V}_{g/cor,k}^n = \int_{t_{k-1}}^{t_k} [\mathbf{g}^n - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \mathbf{V}^n] dt \quad (3.6)$$

其中 $\Delta \mathbf{V}_{f,k}^n$ 为由比力引起的速度增量在导航坐标系下的投影，简称比力积分项。 $\Delta \mathbf{V}_{g/cor,k}^n$ 为由重力加速度和哥氏加速度引起的速度增量在导航坐标系下的投影，简称重力/哥氏积分项。

比力积分项 $\Delta \mathbf{V}_{f,k}^n$ 是由比力引起的速度增量，可以写成如下(Savage,1998)：

$$\Delta \mathbf{V}_{f,k}^n = \frac{1}{2} [\mathbf{C}_{n(k-1)}^{n(k)} + \mathbf{I}] \mathbf{C}_{b(k-1)}^{n(k-1)} \Delta \mathbf{V}_{f,k}^{b(k-1)} \quad (3.7)$$

由于本文实验移动机器人小车没有高角振动且低速行驶，可以假设在相邻积分区间内角速度和比力随时间线性变化，对积分的旋转效应和划桨效应进行补偿(秦永元，2006)，可得：

$$\begin{aligned} \Delta \mathbf{V}_{f,k}^{b(k-1)} &= \int_{t_{k-1}}^{t_k} \mathbf{C}_{b(t)}^{b(k-1)} \mathbf{f}^b dt \\ &\approx \Delta \mathbf{V}_{f,k}^b + \frac{1}{2} \Delta \boldsymbol{\theta}_k \times \Delta \mathbf{V}_{f,k}^b \\ &\quad + \frac{1}{12} (\Delta \boldsymbol{\theta}_{k-1} \times \Delta \mathbf{V}_{f,k}^b + \Delta \mathbf{V}_{f,k-1}^b \times \Delta \boldsymbol{\theta}_k) \end{aligned} \quad (3.8)$$

$\Delta \mathbf{V}_{f,k}^b$ 和 $\Delta \boldsymbol{\theta}_k$ 分别表示加速度计和陀螺在 t_k 时刻输出的的速度增量和角速度增量。把式 $\mathbf{C}_{n(k-1)}^{n(k)} = \mathbf{I} - (\boldsymbol{\zeta}_k \times)$ 代入式(3.7)中可得：

$$\Delta \mathbf{V}_{f,k}^n = [\mathbf{I} - (0.5\boldsymbol{\zeta}_k \times)] \mathbf{C}_{b(k-1)}^{n(k-1)} \Delta \mathbf{V}_{f,k}^{b(k-1)} \quad (3.9)$$

$$\boldsymbol{\zeta}_k = [\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n]_{k-1/2} \Delta t_k \quad (3.10)$$

式中 $t_{k-1/2}$ 表示 t_{k-1} 和 t_k 的中间时刻, $\Delta t_k = t_k - t_{k-1}$.

重力/哥氏积分项中的被积函数在积分周期内变化缓慢, 本文中室内移动机器人平台运行速度大约为1m/s, 属于低速运动, 则式(3.6)可简化如下:

$$\Delta \mathbf{V}_{g/cor,k}^n = [\mathbf{g}^n - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \mathbf{V}^n]_{t_{k-1/2}} \Delta t_k \quad (3.11)$$

3.4.3 位置更新

本节对于位置的更新采用四元数更新法(Shin, 2005), 其中经纬度的更新算法如下:

$$\mathbf{q}_{n(k)}^{e(k)} = \mathbf{q}_{e(k-1)}^{e(k)} \mathbf{q}_{n(k-1)}^{e(k-1)} \mathbf{q}_{n(k)}^{n(k-1)} \quad (3.12)$$

式中 t_{k-1} 时刻的位置四元数 $\mathbf{q}_{n(k-1)}^{e(k-1)}$ 为已知信息, $\mathbf{q}_{n(k)}^{n(k-1)}$ 和 $\mathbf{q}_{e(k-1)}^{e(k)}$ 的表达式为:

$$\mathbf{q}_{n(k)}^{n(k-1)} = \begin{bmatrix} \cos \|\mathbf{0.5}\boldsymbol{\zeta}_k\| \\ \frac{\sin \|\mathbf{0.5}\boldsymbol{\zeta}_k\|}{\|\boldsymbol{\zeta}_k\|} \boldsymbol{\zeta}_k \end{bmatrix} \quad (3.13)$$

$$\mathbf{q}_{e(k-1)}^{e(k)} = \begin{bmatrix} \cos \|\mathbf{0.5}\boldsymbol{\xi}_k\| \\ -\frac{\sin \|\mathbf{0.5}\boldsymbol{\xi}_k\|}{\|\boldsymbol{\xi}_k\|} \boldsymbol{\xi}_k \end{bmatrix} \quad (3.14)$$

其中 $\boldsymbol{\xi}_k = \boldsymbol{\omega}_{ie}^e \Delta t_k$ 表示相邻历元间 e 系对应的旋转矢量。由于速度更新已经完成, 所以在计算 $\boldsymbol{\zeta}_{k-1}$ 时需要用到的中间时刻的速度可以通过线性内插得到。经度和纬度可以从 \mathbf{q}_n^e 中解析获得(Shin, 2005)。高程可以通过中间时刻的速度计算完成:

$$h_k = h_{k-1} - v_{D,k-1} \Delta t_k \quad (3.15)$$

3.4.4 姿态更新

采用更新四元数的方法来完成姿态的更新, 如下所示:

$$\mathbf{q}_{b(k)}^{n(k-1)} = \mathbf{q}_{b(k-1)}^{n(k-1)} \mathbf{q}_{b(k)}^{b(k-1)} \quad (3.16)$$

$$\mathbf{q}_{b(k)}^{n(k)} = \mathbf{q}_{n(k-1)}^{n(k)} \mathbf{q}_{b(k)}^{n(k-1)} \quad (3.17)$$

其中, 对于 b 系从 t_{k-1} 到 t_k 时刻的更新四元数为:

$$\mathbf{q}_{b(k)}^{b(k-1)} = \begin{bmatrix} \cos \|\mathbf{0.5}\boldsymbol{\Phi}_k\| \\ \frac{\sin \|\mathbf{0.5}\boldsymbol{\Phi}_k\|}{\|\boldsymbol{\Phi}_k\|} \boldsymbol{\Phi}_k \end{bmatrix} \quad (3.18)$$

其中 $\boldsymbol{\Phi}_k$ 是 t_{k-1} 时刻的 b 系转动到 t_k 时刻的 b 系所对应的等效旋转矢量。等效

旋转矢量 Φ 的微分方程可以简化成如下(秦永元, 2006; Bortz, 1970):

$$\begin{aligned}\dot{\Phi}_k &\approx \omega_{ib}^b + \frac{1}{2}\Phi_k \times \omega_{ib}^b + \frac{1}{12}\Phi_k \times (\Phi_k \times \omega_{ib}^b) \\ &\approx \omega_{ib}^b + \frac{1}{2}\Delta\theta(t) \times \omega_{ib}^b\end{aligned}\quad (3.19)$$

式中,

$$\Delta\theta(t) = \int_{t_{k-1}}^{t_k} \omega_{ib}^b dt \quad (3.20)$$

根据比力和角速度的双子样假设, 对(3.19)积分得到(Savage,2012):

$$\begin{aligned}\Phi_k &= \int_{t_{k-1}}^{t_k} \left[\omega_{ib}^b + \frac{1}{2}\Delta\theta(t) \times \omega_{ib}^b \right] dt \\ &\approx \Delta\theta_k + \frac{1}{12}\Delta\theta_{k-1} \times \Delta\theta_k\end{aligned}\quad (3.21)$$

其中, $\frac{1}{12}\Delta\theta_{k-1} \times \Delta\theta_k$ 是二阶圆锥误差项, n 系从 t_{k-1} 时刻到 t_k 时刻的转动四元数为:

$$\mathbf{q}_{n(k-1)}^{n(k)} = \begin{bmatrix} \cos\|0.5\zeta_k\| \\ -\frac{\sin\|0.5\zeta_k\|}{\|\zeta_k\|} \zeta_k \end{bmatrix} \quad (3.22)$$

式中 ζ_k 的定义参考式(3.10)。由于位置更新已经完成, 中间位置可通过内插求解。高程方向采用 $h_{k-1/2} = (h_{k-1} + h_k)/2$; 而中间时刻的经纬分量则需要采用内插等效旋转矢量的方法来获得:

$$\mathbf{q}_{\delta\theta} = (\mathbf{q}_{n(k-1)}^{e(k-1)})^{-1} \mathbf{q}_{n(k)}^{e(k)} \quad (3.23)$$

式中 $(\mathbf{q}_{n(k-1)}^{e(k-1)})^{-1}$ 是四元数 $\mathbf{q}_{n(k-1)}^{e(k-1)}$ 的求逆运算。由上式可计算出旋转矢量 $\delta\theta$, 取该旋转矢量的一半 $0.5\delta\theta$, 计算出对应的四元数 $\mathbf{q}_{0.5\delta\theta}$, 中间时刻的位置四元数则可得:

$$\mathbf{q}_{n(k-1/2)}^{e(k-1/2)} = \mathbf{q}_{n(k-1)}^{e(k-1)} \mathbf{q}_{0.5\delta\theta} \quad (3.24)$$

从位置四元数 $\mathbf{q}_{n(k-1/2)}^{e(k-1/2)}$ 中可以解算出中间时刻的纬度和经度。

在计算过程中, 由于计算误差等的影响, 四元数 \mathbf{q}_b^n 会逐渐失去规范化特性, 即四元数的模不等于 1, 所以在每次 \mathbf{q}_b^n 得到更新后或者更新若干次后需要对其进行规范化处理, 如下:

$$q_i = \frac{\hat{q}_i}{\sqrt{\hat{q}_0^2 + \hat{q}_1^2 + \hat{q}_2^2 + \hat{q}_3^2}}, \quad i = 0, 1, 2, 3 \quad (3.25)$$

式中 q_i 和 \hat{q}_i 分别表示规范化处理后和处理前四元数第 i 个元素。

3.5 LiDAR/INS 组合导航算法

图 3-2 描述了本文中 LiDAR/INS 组合导航算法的架构。IMU 输出的陀螺仪和加速度计信息通过 3.3 节中的机械编排算法计算得出位置和姿态信息，不仅为 LiDAR 点云匹配算法提供搜索初值，还与 LiDAR 匹配搜索后的位置和姿态信息进行卡尔曼滤波组合，得到的组合后的位置误差和姿态误差信息用于更新 INS 机械编排的位置、姿态，估计出的惯导陀螺零偏和加速度计零偏用于补偿传感器参数误差。

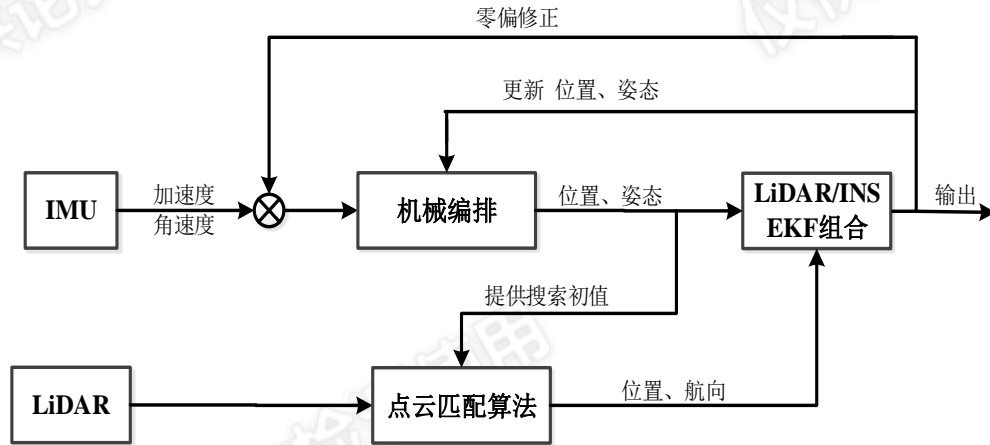


图 3-3 LiDAR/INS 组合导航算法结构图

3.5.1 INS 误差状态模型

受惯性传感器误差、重力误差、初始导航状态误差和机械编排算法计算误差等影响，导航更新获得的导航参数均含有误差。本节通过惯性导航的误差微分方程，来描述 INS 不同误差随时间的变化规律。所采用的方法是基于惯导微分方程进行误差扰动分析或者偏微分取至一阶近似，建立详细的 INS 连续时间误差微分方程，下面将具体给出位置、速度、姿态的微分方程，作为后续的误差传递分析的基础。

位置误差方程：

$$\delta \dot{\phi} = -\frac{v_N}{R_M + h} \delta h + \frac{1}{R_M + h} \delta v_N \quad (3.26)$$

$$\delta \dot{\lambda} = \frac{v_E \sin \phi}{(R_N + h)^2 \cos^2 \phi} \delta \phi - \frac{v_E}{(R_N + h)^2 \cos \phi} \delta h + \frac{1}{(R_N + h) \cos \phi} \delta v_E \quad (3.27)$$

$$\delta \dot{h} = -\delta v_D \quad (3.28)$$

速度误差方程:

$$\begin{aligned} \delta \dot{v}_N = & - \left(2v_E \omega_e \cos \phi + \frac{v_E^2}{(R_N + h) \cos^2 \phi} \right) \delta \phi - \left(\frac{v_N v_D}{(R_M + h)^2} - \frac{v_E^2 \tan \phi}{(R_N + h)^2} \right) \delta h + \frac{v_D}{R_M + h} \delta v_N \\ & - \left(2\omega_e \sin \phi + \frac{2v_E \tan \phi}{R_N + h} \right) \delta v_E + \frac{v_N}{R_M + h} \delta v_D - f_D^n \phi_E + f_E^n \phi_D + \delta f_x^n \end{aligned} \quad (3.29)$$

$$\begin{aligned} \delta \dot{v}_E = & \left(2\omega_e (v_N \cos \phi - v_D \sin \phi) + \frac{v_N v_E}{(R_N + h) \cos^2 \phi} \right) \delta \phi - \left(\frac{v_E v_D}{(R_N + h)^2} + \frac{v_N v_E \tan L}{(R_N + h)^2} \right) \delta h \\ & + \left(2\omega_e \sin \phi + \frac{v_E \tan \phi}{R_N + h} \right) \delta v_N + \frac{v_D + v_N \tan \phi}{R_N + h} \delta v_E \\ & + \left(2\omega_e \cos L + \frac{v_E}{R_N + h} \right) \delta v_D + f_D^n \phi_N - f_N^n \phi_D + \delta f_y^n \end{aligned} \quad (3.30)$$

$$\begin{aligned} \delta \dot{v}_D = & 2v_E \omega_e \sin \phi \cdot \delta \phi + \left(\frac{v_E^2}{(R_N + h)^2} + \frac{v_N^2}{(R_M + h)^2} - \frac{2g}{\sqrt{R_M R_N + h}} \right) \delta h - \frac{2v_N}{R_M + h} \\ & - \left(2\omega_e \cos L + \frac{2v_E}{R_N + h} \right) \delta v_E - f_E^n \phi_N + f_N^n \phi_E + \delta f_z^n \end{aligned} \quad (3.31)$$

姿态误差方程:

$$\begin{aligned} \dot{\phi}_N = & - \left(\omega_e \sin \phi + \frac{v_E \tan \phi}{R_N + h} \right) \phi_E + \frac{v_N}{R_M + h} \phi_D - \omega_e \sin \phi \cdot \delta \phi \\ & - \frac{v_E}{(R_N + h)^2} \delta h - \delta \omega_{ibx}^n \end{aligned} \quad (3.32)$$

$$\begin{aligned} \dot{\phi}_E = & \left(\omega_e \sin \phi + \frac{v_E \tan \phi}{R_N + h} \right) \phi_N + \left(\omega_e \cos L + \frac{v_E}{R_N + h} \right) \phi_D - \frac{1}{R_M + h} \delta v_N \\ & - \frac{v_N}{(R_M + h)^2} \delta h - \delta \omega_{iby}^n \end{aligned} \quad (3.33)$$

$$\begin{aligned} \dot{\phi}_D = & - \frac{v_N}{R_M + h} \phi_N - \left(\omega_e \cos \phi + \frac{v_E}{R_N + h} \right) \phi_E - \frac{\tan \phi}{R_N + h} \delta v_E \\ & - \left(\omega_e \cos \phi + \frac{v_E}{(R_N + h) \cos^2 \phi} \right) \delta \phi + \frac{v_E \tan \phi}{(R_N + h)^2} \delta h - \delta \omega_{ibz}^n \end{aligned} \quad (3.34)$$

式中, $\delta \mathbf{r}^n = [\delta \phi \quad \delta \lambda \quad \delta h]^T$ 表示位置的纬度、经度和高程误差, $\delta \mathbf{v}^n = [\delta v_N \quad \delta v_E \quad \delta v_D]^T$ 表示 n 系下北向、东向和地轴向速度误差, $\phi = [\phi_N \quad \phi_E \quad \phi_D]^T$ 表示 n 系下的姿态误差, $\mathbf{f}^n = [f_N^n \quad f_E^n \quad f_D^n]^T$ 表示加速度计输出在 n 系 (NED) 下的投影, $\delta \mathbf{f}^n = [\delta f_x^n \quad \delta f_y^n \quad \delta f_z^n]^T$ 表示加速度计三轴输出的误差在 n 系的投影, $\delta \omega_{ib}^n = [\delta \omega_{ibx}^n \quad \delta \omega_{iby}^n \quad \delta \omega_{ibz}^n]^T$ 表示陀螺三轴输出误差在 n

系的投影。

以上建立了完整的惯性导航误差动态方程, 基于该方程可以进行纯惯导误差传递分析和建立组合导航 Kalman 滤波状态方程。在进行 LiDAR 搜索匹配时, 准确的搜索初值非常重要, 惯性传感器的残余误差是影响给定初值精度的重要误差源, 需要在组合导航滤波器中对重要的惯性传感器误差参数进行估计和补偿, 主要是加速度计和陀螺的零偏误差。常用的做法是将上述误差模型化, 并将其推广到系统状态量中进行在线估计。

3.5.2 基于 15 维状态向量的卡尔曼滤波算法

卡尔曼滤波算法作为一种最重要的最优估计理论被广泛应用于各个领域, 在组合导航系统的设计方面运用的尤其成功。算法设计的重点在于如何合理地构建系统的状态方程和观测方程。首先选取系统状态量, 为保证系统线性化, 一般采用误差状态模型。改进后的组合导航滤波器状态向量由三个方向的导航状态误差和 IMU 传感器零偏组成, 共 15 维。对于中低精度的惯导系统, 由于 IMU 比例因子误差相比零偏误差对组合导航精度的影响更小, 因此这里不实时估计 IMU 比例因子误差, 可以减小状态向量位数和算法的复杂性。为了减少单位转换对精度的损失, 改进后的组合算法将 x、y 轴的位置误差单位均设成米。

误差状态向量定义为:

$$\mathbf{x}(t) = [(\delta \mathbf{r}_{INS}^n)^T (\delta \mathbf{v}_{INS}^n)^T \boldsymbol{\phi}^T \mathbf{b}_g^T \mathbf{b}_a^T] \quad (3.35)$$

式中, $\delta \mathbf{r}_{INS}^n$ 为惯导的三轴位置误差向量, 分别表示北东地三个方向上的位置误差; $\delta \mathbf{v}_{INS}^n$ 为惯导的三轴速度误差向量, 三个向量分别表示北东地三个方向上的速度误差; $\boldsymbol{\phi}$ 为三维姿态误差向量; \mathbf{b}_g 为三轴陀螺零偏向量; \mathbf{b}_a 为三轴加速度计零偏向量。将零偏误差建模为一阶高斯马尔可夫过程, 其微分方程为:

$$\begin{cases} \dot{\mathbf{b}}_g(t) = -\frac{1}{T_{gb}} \mathbf{b}_g(t) + \mathbf{w}_{gb}(t) \\ \dot{\mathbf{b}}_a(t) = -\frac{1}{T_{ab}} \mathbf{b}_a(t) + \mathbf{w}_{ab}(t) \end{cases} \quad (3.36)$$

LiDAR/INS 组合导航卡尔曼滤波系统的连续时间误差状态方程为:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \quad (3.37)$$

式中, \mathbf{x} 为状态向量, $\mathbf{x}(t)$ 为 t 时刻的系统状态。 $\mathbf{F}(t)$ 为系统动态矩阵; $\mathbf{G}(t)$ 为系统噪声驱动矩阵; $\mathbf{w}(t)$ 为系统激励噪声过程。

对状态方程进行离散化处理得到：

$$\mathbf{x}_k = \Phi_{k/k-1} \mathbf{x}_{k-1} + \mathbf{G}_{k-1} \mathbf{w}_{k-1} \quad (3.38)$$

其中， \mathbf{x}_k 分别表示 k 时刻的系统状态向量， $\Phi_{k/k-1}$ 为离散后的状态转移矩阵， \mathbf{G}_{k-1} 为系统噪声驱动阵， \mathbf{w}_{k-1} 为状态的过程噪声向量。

$\Phi_{k/k-1}$ 和 \mathbf{w}_k 的表达式分别如下：

$$\Phi_{k/k-1} = \exp(\mathbf{F}(t_k)\Delta t) \approx \mathbf{I} + \mathbf{F}(t_k)\Delta t \quad (3.39)$$

$$\begin{cases} E[\mathbf{w}_k] = 0 \\ E[\mathbf{w}_k \mathbf{w}_j^T] = \mathbf{Q}_k \delta_{kj} \end{cases} \quad (3.40)$$

式(3.40)中，

$$\delta_{kj} = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases} \quad (3.41)$$

$$\mathbf{Q}_k \approx \Phi_k \mathbf{G}(t_k) \mathbf{Q} \mathbf{G}(t_k)^T \Phi_k^T \Delta t \quad (3.42)$$

$$\mathbf{Q} = \text{diag}(\delta_g^2, \delta_a^2) \quad (3.43)$$

\mathbf{Q} 为驱动白噪声 $\mathbf{w}(t)$ 的方差强度； δ_g, δ_a 分别为陀螺和加速度计的三轴标准差向量。

LiDAR 搜索匹配算法能给出移动机器人的位置和航向信息，LiDAR 给出的位置可以模型化为真实位置和位置噪声之和，给出的航向可以模型化为真实航向和航向噪声之和，这里位置噪声和航向噪声一般简化为高斯白噪声；INS 输出的位置和姿态可以建模为位置真值和误差之和。将两系统的位置和姿态求差作为松组合 Kalman 滤波器的观测量，建立观测方程为：

$$\mathbf{z}_k = \begin{bmatrix} \mathbf{r}_{IMU}^n - \mathbf{r}_{LiDAR}^n \\ \phi_{IMU}^n - \phi_{LiDAR}^n \end{bmatrix} = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \quad (3.44)$$

式中，观测向量 \mathbf{z}_k 为四维向量，由三轴位置误差向量和一维航向向量组成。之所以没有像 Tang(2015) 中加上俯仰和横滚角，是因为室内环境中的地面大部分只能是近似平坦，而并不是绝对平坦，如果强制将来自 LiDAR 的俯仰角和横滚角一直设为 0，会在计算过程中影响位置误差的估计，因此本节将姿态的观测信息改为只采用航向信息，在计算过程中估计俯仰角和横滚角，但是 LiDAR 观测量中的 z 轴位置仍设为 0。

式中， \mathbf{H}_k 为 4*15 维的量测矩阵； \mathbf{v}_k 是测量噪声向量。由于本文中认为 \mathbf{b} 系和 \mathbf{v} 系是同一坐标系，且不考虑安装角误差和杆臂对整个系统的影响，那么根据 Shin(2005) 论文中扩展卡尔曼滤波的量测模型，本文中的量测矩阵 \mathbf{H}_k 可以简化表示为：

$$\mathbf{H}_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.45)$$

\mathbf{v}_k 是测量噪声向量, 满足 $\mathbf{v}_k \sim N(\mathbf{0}, \mathbf{R}_k)$ 。

$$\mathbf{R}_k = \text{diag}(\delta_r^2, \delta_\phi^2) \quad (3.46)$$

\mathbf{R}_k 为量测噪声方差阵, δ_r, δ_ϕ 是位置误差和姿态误差, 根据 LiDAR 的性能和 LiDAR 搜索匹配方法的精度给出。

在一个卡尔曼滤波周期中, 卡尔曼滤波的信息更新过程可以分为预测过程和更新过程。一步预测的状态及其协方差阵为:

$$\hat{\mathbf{x}}_{k/k-1} = \Phi_{k/k-1} \hat{\mathbf{x}}_{k-1} \quad (3.47)$$

$$\mathbf{P}_{k/k-1} = \Phi_{k/k-1} \mathbf{P}_{k-1} \Phi_{k/k-1}^T + \mathbf{G}_{k-1} \mathbf{Q}_{k-1} \mathbf{G}_{k-1}^T \quad (3.48)$$

在更新过程中首先计算增益矩阵 \mathbf{K}_k 。然后更新系统状态及误差协方差阵。

当 \mathbf{Q}_k 为非负定, \mathbf{R}_k 为正定阵时, \mathbf{x}_k 的最优估计 $\hat{\mathbf{x}}_k$ 可按下述方求解:

$$\mathbf{K}_k = \mathbf{P}_{k/k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (3.49)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k/k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k/k-1}) \quad (3.50)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k/k-1} \quad (3.51)$$

式(3.47)-(3.51)就是离散系统中卡尔曼滤波的基本方程, 只要给定初值 $\hat{\mathbf{x}}_0$ 和 \mathbf{P}_0 , 根据 k 时刻的量测就可以递推计算任意时刻 k 的状态估计 $\hat{\mathbf{x}}_k$ 。然后再将状态误差 $\hat{\mathbf{x}}_k$ 反馈到 INS 机械编排中去修正最后的导航输出量, 反馈后的结果也将作为下一时刻的 LiDAR 搜索匹配初值。

但要正确实现 LiDAR/INS 卡尔曼滤波算法, 除了前面所提到的之外, 还需要注意以下几点:

(1) 在构成量测方程时 LiDAR 和 INS 得到的位姿结果的参考坐标系要统一。本文的 2.3 节详细描述了 LiDAR 的数学模型和定位过程, LiDAR 搜索匹配完成后得到的相对位置是基于世界坐标系下的, 经过高斯投影后得到的坐标系是东北天, 而 INS 机械编排后得到的位置和姿态是基于 n 系下的, 是北东地坐标系, 所以需要转换成同一参考坐标系下。LiDAR 匹配后计算出来的姿态是 LiDAR 的局部坐标系前左上相对于东北天坐标系之间的旋转矢量, 而 INS 得到的姿态是 b 系相对于 n 系的旋转矢量, 所以还需将 LiDAR 计算出的姿态乘以 b 系相对于前左上坐标系的旋转矢量和东北天坐标系相对于北东地坐标系之间的旋转矢量。

(2) 初始位姿要尽可能给地较为准确。MEMS 级别的惯导由于自身设备的噪

声过大，无法进行初始对准，所以在初始化时便要给出较为准确的航向和位置信息。

(3) \mathbf{P} 阵是状态向量的误差协方差阵，可表示为：

$$\mathbf{P} = E\{\mathbf{x}, \mathbf{x}^T\} = \mathbf{diag}(\delta_x^2, \delta_y^2, \delta_h^2, \delta_{vx}^2, \delta_{vy}^2, \delta_{vz}^2, \delta_{\varepsilon x}^2, \delta_{\varepsilon y}^2, \delta_{\varepsilon z}^2) \quad (3.52)$$

\mathbf{P} 阵的初始值 \mathbf{P}_0 可根据以下原则给定： $\delta_x, \delta_y, \delta_h$ 是根据初始位置的精度给定，本文中给出的是厘米级；由于本文的移动机器人系统是从静止状态开始实验，所以 $\delta_{vx}, \delta_{vy}, \delta_{vz}$ 都是较小的值，本文给出的是 0.0001m/s(董绪荣, 1998)； $\delta_{\varepsilon x}, \delta_{\varepsilon y}, \delta_{\varepsilon z}$ 是初始姿态的精度，根据经验，本文给出的分别是 1° 、 1° 和 5° 。

4 组合导航算法近实时性改进

4.1 引言

在本文的 LiDAR/INS 组合导航系统中，尽管改进后的高斯-牛顿搜索方法和组合导航算法能够大大减少 LiDAR 扫描匹配的时间，但 LiDAR 扫描匹配的耗时仍然超过了 IMU 的采样间隔。如果使用改进前的标准 Kalman 滤波算法，即当 LiDAR 观测数据接收后，立即开始进行 LiDAR 数据解算，则后续得到的 IMU 观测数据将会被缓存起来不处理，一直等待到 Kalman 滤波组合更新完成后才能进行下一个 IMU 历元数据的解算，IMU 数据并不能及时处理，而且更新完成后得到的组合导航结果也并不是当前时刻的结果，这就造成了组合导航结果输出的阻塞和延迟，影响系统的实时性能。针对这个问题，本节提出了一种基于状态转移的实时性解决办法，即一步状态转移法 (One-Step-Error-State-Transition, OSEST)。这种状态转移的方法曾经应用在 GNSS/INS 组合系统中(章红平, 常乐等, 2016)，但还没有被应用在 LiDAR/INS 组合系统中。本文采用这种方法是因为该方法不仅计算量小，易于实现，而且可靠性高。本章将详细介绍该方法的具体实现过程和数学模型。但本文还没有把改进的实时性算法移植到嵌入式平台上，真正实现移动机器人平台在室内一边行走一边在所搭载的计算机上形成周围环境的地图，所以只能叫近实时处理算法。

4.2 OSEST 算法详细过程

第 4.1 节中提到 LiDAR 匹配算法耗时过长是实时性算法的一个主要矛盾，为了解决由于 LiDAR 匹配算法耗时过长所带来的数据延迟、等待等问题，本文将 LiDAR 扫描匹配、组合更新和 INS 机械编排、Kalman 预测分解成具有优先级别的两个线程独立运行。处理时序示意如图 4-1 所示，在 LiDAR 的采样时刻 t_k ，开始在 LiDAR 线程中进行 LiDAR 扫描匹配算法解算，并保存该时刻的预测误差协方差矩阵和机械编排解算的导航状态；同时在 IMU 线程中，一直保持着 INS 机械编排和 Kalman 预测的计算。当 LiDAR 扫描匹配完成后，开始进行组合更新计算，但组合更新计算完成后的时刻已经不是 t_k 了，所以还需要将滞后计算出来的反映 LiDAR 采样时刻 t_k 的状态参数估计量 \mathbf{x}_{t_k} 以及状态参数协方差估计量 \mathbf{P}_{t_k} 正确转移到计算完成时刻（即当前时刻，图中为 t_j ），以修正当前时刻的 INS

机械编排误差，输出实时的组合导航结果。OSEST 算法就是将 t_k 时刻的状态参数估计量 \mathbf{x}_{t_k} 以及状态参数协方差估计量 \mathbf{P}_{t_k} 一步转移得到 t_j 时刻的状态参数估计量 \mathbf{x}_{t_j} 以及状态参数协方差估计量 \mathbf{P}_{t_j} ，4.3 节是具体推导该方法的过程。

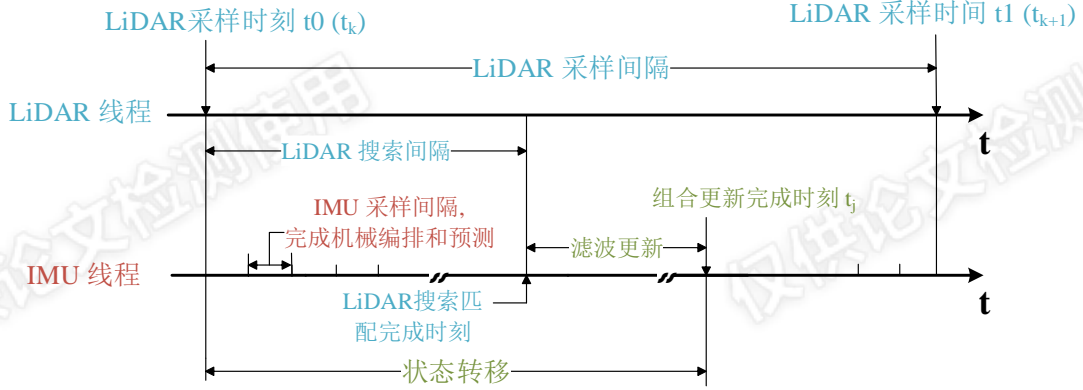


图 4-1 处理时序示意图

4.3 建立 OSEST 算法的数学模型

本节将根据 OSEST 的实现过程推导出其具体的数学公式。卡尔曼滤波中状态转移是根据 k 时刻的状态估计 j 时刻 ($j > k$) 的状态，即根据 k 个观测向量 $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k$ 对状态向量 \mathbf{x}_j 作线性最小方差估计 $\hat{\mathbf{x}}_{j/k}$ (Shin, 2005):

$$\hat{\mathbf{x}}_{j/k} = \mathbf{E}[\mathbf{x}_j | \mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k] \quad (4.1)$$

考虑到 $\Phi_{k+1/k-1} = \Phi_{k+1/k} \Phi_{k/k-1}$ ，则状态向量 \mathbf{x}_j 可由等式(3.28)推导得出:

$$\mathbf{x}_j = \Phi_{j/k} \mathbf{x}_k + \sum_{i=k+1}^j \Phi_{j/i} \mathbf{G}_{i/i-1} \mathbf{w}_{i-1}, \quad k < j \quad (4.2)$$

将式(4.2)代入式(4.1)中，得到:

$$\begin{aligned} \hat{\mathbf{x}}_{j/k} &= \mathbf{E}[\mathbf{x}_j | \mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k] \\ &= \mathbf{E}[(\Phi_{j/k} \mathbf{x}_k + \sum_{i=k+1}^j \Phi_{j/i} \mathbf{G}_{i/i-1} \mathbf{w}_{i-1}) | \mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k] \\ &= \mathbf{E}[\Phi_{j/k} \mathbf{x}_k | \mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k] + \sum_{i=k+1}^j \Phi_{j/i} \mathbf{G}_{i/i-1} \mathbf{E}[\mathbf{w}_{i-1} | \mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k] \end{aligned} \quad (4.3)$$

根据等式(3.28)和(3.34)可知， \mathbf{w}_{i-1} 只对 \mathbf{x}_{i-1} 有影响，与观测值 $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k$ 不相关。而且，由于 \mathbf{w}_{i-1} 为白噪声，是零均值的随机变量序列，与时间无关，因此等式(4.3)可以简化为:

$$\hat{\mathbf{x}}_{j/k} = \Phi_{j/k} \mathbf{E}[\mathbf{x}_k | \mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k] = \Phi_{j/k} \hat{\mathbf{x}}_k \quad (4.4)$$

其中， $\hat{\mathbf{x}}_k$ 为状态向量 \mathbf{x}_k 更新后的估计量。

定义 $\tilde{\mathbf{x}}_{j/k}$ 为 \mathbf{x}_j 和 $\hat{\mathbf{x}}_{j/k}$ 的误差，如下所示:

$$\tilde{\mathbf{x}}_{j/k} = \mathbf{x}_j - \hat{\mathbf{x}}_{j/k}$$

$$\begin{aligned}
&= \Phi_{j/k} \mathbf{x}_k + \sum_{i=k+1}^j \Phi_{j/i} \mathbf{G}_{i/i-1} \mathbf{w}_{i-1} - \Phi_{j/k} \hat{\mathbf{x}}_k \\
&= \Phi_{j/k} \tilde{\mathbf{x}}_k + \sum_{i=k+1}^j \Phi_{j/i} \mathbf{G}_{i/i-1} \mathbf{w}_{i-1}
\end{aligned} \tag{4.5}$$

由等式(3.30)可得出 \mathbf{w}_{i-1} 的协方差为:

$$\mathbf{E}[\mathbf{w}_{i-1} \mathbf{w}_{i-1}^T] = \mathbf{Q}_{i-1} \tag{4.6}$$

则 $\tilde{\mathbf{x}}_{j/k}$ 的协方差矩阵可表示为:

$$\begin{aligned}
\mathbf{P}_{j/k}^- &= \mathbf{E}[\tilde{\mathbf{x}}_{j/k} \tilde{\mathbf{x}}_{j/k}^T] \\
&= \Phi_{j/k} \mathbf{E}[\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^T] \Phi_{j/k}^T + \sum_{i=k+1}^j \Phi_{j/i} \mathbf{G}_{i/i-1} \mathbf{E}[\mathbf{w}_{i-1} \mathbf{w}_{i-1}^T] \mathbf{G}_{i/i-1}^T \Phi_{j/i}^T \\
&= \Phi_{j/k} \hat{\mathbf{P}}_k \Phi_{j/k}^T + \sum_{i=k+1}^j \Phi_{j/i} \mathbf{G}_{i/i-1} \mathbf{Q}_{i-1} \mathbf{G}_{i/i-1}^T \Phi_{j/i}^T
\end{aligned} \tag{4.7}$$

由于 $\mathbf{G}_{i/i-1}$ 是正定矩阵, 因此满足 $\mathbf{G}_{i/i-1} \mathbf{Q}_{i-1} \mathbf{G}_{i/i-1}^T = \mathbf{Q}_{i-1}$.

定义 $\mathbf{M}_{j,k+1} = \sum_{i=k+1}^j \Phi_{j/i} \mathbf{Q}_{i-1} \Phi_{j/i}^T$, 可以得到:

$$\begin{aligned}
\mathbf{M}_{k+1,k+1} &= \mathbf{G}_{k+1/k} \mathbf{Q}_k \mathbf{G}_{k+1/k}^T = \mathbf{Q}_k \\
\mathbf{M}_{j+1,k+1} &= \mathbf{Q}_j + \Phi_{j+1/j} \mathbf{M}_{j,k+1} \Phi_{j+1/j}^T, \quad (j > k+1)
\end{aligned} \tag{4.8}$$

综上所述, 状态转移公式可以总结为:

$$\begin{aligned}
\hat{\mathbf{x}}_{j/k} &= \Phi_{j/k} \hat{\mathbf{x}}_k \\
\mathbf{P}_{j/k}^- &= \Phi_{j/k} \hat{\mathbf{P}}_k \Phi_{j/k}^T + \mathbf{M}_{j,k+1}
\end{aligned} \tag{4.9}$$

由式(4.9)可以看出, 状态向量和它的协方差阵可以由累积的状态转移矩阵及状态噪声阵得到, 把 LiDAR 采样时刻看作是 k 时刻, LiDAR/INS 组合更新计算完成时刻看作是 j 时刻, 利用式(4.9)可以把 k 时刻的状态参数更新估计量以及状态参数协方差更新估计量转移到 j 时刻, 进而在更新计算完成时刻进行组合更新。

5 实测实验与分析

5.1 引言

前面 2、3、4 节系统地描述了本文对整个 LiDAR/INS 组合系统的算法方面的改进。文章为了验证前面章节中提出的改进算法的性能，搭建了一套 LiDAR/INS 组合的室内 SLAM 系统的软硬件平台进行仿真和实验验证。本章第 2 节详细介绍了文本搭建的软硬件平台；第 3 节分别计算了第二章提出的 LiDAR 高斯-牛顿匹配算法和改进前的 IMLE 算法的匹配耗时，并进行对比分析，验证了高斯-牛顿算法的高效性；第 4 节对比分析了 LiDAR/INS 组合的定位制图结果、仅靠 INS 姿态辅助的 LiDAR 定位制图结果以及单一 LiDAR 匹配的定位制图结果，证明了本文提出的 LiDAR/INS 组合算法的可靠性；第 5 节通过对比后处理算法和近实时算法的输出延迟，验证了第 4 章所提的近实时算法的可行性。

5.2 室内移动测图的 UGV 平台设计

本文将测量传感器 LiDAR 和惯导集成搭载在移动小车平台上构成整个系统的硬件平台，称作室内移动测图的 UGV(Unmanned Ground Vehicle)平台(简称 UGV 平台)，如图 5-1 所示。IMU、电源、所需的连接线全部被封装在盒子里面，把 LiDAR 固定安装在盒子的顶端，只在盒子的旁边留出接口，这样集成不仅方便安装和拆卸，而且让整个 UGV 平台看起来干净整洁。



图 5-1 组合系统的 UGV 硬件平台

所用的 LiDAR 和 IMU 传感器的性能参数如表 5-1 和表 5-2 所示。LiDAR 是整个室内 SLAM 系统中的核心传感器，承担着定位和制图两项工作，所选用的型号是 Hokuyo 公司生产的 UTM-30LX-EW。采样频率为 40Hz，扫描的角度范围为 270°，角度分辨率为 0.25°，能够探测到的距离范围为 0.1m-30m，测量精度为毫米级，属于中等精度的 LiDAR，LiDAR 直接将采集到的距离数据通过网口传送给计算机。所选用的 IMU 型号是 MTiG，属于 MEMS 级别的惯导，采样频率为 200Hz，角度随机游走和速度随机游走分别为 $3^\circ/\sqrt{h}$ 和 $0.12\text{m/s}/\sqrt{h}$ ，陀螺零偏 $200^\circ/h$ ，加速度计零偏为 200mGal ($1\text{Gal} = 1\text{cm/s}^2$)，MTiG 以增量形式输出陀螺和加速度计的原始测量值，用于后续的组合导航解算 (Xsens Technologies, 2014)。

表 5-1 LiDAR 主要性能指标

Hokuyo UTM-EX 型 LiDAR 参数

采样频率	40HZ
测量率	43200 个点/每秒
波长	905nm
扫描距离范围	0.1m-30m
测量精度	0.1~10m: $\pm 30\text{mm}$
	10m~30m: $\pm 50\text{mm}$
测量噪声	10mm
扫描角度范围	270°
角度分辨率	0.25°
接口	Ethernet 100BASE-TX
输出	同步脉冲
重量	210g

表 5-2 MTiG 的主要性能指标

采样频率	200Hz
零偏	200 degree/h
加速度计零偏	2000 mGal (1 Gal=1 cm/s ²)
角度随机游走	$3^\circ/\sqrt{h}$
速度随机游走	$0.12\text{m/s}/\sqrt{h}$

重量	55g
----	-----

图 5-2 是基于 Visual Studio 2012 集成开发环境和 Qt SDK 开发的软件平台，在本文中用于数据后处理和定位制图工作，取名叫 NAVIS(Tang,2014)。本文的工作是改进 NAVIS 中的算法部分。图 5-2 显示了 NAVIS 软件的 GUI 界面，它主要由资源管理窗口，地图显示窗口，实时信息窗口和控制工具栏组成。资源管理窗口存储着所有系统中所需的数据源，如概率地图，UGV 机器人，机器人的轨迹以及 LiDAR 和 IMU 的原始数据；地图显示主窗口显示出通过算法处理后得到的 UGV 定位和制图结果；机器人的位置姿态等信息将实时地显示在软件下方的实时信息窗口中；在软件上方的控制工具栏中，通过点击按钮可以执行相对应的数据处理或者仿真工作。软件右侧的 Camera Video 等窗口是为以后搭载视觉传感器准备的。

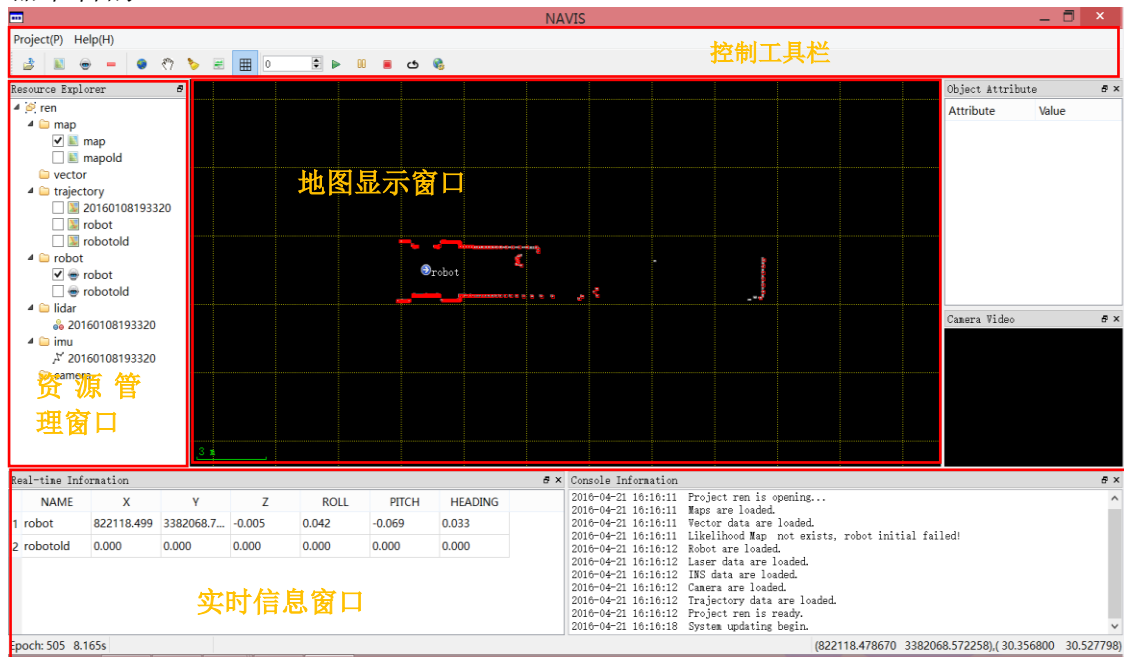
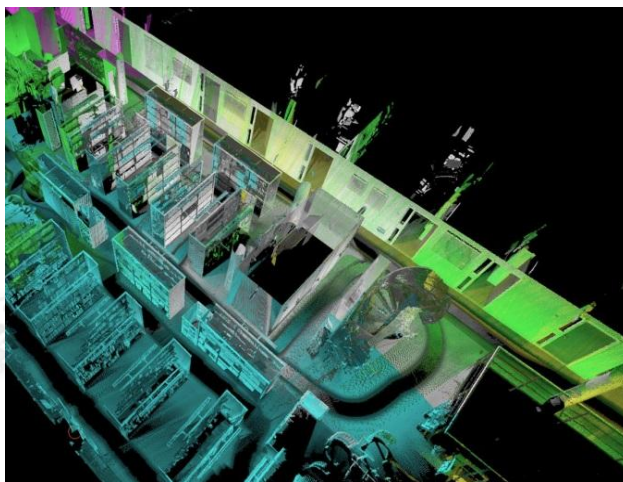


图 5-2 组合系统的 NAVIS 软件平台

本文后处理工作中的实验数据均是从 FGI 开放式图书馆中采集得到。图 5-4(a) 是图书馆的 3D 环境效果图，从中可以看到，馆内的特征信息非常丰富，有多个矩形的书架、柜子和门等，图 5-4 (b)显示了真实环境中部分书架的形状。这些书架、柜子以及门的边缘特征易于提取，对于 LiDAR 来说都是非常好的特征点。该图书馆占地面积 $28*12m^2$ ，不仅有长廊还有多处拐角，既有特征信息丰富的地方，也有特征不明显匹配效果不好的地方，可以满足本文实验中所需的场景。



(a)



(b)

图 5-4 (a) FGI 图书馆的 3D 效果图；(b) FGI 图书馆的部分真实环境

5.3 高斯-牛顿算法与 IMLE 算法的耗时性分析

本文采用基于高斯-牛顿迭代法 LiDAR 匹配搜索，主要是因为该算法能大大减少匹配耗时，这样不仅可以节省总制图时间，还为第 4 章所提出的近实时性改进算法提供了便利，即使不降低 LiDAR 的采样率，LiDAR 匹配算法仍然能在 25ms 的采样周期内处理完成。

为了验证 2.3 章所提出的高斯-牛顿算法的性能，本节将用高斯-牛顿算法的匹配搜索耗时与用 IMLE 算法的耗时进行了对比分析，如图 5-5 所示。每接收到一帧 LiDAR 点云数据，就要进行一次匹配搜索。由图 5-5 可看出，在整个运行

过程中，高斯-牛顿算法所需的匹配耗时都远远小于 IMLE 算法的匹配耗时。从表 5-3 的统计结果可以看到，利用高斯-牛顿算法的 LiDAR 数据平均匹配耗时为 10.89ms，最大值为 14.44ms，而利用 IMLE 算法的平均匹配耗时为 58.53ms，最大值高达 88.16ms。统计结果表明，高斯-牛顿算法使得 LiDAR 数据的平均匹配耗时比原来减小了约 4 倍，最大值减小了约 5 倍。因此，根据图 5-5 和表 5-3 的结果可以看出，本文所采样的高斯-牛顿算法能够使得 LiDAR 搜索匹配耗时大大减小，提高了系统的工作效率。

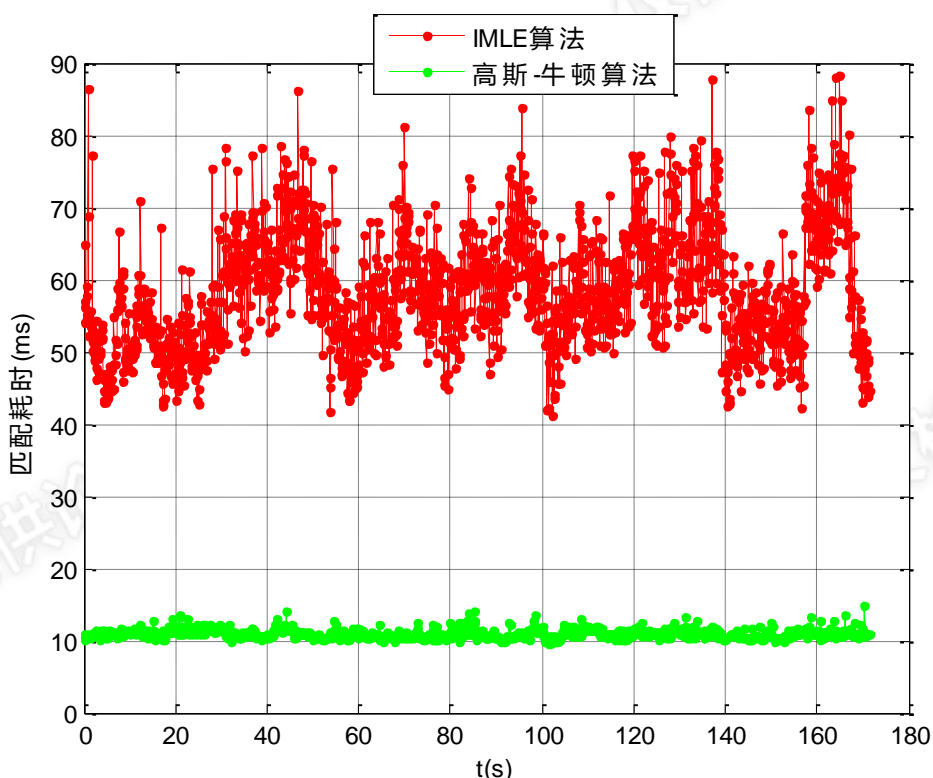


图 5-5 高斯-牛顿算法和 IMLE 算法匹配耗时对比

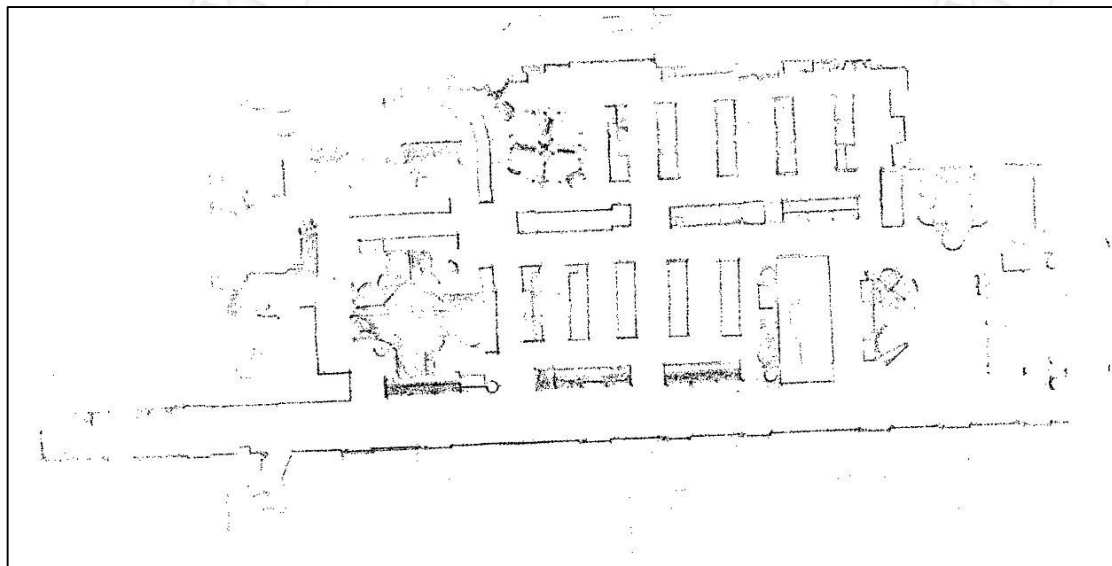
表 5-3 高斯-牛顿算法和 IMLE 算法匹配耗时统计表

	高斯-牛顿算法	IMLE 算法
平均值(ms)	10.89	58.53
最大值(ms)	14.44	88.16

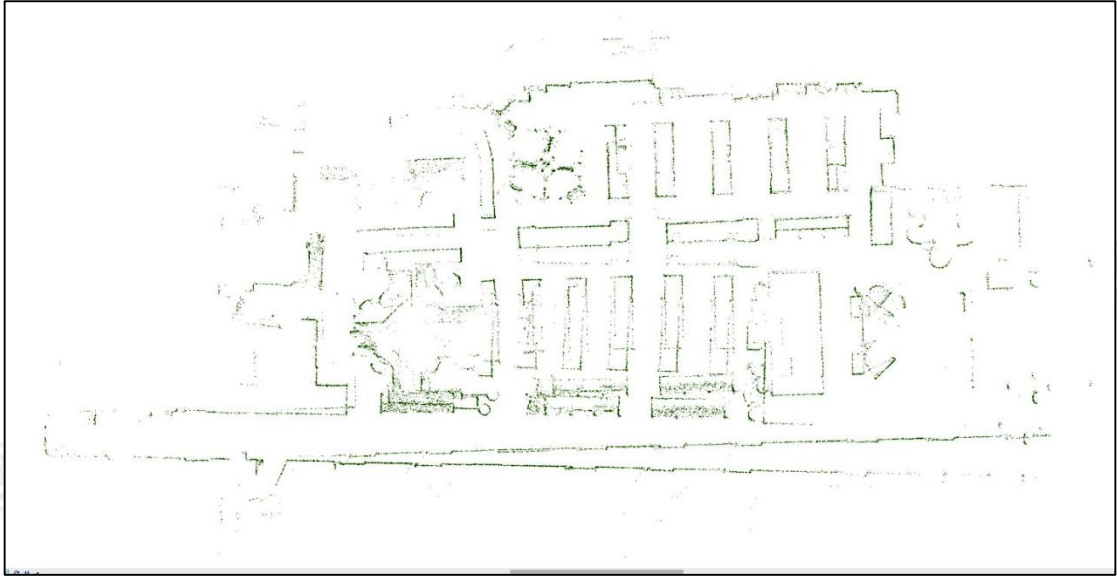
5.4 INS 辅助对高斯-牛顿匹配的影响分析

第 2.3.3 节提到, 搜索初值给定地准确与否决定着高斯-牛顿算法是否能正确迭代得到最优解。根据第 3 节的理论分析, LiDAR/INS 组合导航算法能够为 LiDAR 搜索匹配提供更为准确的初值, 因此, 本节将分别对有 INS 辅助和无 INS 辅助的系统进行实验, 并将制图的结果与高精度的 TLS(Terrestrial Laser Scanner, FARO Focus^{3D} 330X)的制图结果进行对比分析。

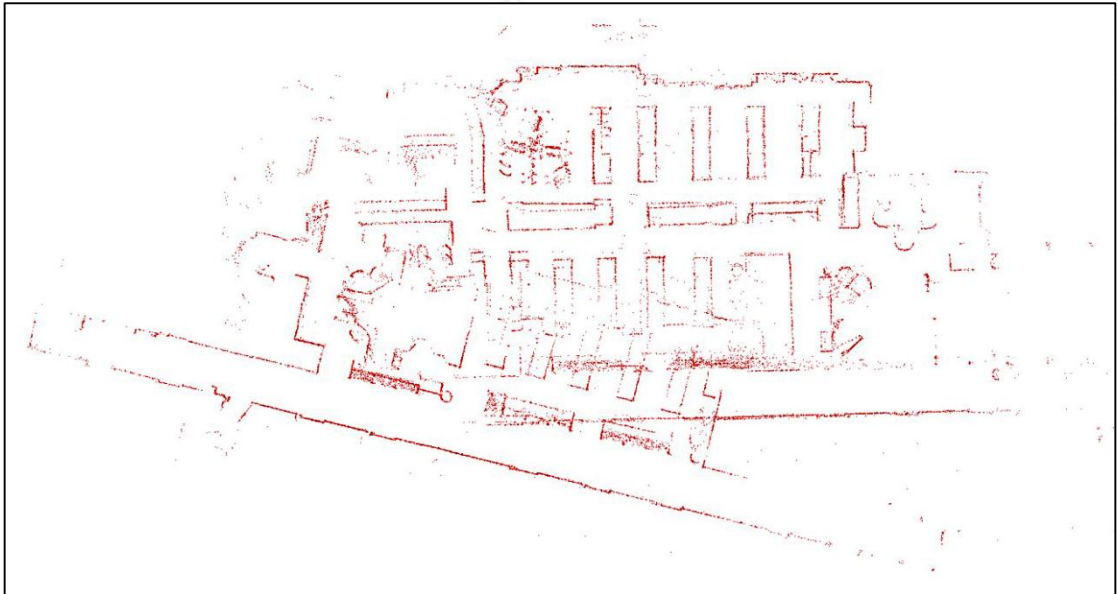
图 5-6(a)是利用 INS 位置和姿态信息与 LiDAR 进行组合, 将组合后的结果对 LiDAR 匹配进行辅助后的制图结果; 图 5-6(b)和 5-6(c)分别是仅靠惯导陀螺输出积分得到的姿态信息对 LiDAR 搜索提供初值和完全无 INS 辅助时的制图结果, 图 5-6(d)是 TLS 的制图结果。由此可以看出, 图 5-6(a)和 5-6(d)绘制出的形状基本一样, 虽然图 5-6(a)中物体的轮廓比 5-6(d)中对应物体的轮廓噪声更大, 这主要是由于本文所选用的 Hokuyo 的 LiDAR 精度不如 TLS 的精度高; 图 5-6(b)显示在开始一段时间绘制出的地图形状较为准确, 但长时间工作后便开始出现偏差; 而图 5-6(c)中显示的地图完全是混乱的。



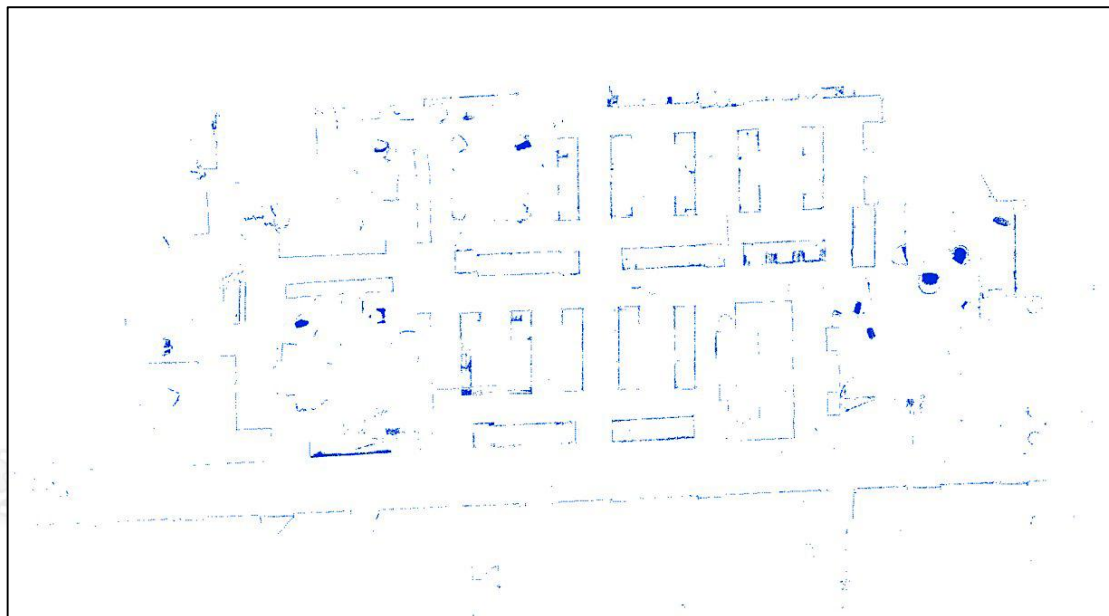
(a)



(b)



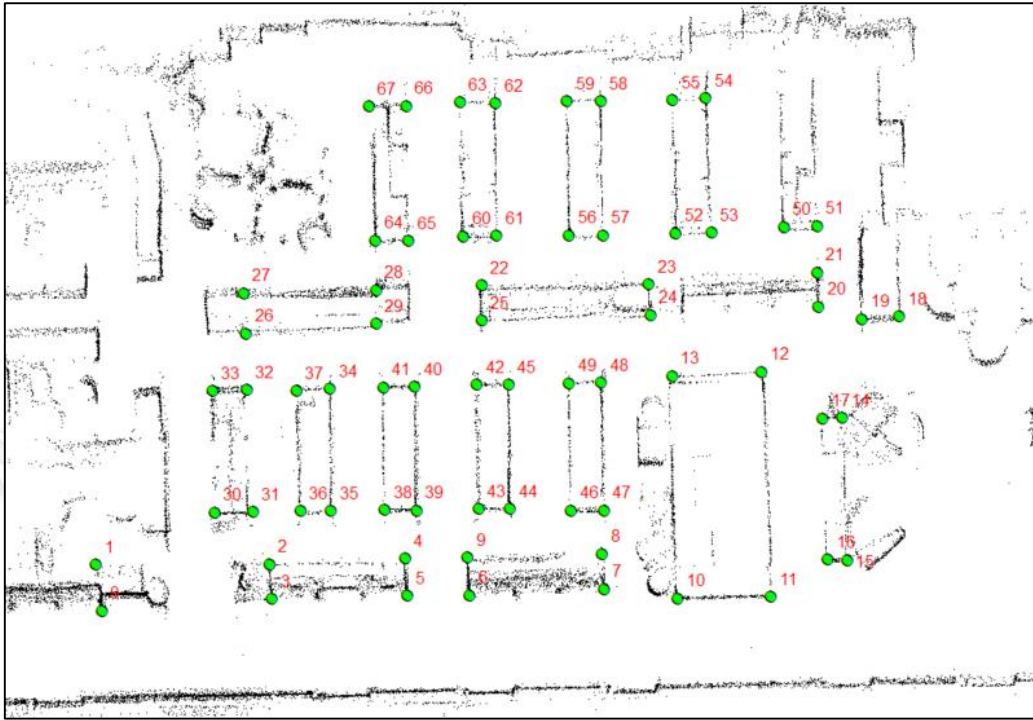
(c)



(d)

图 5-6 (a)INS 辅助 LiDAR 的制图结果; (b)仅利用 INS 姿态信息辅助 LIDAR 的制图结果;
(c)LiDAR 单独的制图结果; (d)TLS 绘制的参考地图

为了验证改进后 LiDAR/INS 组合算法的制图效果, 图 5-7(a)(b)是将图 5-6(a)(d)结果中比较清晰的一部分截取出来, 并选择了不能移动的书架拐角或和墙壁拐角作为主要特征点, 共计 67 个特征点, 选择 TLS 制图结果中特征点的坐标为参考真值, 通过统计出图(a)对应特征点的坐标相对于参考真值的 RMS 值, 得出改进后 LiDAR/INS 组合算法的制图精度。表 5-4 列出了统计结果, RMS 值为 0.0562m, 属于厘米级, 由于本文中地图的最小分辨率为 0.01m, 加上考虑到人工选取特征点的操作误差, 所以 LiDAR/INS 组合算法的制图精度是可以接受的。由于在室内 SLAM 中, 制图和定位是相辅相成的, LiDAR 扫描出的移动机器人平台到环境中障碍物的距离信息是一定的, 因此, 如果地图绘制正确, 那么说明移动机器人平台的定位结果也是准确的。



(a)



(b)

5-7 (a)部分 LiDAR/INS 组合制图结果及所选的特征点；(b)部分 TLS 制图结果及所选的特征点

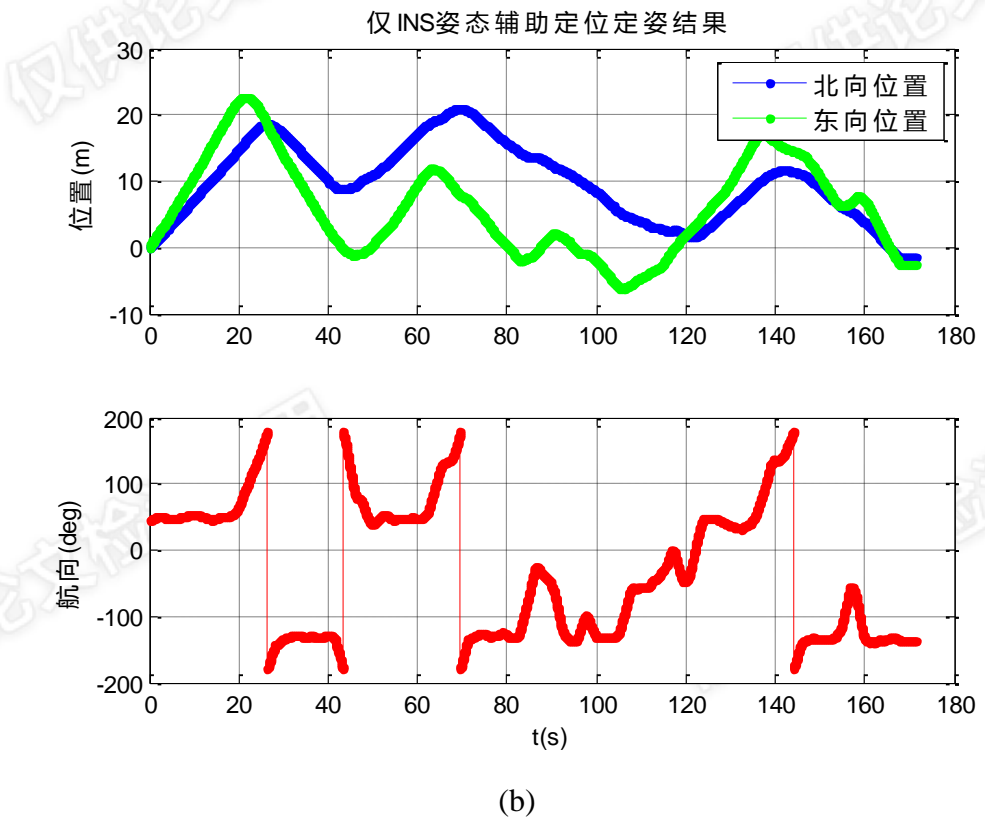
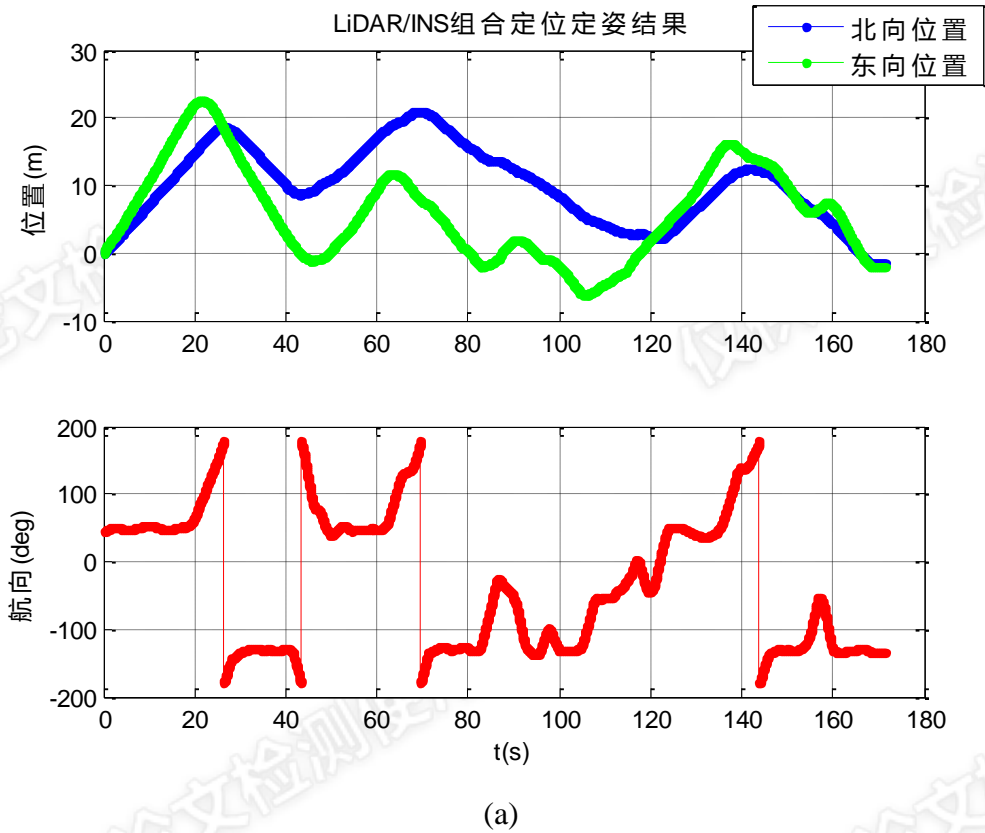
表5-4. 所选特征点的结果精度统计表

	Feature Points	RMS (m)
LiDAR/INS组合制图	67	0.0562

图 5-8(a)、(b)、(c)分别是 LiDAR/INS 组合定位定姿结果、仅靠 INS 姿态辅助 LiDAR 定位结果和单一 LiDAR 进行搜索匹配的定位结果。将移动机器人平台的初始位置设为原点，初始航向是与北向的夹角，为 45° 。图 5-9 和图 5-10 分别是 LiDAR/INS 组合定位与仅靠 INS 姿态辅助 LiDAR 定位结果之间的差异和 LiDAR/INS 组合定位与 LiDAR 单独匹配定位结果之间的差异。表 5-5 和表 5-6 是相应的差异统计结果。从图 5-9 的曲线中反映出位置、航向误差都是在大约 100s 以后突然开始增大，位置误差逐渐增加到了接近 1m 左右再开始缓慢下降，而姿态误差是快速增大到 3° 左右再缓慢上升。从表 5-5 的统计结果中可以看出，北向位置的平均差异为 0.201m，最大为 0.861m，RMS 为 0.342m；东向位置的平均差异为 0.170m，最大为 0.889m，RMS 为 0.300m；航向的平均差异为 1.350° ，最大为 4.314° ，RMS 为 2.050° 。在前 100s 期间，两者的差异非常小，因此反映在地图上，前 100s 两者的制图结果近似，从第 100s 后仅靠 INS 姿态辅助 LiDAR 定位的地图绘制开始出现偏差现象。

图 5-10 中的差异曲线表明，北向位置和东向位置在运行开始后 8s 左右，单一 LiDAR 匹配与 LiDAR/INS 组合的结果差异开始逐渐增大，表 5-6 显示北向位置平均差异为 4.741m，最大为 7.867m，RMS 为 5.002m，东向位置的平均差异为 6.439m，最大为 10.354m，RMS 为 6.734m，在室内环境中，位置误差如果达到了 5、6m，完全可以判定为定位结果错误。图 5-10 中的航向误差曲线表明，从开始到第 60s 期间，航向差异都比较小，基本维持在 1° 以下，到第 60s 后，航向误差开始增大，但从 60s 至 100s 期间，航向误差也都维持在 2° 以下，到第 100s 后，航向误差开始突然增大到 15° 左右。图 5-9 中也是从 100s 左右之后，位置、航向误差开始增大。从图 5-8(a)中 100s 之后的航向曲线可看出，这段时间移动机器人平台转弯次数较多，航向变化明显，可以表明，动态信息越丰富的情况下越能发挥 INS 的作用。表 5.5 中的航向误差统计结果显示，航向误差平均为 6.786° ，最大为 17.268° ，RMS 为 9.931° ，这个误差已经远远超出 LiDAR 的角度分辨率 0.25° 几十倍。因此，在没有 INS 辅助，仅靠 LiDAR 单独进行高斯-

牛顿迭代匹配时的定姿结果也是不准确的,而且动态过大时仅靠 INS 的姿态信息也无法给出准确的结果。



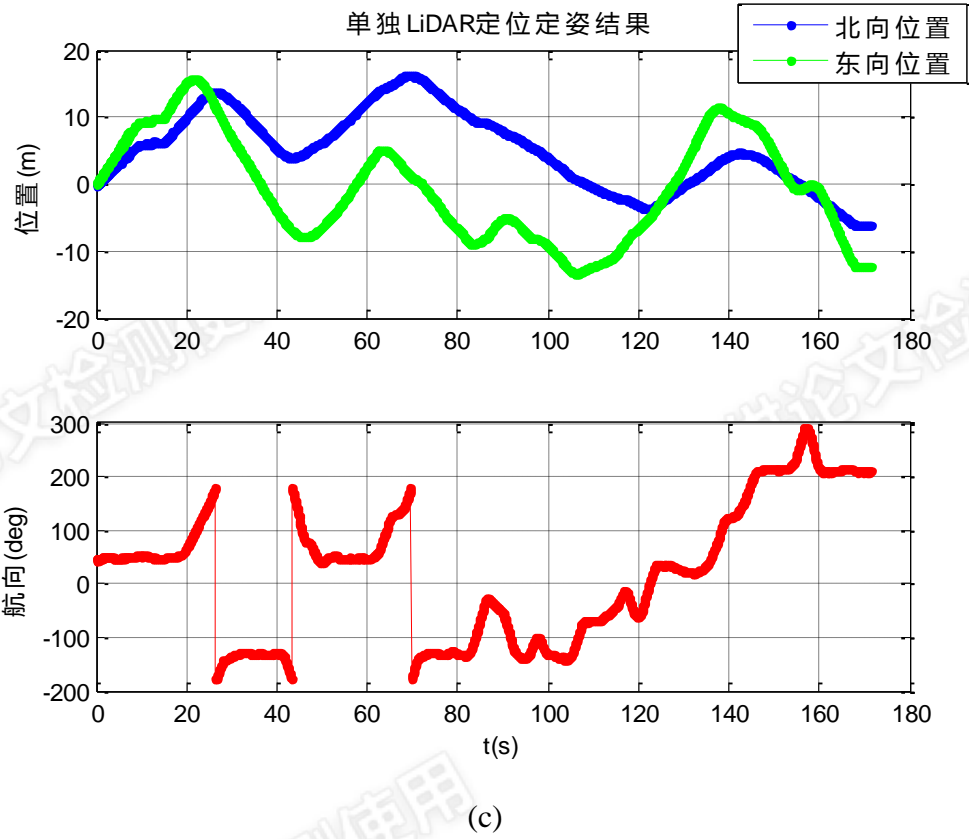


图 5-8 (a) LiDAR/INS 组合定位定姿结果; (b) 仅 INS 姿态信息辅助定位定姿结果; (c) LiDAR

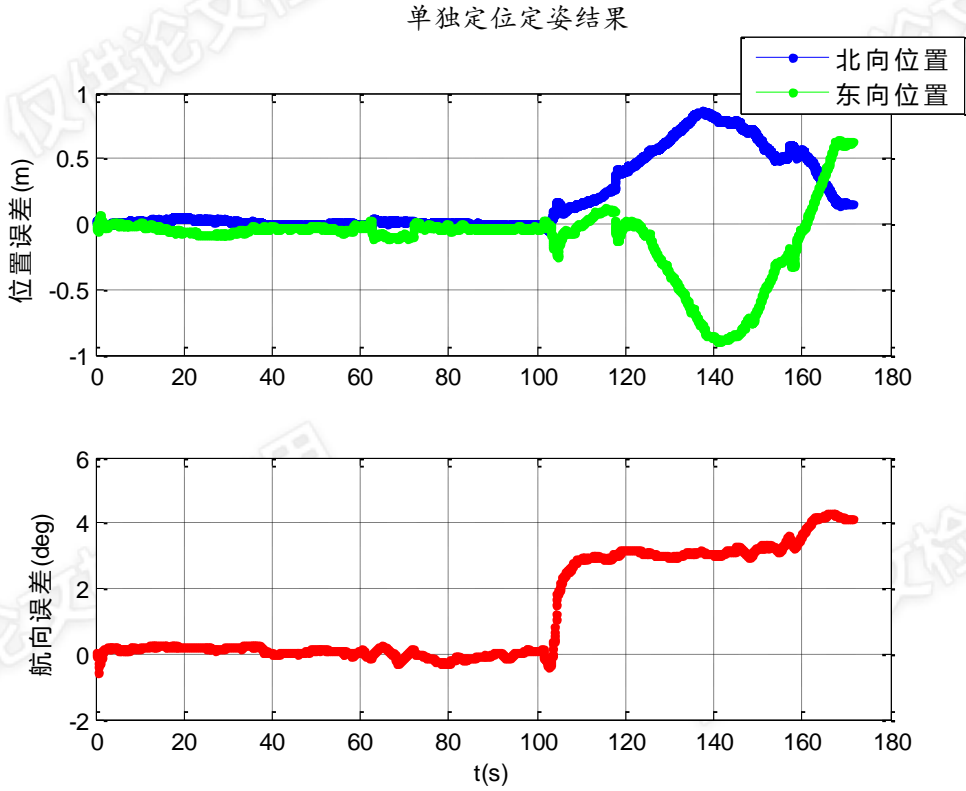


图 5-9 LiDAR/INS 组合定位结果与仅 INS 姿态辅助 LiDAR 定位结果之间的差异

表 5-5 LiDAR/INS 组合定位结果与仅 INS 姿态辅助 LiDAR 定位结果的差异统计

	RMS ERROR	MEAN ERROR	MAXIMUM ERROR
北向位置	0.342(m)	0.202(m)	0.861 (m)
东向位置	0.301m)	0.170(m)	0.889 (m)
航向	2.050 (degree)	1.351(degree)	4.314(degree)

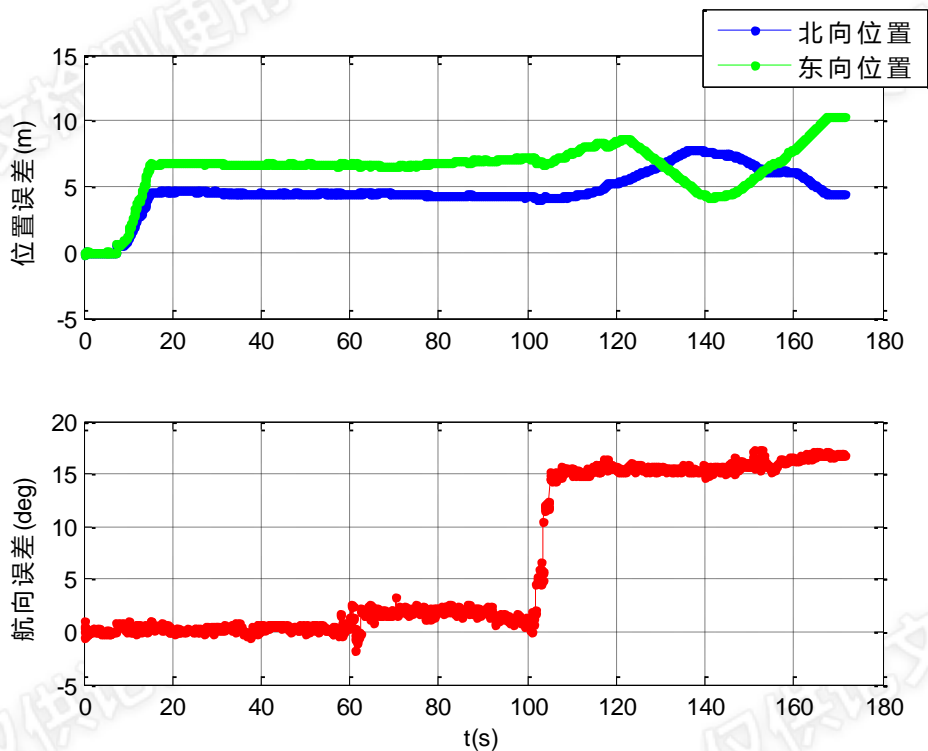


图 5-10 LiDAR/INS 组合定位结果与 LiDAR 单独定位结果之间的差异

表 5-6 LiDAR/INS 组合定位结果与 LiDAR 单独定位结果的差异统计

	RMS ERROR	MEAN ERROR	MAXIMUM ERROR
北向位置	5.002(m)	4.741 (m)	7.867(m)
东向位置	6.734m)	6.439 (m)	10.354(m)
航向	9.931(degree)	6.7861(degree)	17.268 (degree)

5.6 近实时算法的实验验证

5.6.1 精度验证

为了验证本文提出的近实时算法的定位和制图精度，图 5-11 是由本文改进后算法绘制成的室内地图，显示了 FGI 的平台图。后处理算法和近实时算法所处理得到的移动机器平台的轨迹也显现在图中。从图中的轨迹结果可以看出，绿色轨迹曲线与红色轨迹曲线吻合地非常好，这也意味着本文提出的近实时算法与后处理算法定位精度处于同一水平。

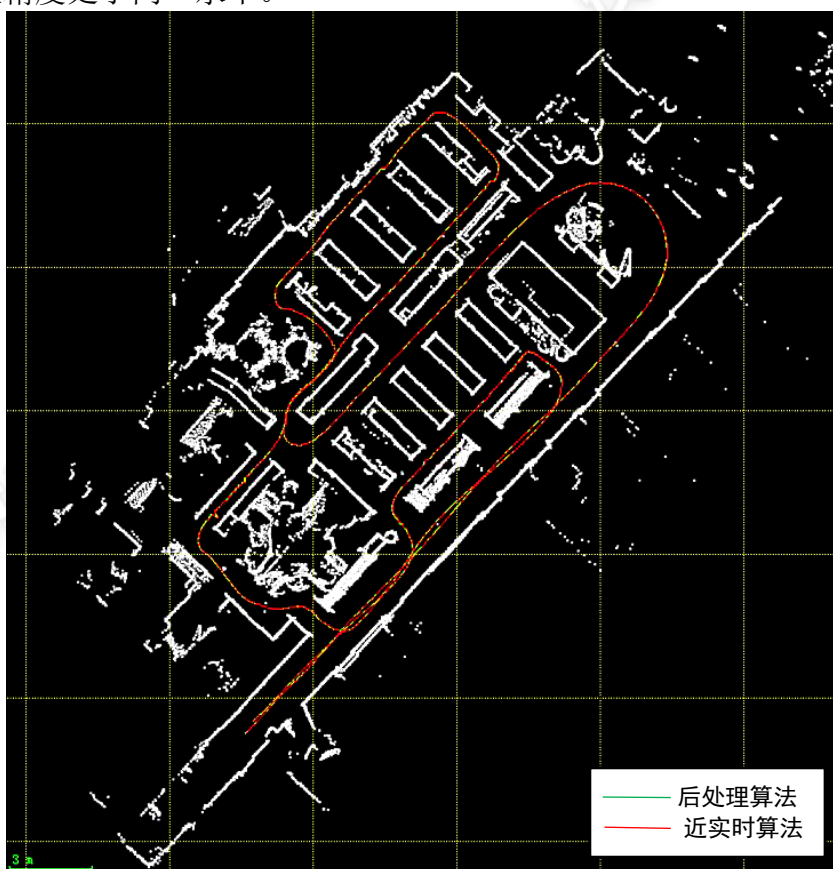


图 5-11 后处理算法和近实时算法处理得到的轨迹图

整个运动过程持续了 176s。图 5-12 显示了后处理算法和近实时算法的定位结果之间的差异，表 5-7 是图 5-12 的统计结果。根据表 5-7，北向位置和东向位置误差的 RMS 分别是 0.014m 和 0.44m；航向误差的 RMS 是 0.198° 。定位误差在厘米级，航向误差也在 LiDAR 的角度分辨率 0.25° 之内。所以，可以得出结论：近实时算法的定位定姿精度与后处理算法可以认为是在同一水平。

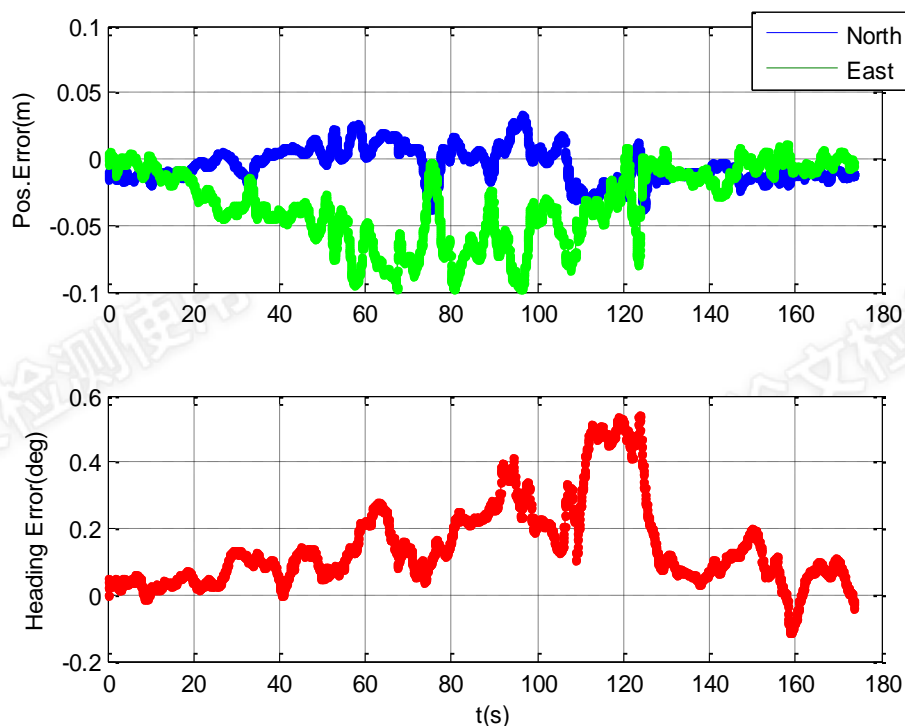


图 5-12 后处理算法和近实时算法处理的定位差异图

表 5-7 后处理算法和近实时算法处理的定位差异

	RMS ERROR	MEAN ERROR	MAXIMUM ERROR
北向位置	0.014(m)	0.012(m)	0.038(m)
东向位置	0.044m)	0.034 (m)	0.098 (m)
航向	0.198(degree)	0.151 (degree)	0.542 (degree)

除了比较定位精度，还需验证近实时算法的制图精度。5.4 节中的实验已经给出了后处理算法和 TLS 的制图结果，并且验证了后处理算法的制图精度，图 5-13 截取了利用近实时算法处理绘制的部分地图，选取了与 5.4 节中 TLS 制图结果中相同的特征点，以 TLS 的结果为参考地图，统计出近实时算法地图中这些特征点的坐标的误差的 RMS 为 0.073m。后处理算法与近实时算法得到的误差的 RMS 小于 0.02m，考虑到人工取点过程中存在的操作误差，可以认为近实时算法与后处理算法的制图精度也为同一等级。

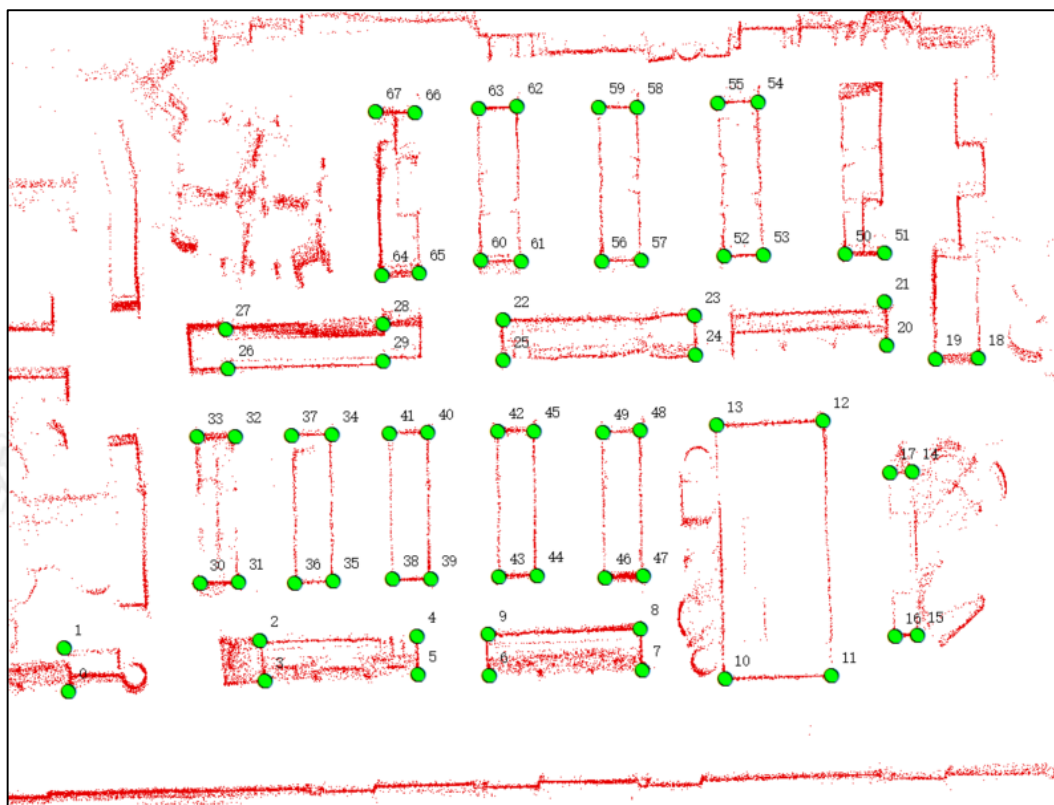


图 5-13 近实时算法绘制的部分地图和相应的特征点

表5-8. 所选特征点的结果精度统计表

	Feature Points	RMS (m)
LiDAR/INS组合制图	67	0.073

5.6.2 实时性验证

为了评估近实时算法的性能，图 5-14 是它每一次输出结果的时间延迟与后处理的进行比较的结果。从图 4-1 的理论过程上来说，当没有接收到 LiDAR 数据时，将会输出 INS 机械编排的结果，那么这时每一次输出结果的时间延迟就是 INS 机械编排的耗时。当接收到 LiDAR 数据时，对于后处理算法来讲，每一次输出结果的时间延迟等于完成 LiDAR 扫描匹配、IMU 机械编排和 kalman 更新的时间总和；但是对于近实时算法来说，每一次输出结果的时间延迟只由 IMU 机械编排、kalman 更新和一步状态转移过程的耗时组成，耗时更短。

如图 5-14 所示，当接收到 LiDAR 数据时，近实时算法中每一次输出结果的时间延迟远远小于后处理算法中的结果。从表 5-9 的统计结果来看，后处理算法中的平均输出延迟为 0.841ms，最大为 14.509ms；而后处理算法中的平均输出延

迟为 0.285ms，最大为 3.760ms。近实时算法将平均输出延迟减少了 3 倍，最大值减少了大约 4 倍。之所以能将大大减小输出延迟，是因为近实时算法利用率双线程处理的方案，并用一步状态转移的方法使得之前的更新信息(即 LiDAR 扫描匹配得到的结果)来修正当前的 INS 机械编排的结果，修正后的结果可以直接输出，不需要等待当前 LiDAR 扫描匹配完成后才输出。因此，根据图 5-14，本文提出的近实时算法可以有效地减少输出延迟，保证其实时性能。

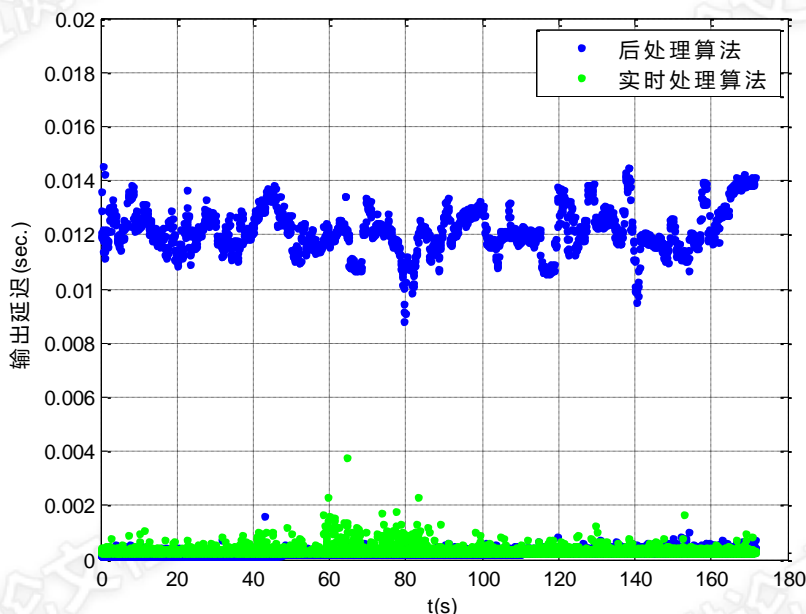


图 5-14 后处理算法和近实时算法处理的输出延迟对比图

表 5-9 后处理算法和近实时算法处理的输出延迟统计

	平均值	最大值
后处理算法	0.841ms	14.509ms
近实时算法	0.285ms	3.760ms

6 总结与展望

6.1 论文工作总结

本文针对现有的基于 LiDAR/INS 组合的室内 SLAM 算法的不足之处提出了一套改进 LiDAR/INS 组合定位定姿算法，并将原有的后处理方案改进成了实时方案，最后在所搭建的 LiDAR/INS 组合移动测图机器人平台上测试并分析了改进算法的性能，验证了改进算法的可行性。本文的主要研究成果如下：

(1) 详细介绍了多分辨率栅格地图的创建过程和 LiDAR 搜索匹配算法的原理和具体实现过程。通过实验评估了本文所采用的高斯-牛顿匹配算法的性能。实验表明，采样高斯-牛顿算法后 LiDAR 搜索匹配的平均匹配耗时为 10.9ms，最长为 14.4ms。该算法能在保证制图精度的前提下，无需设置搜索范围，大大减小 LiDAR 搜索匹配耗时。

(2) 针对高斯-牛顿迭代容易陷入局部最优的现象，提出了一套改进的 LiDAR/INS 组合算法，通过实时估计惯导零偏补偿惯导误差，来提供更准确的搜索初值，从而有效避免陷入局部最优。设计实验分析了改进后 LiDAR/INS 组合算法的制图结果和定位结果。实验结果表明，与 TLS 所绘制的参考地图相比较，本文所提出的 LiDAR/INS 组合导航算法的制图误差可以达到 0.056m，属于厘米级制图精度；即使在动态较大的情况下，改进后的 LiDAR/INS 的组合算法仍然能够有效地帮助 LiDAR 匹配避免陷入局部最优，而且动态信息越丰富，INS 发挥的作用越大。

(3) 针对 LiDAR 匹配耗时远远大于 IMU 解算耗时而导致标准 Kalman 组合结果无法实时输出的问题，本文提出了一种分步完成观测更新再将误差状态及其协方差一步转移到当前时刻的 Kalman 滤波数学模型，保证了系统的实时性。通过精度和实时性实验证明：1. 所提出的近实时算法能获得与后处理算法相同的定位和制图精度(厘米级)；2. 近实时算法将后处理算法的最长耗时从 14.51ms 下降到了 3.76ms。显著提高了实时性能。所以本文提出的近实时算法能在不降低制图精度的同时，有效减小结果输出的延迟。

6.2 工作展望

未来的工作主要是本文工作的扩展，具体包括：

(1) 将所提出的准实时算法从平板电脑移植到嵌入式平台上，实现真正的实时 SLAM 系统。

(2) 将在本文所搭建的硬件平台上，再搭载一个垂直放置的 2D LiDAR，扫描断面轮廓，结合 INS 的定位定姿信息，形成 3D 测图。由于 INS 能测出俯仰角、横滚角信息，所以 INS 在 3D 的环境测图应用中发挥的作用将比在 2D 环境中更明显。

(3) 搭载上视觉传感器，集成一套 LiDAR/INS/Camera 组合的室内 SLAM 系统。视觉传感器能够从环境中提取比 LiDAR 更丰富的信息，由于其自主性、可靠性好而且更廉价，使得在图像处理领域广泛运用。因此，后续的工作将实现一套 LiDAR/INS/Camera 的多传感器组合导航定位算法。

上述工作将更加丰富本文所研究的多传感器组合的室内高精度移动测图方案。室内环境中的 SLAM 技术是一项非常重要且有技术难度的研究课题，随着国内外相关研究的不断深入和发展，该技术将在室内高精度移动测图、机器人室内定位等领域发挥重要作用。

参考文献

- [1] 陈卫东, 张飞. 移动机器人的同步自定位与地图创建研究进展[J]. 控制理论与应用, 2005, 22(3):455-460.
- [2] S. Thrun., W. Burgard, D. Fox., Probabilistic Robotics, MIT Press, 2005.
- [3] S. Huang, G. Dissanayake. Convergence and Consistency Analysis for Extended Kalman Filter Based SLAM, In IEEE Trans. on Robotics, 2(5), Oct. 2007.
- [4] S. Thrun, D. Fox, W. Burgard, F. Dellaert, Robust Monte Carlo Localization for Mobile Robots, In Artificial Intelligence, 128, 2001.
- [5] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem, In AAAI National Conference on Artificial Intelligence, 2002.
- [6] S. Thrun, M. Montemerlo. The GraphSLAM Algorithm With Applications to Large-Scale Mapping of Urban Structures, In Proc. of the Int. Journal on Robotics Research, 2005.
- [7] F. Lu, E. Milios, Globally Consistent Range Scan Alignment for Environment Mapping, In Autonomous Robots, 1997.
- [8] G. Grisetti, C. Stachniss, W. Burgard. Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters, In Trans. on Robotics , 23(1), Feb. 2007.
- [9] R. Vincent, B. Limketkai, M. Eriksen. Comparison of indoor robot localization techniques in the absence of GPS, In Proc. of SPIE: Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XV of Defense, Security, and Sensing Symposium, April 2010.
- [10] S. Kohlbrecher, J. Meyer, O. Von Stryk, U. Klingauf. A Flexible and Scalable SLAM System with Full 3D Motion Estimation, In the Int. Symp. on Safety, Security and Rescue Robotics (SSRR), Nov. 2011.
- [11] Hess W, Kohler D, Rapp H, et al. Real-time loop closure in 2D LIDAR SLAM[C]// IEEE International Conference on Robotics and Automation. IEEE, 2016:1271-1278.
- [12] Soloviev, A. Tight coupling of GPS, laser scanner, and inertial measurements for navigation in urban environments. In Proceedings of the Position, Location and Navigation Symposium, 2008 IEEE/ION, Monterey, CA, USA, 5-8 May 2008; pp. 511-525.
- [13] Klein, I.; Filin, S. Lidar and INS Fusion in Periods of GPS Outages for Mobile Laser Scanning Mapping Systems. ISPRS-Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 2011, 3812, 231-236.
- [14] Tang J, Chen Y, Jaakkola A, et al. NAVIS-An UGV Indoor Positioning System Using Laser Scan Matching for Large-Area Real-Time Applications[J]. Sensors, 2014, 14(7):11805-11824.
- [15] 杨景阳. 微惯性辅助的激光雷达室内同步建图与定位技术研究[D]. 南京航空航天大学, 2014.
- [16] 李荣冰, 刘建业, 曾庆化,等. 基于 MEMS 技术的微型惯性导航系统的发展现状. 中国惯性技术学报, 2004, 12(6):88-94.
- [17] 高钟毓. 惯性导航系统技术. 清华大学出版社, 2012.
- [18] Bailey T, Durrant-Whyte HF. Simultaneous localization and mapping (SLAM):Part II, State of the art. IEEE Robotics and Automation Magazine, 2006, 13(3): 108~127.
- [19] Fotopoulos, G.; Cannon, M. An overview of multi-reference station methods for cm-level positioning. GPS Solut. 2001, 4, 1-10.
- [20] Czapla T, Wrona J. Technology Development of Military Applications of Unmanned Ground Vehicles Vision Based Systemsfor UAV Applications. Springer International Publishing, 2013:293-309.
- [21] 李德仁. 移动测量技术及其应用[J]. 地理空间信息, 2006,4(4): 1~5.
- [22] 章大勇. 激光雷达/惯性组合导航系统的一致性与最优估计问题研究[D]. 国防科学技术大学, 2010.
- [23] 王平, 鲁可, 曹云峰,等. 微小型飞行器室内激光/INS 融合定位方法研究. 飞行器测控学报, 2013, 32(4):340-344.
- [24] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision (darpa)," in DARPA Image Understanding Workshop, Apr 1981, pp. 121-130.

- [25] Burguera A, González Y, Oliver G. On the use of likelihood fields to perform sonar scan matching localization.[J]. *Autonomous Robots*, 2009, 26(4):203-222.
- [26] Shin, Eun-Hwan. Estimation techniques for low-cost inertial navigation [microform][J]. 2005.
- [27] Tang J, Chen Y, Niu X, et al. LiDAR Scan Matching Aided Inertial Navigation System in GNSS-Denied Environments[J]. *Sensors*, 2015, 15(7):16710-28.
- [28] Savage P G. Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms[J]. *Journal of Guidance Control & Dynamics*, 1998, 21(1):19-28..
- [29] 秦永元. 惯性导航[M]. 科学出版社, 2006.
- [30] Bortz, J. E. A new concept in strapdown inertial navigation[J]. Massachusetts Institute of Technology, 1970.
- [31] Savage P G. Strapdown inertial navigation integration algorithm design. I: Attitude algorithms.[J]. *Journal of Guidance Control & Dynamics*, 2012, 21(1):19-28.
- [32] 王光辉. 基于时滞多传感器数据的信息融合滤波[D]. 黑龙江: 黑龙江大学, 2013.[Wang Guanghui. Information Fusion Estimator of Multi-sensor Linear Discrete Systems with Constant Time-delay[D]. Heilongjiang: Heilongjiang University, 2013.]
- [33] Kailath T, Sayed A H, Hassibi B. Linear estimation[M]. Prentice Hall Upper Saddle River, NJ, 2000.
- [34] 卢晓. 观测时滞系统的最优滤波[D]. 大连: 大连理工大学, 2008.[Lu Xiao. Optimal Filtering for Systems with Delayed Measurements[D]. Dalian: Dalian University of Technology, 2008.]
- [35] 李林林. 测量数据延迟下的不完全量测滤波研究[D]. 南京: 南京理工大学, 2012.[Li Lingling. Filtering Theory and Method of Partial Observation with the Delayed Measurement[D]. Nanjing: Nanjing University of Science and Technology, 2012.]
- [36] 章红平, 常乐, 汪宏晨,等. GNSS/INS 紧组合算法实时性改进与嵌入式验证[J]. *东南大学学报(自然科学版)*, 2016, 46(4):695-701.
- [37] Guivant J E, Nebot E M. Optimization of the simultaneous localization and map-building algorithm for real-time implementation [J]. *IEEE Transactions on Robotics & Automation*, 2001, 17(3):242-257.
- [38] Miller, M.; Uijt D.H.M.; Soloviev, A.; Veth, M.; Klausutis, T.T.J. Navigation in GPS Denied Environment: Feature Aided Inertial Systems; Air Force Research Lab: Eglin AFB, FL, USA, 2010.
- [39] Aghamohammadi A , Taghirad H D, Tamjidi A H, et al. Feature-Based Laser Scan Matching For Accurate and High Speed Mobile Robot Localization[C]// *European Conference on Mobile Robots*, Emcr 2007, September 19-21, 2007, Freiburg, Germany. DBLP, 2007.
- [40] Aghamohammadi A A, Tamjidi A H, Taghirad H D. SLAM Using Single Laser Range Finder[J]. *Ifac Proceedings Volumes*, 2008, 41(2):14657-14662.
- [41] Besl P J, McKay N D. A method for registration of 3-D shapes[J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 1992, 14(2):239-256.
- [42] Bosse M C. Atlas: a framework for large scale automated mapping and localization[M]. Massachusetts Institute of Technology, 2004.
- [43] Censi A. An ICP variant using a point-to-line metric[C]// *IEEE International Conference on Robotics and Automation*. IEEE Xplore, 2008:19-25.
- [44] Bachrach, A.; Prentice, S.; He, R.; Roy, N. RANGE-Robust autonomous navigation in GPS-denied environments. *J. Field Robot.* 2011, 28, 644–666.
- [45] Steux, B.; El Hamzaoui, O. TinySLAM: A SLAM algorithm in less than 200 lines C-language program. In *Proceedings of the Control Automation Robotics & Vision (ICARCV)*, Singapore, 7–10 December 2010; pp. 1975–1979.
- [46] Pfister, S.T. Algorithms for Mobile Robot Localization and Mapping, Incorporating Detailed Noise Modeling and Multi-Scale Feature Extraction. Ph.D. Thesis, California Institute of Technology, Pasadena, CA, USA, 2006.
- [47] Moravec H, Elfes A. High resolution maps from wide angle sonar[A]. In *Proceedings of the IEEE International Conference on Robotics and Automation(ICRA'85)*[C]. IEEE, 1985: 116-121
- [48] Elfes A. Sonar-based real-world mapping and navigation[J]. *IEEE Journal on Robotics and Automation*, 1987, 3(3): 249-265

- [49] 徐则中. 移动机器人的同时定位和地图构建[D]. 浙江大学, 2004.
- [50] Chong K S, Kleeman L. Mobile Robot Map Building from an Advanced Sonar Array and Accurate Odometry[J]. *International Journal of Robotics Research*, 1999, 18(1):20-36.
- [51] Leonard J, Durrantwhyte H, Cox I. Dynamic map building for autonomous mobile robot[J]. *International Journal of Robotics Research*, 1992, 11(4):286-298.
- [52] M. Habbecke and L. Kobbelt, "Iterative Multi-View Plane Fitting," in *Vision, modeling, and visualization*, 2006, pp. 73–80.
- [53] 王森, 杨克俭. 基于双线性插值的图像缩放算法的研究与实现[J]. *自动化技术与应用*, 2008, 27(7):44-45.
- [54] 张仲颖. 计算机视觉中运动分析的金字塔结构方法[D]. 北方交通大学 北京交通大学, 1994.
- [55] 董绪荣. GPS/INS 组合导航定位及其应用[M]. 国防科技大学出版社, 1998.
- [56] Xsens Technologies B.V. MTi User Manual, Document MT0605P, Revision E.17 January 2014.
- [57] Qt User Interface Documentation: Qt5.2 Class Reference. Available online: <http://qt-project.org/dot/qt-5/topics-ui.html> (accessed on 23 December 2013).
- [58] Bosse M C. Atlas: a framework for large scale automated mapping and localization[M]. Massachusetts Institute of Technology, 2004.
- [59] Censi A. An ICP variant using a point-to-line metric[C]// *IEEE International Conference on Robotics and Automation*. IEEE, 2008:19-25.
- [60] Biber, P., Fleck, S., & Straßer, W. (2004). A probabilistic framework for robust and accurate matching of point clouds. In *Proceedings of the 26th pattern recognition symposium (DAGM)*.
- [61] Bosse, M., Newman, P. M. Leonard, J.J. & Teller, S. (2004). SLAM in large-scale cyclic environments using the atlas framework. *International Journal of Robotics Research*, 23(12), 1113–1139.
- [62] Thrun, S.; Leonard, J.J. Simultaneous Localization and Mapping. In *Springer Handbook of Robotics*; Springer Berlin Heidelberg: Berlin, Germany, 2008; pp. 871–889.
- [63] Kaijaluoto, R., Kukko, A., Hyypä J., 2015. Precise Indoor Localization For Mobile Laser Scanner. *Indoor-Outdoor Seamless Modelling, Mapping and Navigation*, 21–22 May 2015, Tokyo, Japan. *The International Archives of the Photogrammetry, Remote Sensing and Spatial*
- [64] Jung J, Yoon S, Ju S, et al. Development of Kinematic 3D Laser Scanning System for Indoor Mapping and As-Built BIM Using Constrained SLAM. [J]. *Sensors*, 2015, 15(10):26430-26456.
- [65] Jung J, Hong S, Jeong S, et al. Productive modeling for development of as-built BIM of existing indoor structures[J]. *Automation in Construction*, 2014, 42(2):68-77.

致谢

当论文写到致谢的时候，突然感慨良多。时间过得真的好快，一晃我在武大都呆了7年了，在这七年里，我们都成长了很多。感谢我的母校--武汉大学，在这所美丽又包容的学校里，我能在工学部主教的英语课堂上抬头就看见繁盛的樱花树；我能在上完体育课路过情人坡时，惊喜地发现金黄的银杏树叶铺满整个山坡；我能在人文馆、在活动中心、在会议室聆听到学校请来的各领域大牛的讲座；更重要的是，我能在这个学校遇见这么多珍贵的人。

牛小骥教授是我研究生生涯中的第一导师，第一次见到牛老师，就觉得牛老师倍感亲切，在研究生的三年里，更是为牛老师正直的品格、严谨的科研作风、谦逊的学习态度感到由衷的敬佩。感谢牛老师这三年来对我学术科研上的悉心指导，感谢牛老师耐心地帮我分析找工作的形势，为我今后的人生道路指点迷津。从牛老师那里学习到的不仅仅是科研知识，更重要的是如何成为一个高尚的人。感谢与牛老师的相遇，牛老师将成为我这一生学习的榜样。感谢我们组的章红平教授，他开朗的性格、平易近人的态度为组里营造了轻松融洽的学习氛围。

唐健老师是我的第二导师，是一位年轻有为的老师。在学习室内 SLAM 方面一直得益于唐老师的悉心指导，唐老师的编程能力之强、逻辑之缜密让我感到由衷地佩服。非常感谢唐老师对我编程上的悉心指导和在研究工作上毫无保留的支持，并认真耐心地指导我如何写好论文。本论文的完成也是在唐老师搭建的 NAVIS 软件平台上进一步的工作，没有唐老师搭建的软件平台我不可能顺利的进行这篇论文中的研究。对恩师们的感恩之情，无以言表，在这里，我谨向恩师们表示最崇高的敬意和感激。

感谢已经毕业了班亚龙师兄和程政师兄，在移动机器人平台搭建初期，班师兄和程师兄都给予过很多硬件方面的指导。班师兄认真严谨的科研态度是我学习的楷模，程师兄乐观积极的生活态度也让人称颂。

感谢坐在我旁边的常乐同学，平时算法学习上遇到的问题，多亏他的耐心讲解。虽然常乐同学和我同级，但是他的学术能力远在我之上，而且大部分的时间都泡在实验室钻研学术，其认真的工作态度令人佩服。感谢旷俭师兄在我调试 LiDAR/INS 组合导航算法时细心地帮我指出程序中的错误，没有他的帮助，我可能需要花费更多的时间去找出问题，这个过程中我也发现了我在写程序上不够

细心的缺点。感谢我们导航组的张提升老师、郭文飞老师、张全师兄、陈起金师兄对我学习上的帮助。感谢毛大智师弟和陈映秋师妹对本论文中室内 SLAM 实验方面的帮助。

感谢硬件机房的所有人：感谢徐良春师兄、严昆仑师兄、张鹏辉师兄、李冰师兄、蔡磊、黄夙寒、李卓师妹、刘蘅嵘师妹、祁发瑞师弟、谭俊雄师弟。是你们让整个硬件机房充满了欢声笑语，大家互帮互助、团结友爱，在轻松、和谐的氛围里学习。

感谢 2014 级硕士班的所有同学，很开心能与来自天南地北的大家相聚于此。

感谢我在武大的好朋友刘向东、石建炜同学，我们三个人是本科同班同学、研究生校友，七年的同窗之情让我们友谊更加深厚，而今毕业之际，你们一个南下、一个东去，唯有希望你们各自保重，苟富贵，勿相忘。

特别感谢我的父母，感谢你们把我生在一个如此有爱的家庭，感谢你们二十多年一直呵护我成长、无私奉献，从现在开始，换我来为你们挡风遮雨。

最后，感谢人生道路中出现在我生命中的每一个人，是你们，让我成为了现在的我。