

工程硕士学位论文

基于云端的移动机器人室外环境定位系统
的设计与实现

**CLOUD BASED REAL TIME LOCALIZATION FOR
GROUND MOBILE ROBOT IN OUTDOOR
ENVIRONMENT**

庞溯



哈尔滨工业大学
2016 年 12 月

国内图书分类号：TP393.4

学校代码：10213

国际图书分类号：681.14

密 级：公开

工程硕士学位论文

基于云端的移动机器人室外环境定位系统的 设计与实现

硕 士 研 究 生：庞溯

导 师：朱晓蕊 教授

申 请 学 位：工程硕士

学 科：控制工程

所 在 单 位：深圳研究生院

答 辩 日 期：2016 年 12 月

授予学位单位：哈尔滨工业大学

Classified Index: TP393.4

U.D.C: 681.14

Dissertation for the Master's Degree of Engineering

**CLOUD BASED REAL TIME LOCALIZATION FOR
GROUND MOBILE ROBOT IN OUTDOOR
ENVIRONMENT**

Candidate:	Pang Su
Supervisor:	Prof. Zhu Xiaorui
Academic Degree Applied for:	Master's Degree of Engineering
Speciality:	Control Engineering
Affiliation:	Shenzhen Graduate School
Date of Defence:	December, 2016
Degree-Conferring-Institution:	Harbin Institute of Technology

摘 要

定位一直是户外移动机器人导航系统的一个重要组成部分。尤其对于运行于户外大场景的机器人而言，现今主流的定位方法仍然是基于惯导技术和全球卫星定位系统（GPS），其定位功能一直受限于惯性传感器的精度、GPS 信号的质量以及机器人本体处理器有限的运算和存储能力。云机器人是一个新兴的机器人领域，它依赖于云计算、云存储和其他互联网技术，使得机器人受益于云端服务器强大的计算能力、存储能力以及资源共享能力。通过网络连接，将大量的实验数据以及繁重的计算任务转移到云端，从而大幅降低机器人成本，增加机器人的便携性。本课题的主要目的是设计一种基于云端的机器人架构以实现运行于户外大场景的移动机器人的精确定位。从而解决传统方法因过于依赖于惯性导航传感器以及 GPS 信号而导致定位不准确的问题。

整个系统主要分为离线过程和在线过程两个部分。离线过程主要包括提取道路网信息以及采集带有位置信息的参考图像。前者是为了提取新的道路网信息并将其更新至云端的服务器中以备定位使用。参考图像数据库中的每幅图像都包含一个道路网上的地理坐标与其相对应，用于通过视觉补充提高机器人的定位精度。在线过程主要是基于云端的机器人定位过程，通过道路网信息计算机器人倾斜模型（RTI），采用粒子滤波融合 RTI 模型、传感器数据以及云端视觉补充结果得出准确的定位结果。同时，对于机器人和云端通信所产生的时间延迟而导致的定位误差，本文也采用了有效的误差补偿算法对其进行了补偿。

本文设计了三个场景的实验来验证定位算法。三个场景分别选择了不同的道路类型，其中包含了次干道、支路、城市峡谷以及隧道等具有较高定位难度的复杂城市道路环境。最长实验路线长度为 13.1 千米。良好的实验结果充分验证了本文方法的实用性以及算法的有效性，适用于户外大场景的机器人自主实时定位。

关键词：云机器人；定位；粒子滤波；户外大场景

Abstract

Localization is an important problem for the navigation of outdoor mobile robots. Robot localization especially for large-scale exploration is mainly based on inertial navigation technology and Global Positioning System (GPS). The performance has been limited by the sensor measurement accuracy, GPS signal and other inherent computation and storage constraints of the onboard processor. Cloud robotics is a new field of robotics, relying on cloud computing, cloud storage and other Internet technology, makes the robot benefit from the powerful computation, storage and shared resources of the cloud server. The large amount of experiment data and computation load can be distributed to the cloud to lower the cost and increase the portability of the robot. The main purpose of this project is to propose a cloud based real time localization architecture to allow a ground mobile robot to identify its location in large scale outdoor environment without the constraints of onboard inertial navigation sensors and the GPS.

The proposed cloud based architecture consists of two phases: offline and online. The extraction of road network information and the construction of the reference images constitute the offline phase. The former is to extract new road networks and update the existing road network map stored in the cloud. To further improve localization accuracy for mobile robot, a reference image database is constructed in the proposed framework. Image of a specific location in the road network is taken as a reference for localization. The vision based supplement is used to increase the accuracy of localization. The online phase is for mobile robot localization based on the cloud. The update of the road network map and the computation of the robot terrain inclination (RTI) model as well as reference image matching are executed in the cloud. A particle filter with network delay compensation localization algorithm is executed on the mobile robot based on the local RTI model and recognized location sent from the cloud.

The proposed methods are tested on three different challenging outdoor scenarios with a ground mobile robot equipped with minimal onboard hardware where the longest trajectory was 13.1km. The three scenarios consist of different road types including service way, urban canyon and tunnel etc which are challenging for localization process. Experimental results show that this method could be applicable to large-scale outdoor environments for autonomous robots in real-time.

Keywords: Cloud Robotics, localization, partical filter, outdoor environment

目 录

摘 要	I
Abstract	II
第 1 章 绪 论	1
1.1 课题研究的背景和意义	1
1.2 国内外研究现状	2
1.3 本课题主要研究内容	5
1.4 本文结构	6
第 2 章 系统总体结构	7
2.1 离线过程	7
2.1.1 道路网信息的提取	7
2.1.2 采集带有位置信息的参考图像	8
2.2 在线过程	9
2.2.1 云端服务器	9
2.2.2 移动机器人	10
2.2.3 在线过程运行方式	10
2.3 本章小结	10
第 3 章 云端服务器部分算法	12
3.1 道路网信息的处理	12
3.1.1 坐标转换算法	12
3.1.2 机器人地形倾斜模型 (RTI model) 的计算	14
3.2 视觉补充	15
3.3 本章小结	16
第 4 章 移动机器人部分算法	18
4.1 粒子滤波算法	18
4.2 基于粒子滤波的定位算法	19
4.2.1 运动模型	19
4.2.2 测量值模型	20
4.2.3 权重计算及重采样	21
4.2.4 感知复位	22
4.2.5 定位算法伪代码	22

4.3 网络延迟补偿算法	24
4.4 本章小结	25
第 5 章 实验设计与结果分析	27
5.1 实验设计	27
5.1.1 实验平台	27
5.1.2 实验方案及场景选择	28
5.2 实验结果分析	30
5.2.1 实验一结果分析	30
5.2.2 实验二结果分析	34
5.2.3 实验三结果分析	39
5.3 本章小结	44
结 论	45
参考文献	46
哈尔滨工业大学学位论文原创性声明和使用权限	50
致 谢	52

第 1 章 绪 论

1.1 课题研究的背景和意义

卡内基梅陇大学的 James Kuffner 教授于 2010 年在 Humanoids 会议上提出了“Cloud Robotics（云机器人）”^[1]的概念，这一概念的提出即引起了大家对之的广泛讨论和兴趣，在此之后许多相关学科的研究人员和专家都对这个相对较新的研究领域提出了建议和设想^[2,3]，并为机器人领域未来的发展提供了一个新的方向。

云机器人得益于云计算、云存储和其他相关互联网技术，使得机器人受益于云端服务器强大的计算能力、存储能力以及资源共享能力。云机器人使得机器人充分利用数据传输率的快速增长来卸载一些没有硬性规定实时要求的任务，这一点对移动机器人尤为有利，因为机载计算的实现需要额外的电量需求，这就降低了机器人的运行时间以及移动便携性能。云机器人通过网络连接，将大量的实验数据以及繁重的计算任务转移到云端服务器，从而大幅降低机器人本体成本，增加机器人的便携性。

目前移动机器人已经广泛地应用于工业生产和日常生活的方方面面，比如工厂中的货物搬运、危险环境探测以及社区巡逻服务等，既提高了整体的工作效率也适当保证了相关领域从业人员的安全，同时使得人类从一些繁重的体力劳动当中解放出来。移动机器人实现自主导航的基础就在于移动机器人的定位，是机器人完成其他复杂任务的前提条件，机器人通过对外部环境的检测从而推算出机体在环境中的位置。现今对于室外环境的机器人定位问题主要的方式有三种：基于全球卫星定位系统（GPS）定位、基于惯性导航定位、组合导航定位。

基于 GPS 的定位方法是最为简单和直接的方法。只需在机器人机体上安装好 GPS 信号接收机来接收卫星信号即可^[4-7]。该种方法的定位效果充分依赖于 GPS 信号的质量。所以在相对开阔的场景，由于环境对 GPS 信号的干扰和遮挡较小，能够获得较为准确的定位结果。但是对于城市中相对复杂户外环境，由于道路两旁建筑物对于 GPS 信号的遮挡和反射，以及在隧道中无法接收到 GPS 信号，都使得定位结果非常不准确或者直接导致定位失败^[4]。

基于惯性导航定位的方法属于自主性的导航方法。主要是利用惯性传感器（比如磁力计、加速度计、陀螺仪等等）结合里程计或者编码器采集的机器人速度信息推算机器人的位置。由于定位结果主要依赖于机载的各种传感器，从而不会像基于 GPS 的方法一样对周围环境非常敏感。但是传感器本身的特性将会对定位结

果产生巨大的影响，比如传感器的累计误差，如果不能及时对其进行校准，产生的误差和时间的推移呈正相关的关系^[8]，传感器本身的随机误差以及安装误差等也都会对定位结果产生明显的影响。

针对基于 GPS 和惯性导航方法的优缺点，组合导航方法采用结合两种方法的方式，通常通过信息融合算法将两种定位方式的结果进行融合。利用两种方法各自的优点，得出更为准确的定位结果^[8]。但是这种方法仍然存在明显的缺点，比如鲁棒性较差，当错误的估计状态产生于系统当中，如果没有适当的修正，会产生连锁效应，很可能导致接下来的估计状态错误，使得系统不稳定^[9]，还有当 GPS 信号丢失较长时间之后，系统的定位精度会降低。

云机器人作为一个新兴领域，对于许多传统问题都提供了一套新的解决方案，因此，采用基于云端的机器人架构，利用云端强大的存储和计算能力，研究一种不依赖于 GPS 信号且具有强鲁棒性的户外移动机器人定位方法，使其在户外大场景长时间稳定持续保持良好的定位效果具有重要的意义。

1.2 国内外研究现状

在过去的十年间有许多国内外的学者和研究团队把目光放在室外环境的移动机器人定位上，其中大部分的算法是基于 GPS 的。但是对于一些城市中的环境（比如城市峡谷，次干道以及隧道），GPS 信号会被严重的干扰和屏蔽使得在这些场景的定位效果会较差，定位误差会超出 10 米^[6, 7, 10]。为此国内外很多学者运用激光扫描仪或者声呐采集点云用来表征周围环境，并根据合适的点云匹配算法进行定位和导航。这种方法的一个非常明显的缺点就是运用点云来表征周围环境而生成的地图数据量和计算量都过于庞大^[11]，通常无法满足系统的实时性需求。之后 Nucher 通过结合 kd-tree 以及点云数据的降采样对点云数据进行处理降低了点云的数据量从而降低了部分计算量^[12]。许多学者还通过将点云地图转换为占据栅格地图（occupancy grid map）来减少整体的计算量^[13]。因为整个空间的地图都要存储在机器人本体上，所以即使是二维的栅格地图同样数据量庞大并且需要花费大量的系统资源用于算法的相关运算。在 2007 年，Fairfield 团队提出通过采用八叉树模型来进一步简化占用栅格地图，并且运用该方法完成了一个水下隧道建模的项目^[14]。

计算机视觉技术的引入在近几年也受到国内外学者的广泛关注^[15]。不同的基于特征的视觉算子 SIFT（Scale Invariant Feature Transform）以及 PIRF（Position Invariant Robust Feature）都被用在复杂的室外环境探索中^[16, 17]。但是这些算法需要的不可预测的较长计算时间使得他们的实时性无法得到保证。基于外包地图

(Outsourcing map based)的方法是另一种估计机器人位置的方法。Christian 等人提出了一种在 GPS 信号不稳定或被屏蔽时利用道路网信息和高程信息估计来定位的方法^[18]。我们实验室朱老师带领的研究团队也提出了一种通过先验地形地图进行定位的方法^[19, 20]。Majdik 在 2015 年提出了一种基于外包地图的定位方法, 通过将小型无人机采集的图像同街景图像进行匹配来对无人机进行定位和跟踪同时通过将采集的图像投影到事先制作的三维模型上对整个周围环境进行细致的建模, 整个系统的运行不依赖于 GPS 信号。但是整个系统的计算复杂度仍然较高并不能满足实时性的要求^[21]。

然而, 对于移动机器人在户外大场景应用的探索而言以上的方法都还是存在明显的缺陷和不足的。应用在救援和军事等领域的移动机器人都是需要在较大的工作环境中长时间工作的。移动机器人在大场景当中工作的一大挑战就是由于周围环境的复杂以及整体规模之大, 产生的大量数据需要处理, 对于这大量数据的存储和相应的计算都是相对困难的^[22]。比如, 分别赢得 DARPA 挑战赛和城市挑战赛冠军的 Stanley 号和 Boss 号无人驾驶车都装备了 6-10 个计算机和超过 1TB 的硬盘来计算和存储系统运行产生的数据^[23, 24], 如图 1-1 和图 1-2 所示。为了减少数据量, Lankenau 提出了一种基于混合拓扑-标尺地图来表征环境的定位方法, 但是该方法只适用于结构化的非复杂环境^[25]。为了解决在大范围非结构化复杂环境中的定位问题, Bonev 等人提出了一种基于机器学习的视觉定位方法^[26]。Bradley 等人提出了从图像中提取“带权重的梯度方向直方图”特征, 并通过与事先采集的图像路标匹配这些特征来进行定位^[27], 这种方法有效地减小了计算量但是在户外大场景中的许多视觉上相似的场景以及光照变化会使得图像匹配算法得出错误或模糊的匹配结果从而导致定位失败^[28]。许多基于视觉的定位算法都会由于环境的变化而导致定位失败。

近年来许多学者尝试克服移动机器人自主在户外大场景长时间连续作业的难度。Zhao 等人提出了一套基于激光的 SLAM(simultaneous localization and mapping, 同时定位与制图)算法来同时在动态的环境中检测和跟踪移动的物体, 但是作者本人发现该方法在追踪静止或者移动的物体时耗时较多。Badino 等人提出了一种基于特征改变预测的新的概念方法来学习环境的变化情况, 利用这种学习方法事先对周围环境进行分析和建模, 然后利用环境模型信息来预测环境中不同条件对于特征产生的影响^[29]。这种方法最大的问题就在于对于存储空间的要求过大, 因为需要存储机器人可能运行的场景的不同环境状况的信息和地图。一般来讲, 对于在城市中道路密度较高的地区, 通常需要超过 1G 的空间来存储仅仅 6 平方千米的地图信息。因此, 对于许多机器人, 尤其是低成本的机器人来讲, 存储一个

机器人可能运行的大场景的地图信息，比如整个城市，是不现实的。另一方面，如果实时环境有所改变，对于机载的嵌入式处理器来讲，及时地更新这些地图信息也将是非常困难的。

云机器人作为 Kuffner 教授在谷歌首先提出的一个新的概念，对于以上移动机器人户外大环境探索面临的许多问题提供了一个非常有希望的解决方案^[30, 31]。云机器人把云计算的概念运用到机器人领域，机器人可以将部分没有实时性要求的任务分配到云端服务器计算以减轻自身的负担，还可将实验过程产生的大量数据信息存储在云端服务器，也能够通过服务器运用网络上丰富的资源。目前只有少量的云机器人相关文献，现有的文献主要的应用领域有基于云端的同步定位与制图，物体跟踪和识别以及多机器人协同作业的研究。Arumugam 为了解决同步定位与制图 SLAM 过程中地图数据量过大和计算负荷较高这一系列问题而提出了一套基于云计算的架构“DAvinCi”，借助于云端服务器的强大计算能力和存储能力来显著提高 SLAM 算法的运行时间^[32]。Riazuelo 等人也提出了一种基于云端的视觉 SLAM 方法，将计算复杂度较高的地图优化和大量的地图数据都转移到云端，从



图 1-1 2005 年 DARPA 挑战赛冠军 Stanley 号无人车

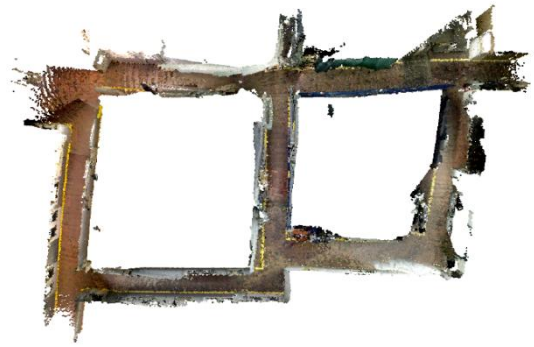


图 1-2 赢得 2007 年 DARPA 城市挑战赛冠军的 Boss 号无人车

而使得机器人本体极度轻量化，只包含了一个普通摄像头和低端的嵌入式处理器^[33]。Kehoe 设计并实现了一套基于云端机器人概念的识别与抓取日常物体的系统^[34]。多机器人协同作业是另一个能够从云机器人概念获益的领域^[35, 36]。Hu 教授的研究团队提出了一个云机器人的架构来解决之前网络机器人所遇到的一系列问题并讨论了对于一些特别情况的应对方法以及云机器人更多的潜在应用领域^[35]。为了提升多机器人协同工作的效率，各个机器人之间需要信息共享，Wang 教授等人提出了利用一套云机器人系统使得多台装备较差的机器人能够在一个复杂环境中协同进行定位^[36]。另一个很重要的方面在于，引入云端服务器将大幅地降低机器人平台的硬件要求，使得装备低端简单的低成本机器人同样能出色的完成复杂的探索任务。Mohanarajah 团队提出了一套运用多台连接到云端服务器的低成本低端机器人协同采集数据并传输到云端，如图 1-3 所示，并在云端完成复杂的三维环境建模的云机器人系统^[37]。Barbosa 也通过将一款运行 Rosjava 应用的低端安卓设备连接至云端来完成一些相对复杂的任务^[38]。



a) Mohanarajah 团队使用的基于云端的低成本机器人



b) 在云端完成的三维环境建模

图 1-3 Mohanarajah 等人提出采用云机器人架构完成室内三维环境建模

1.3 本课题主要研究内容

本课题主要提出了一种基于云端的移动机器人户外大场景定位方法。整个系统主要分为离线过程以及在线过程两个部分。离线过程主要包括提取道路网信息以及采集带有位置信息的参考图像。前者是为了提取新的道路网信息并将其更新至云端的服务器中以备定位使用。参考图像数据库中的每幅图像都包含一个道路网上的地理坐标与其相对应，用于通过视觉补充提高机器人的定位精度。所用到的地图信息都可以通过谷歌地球、OpenStreet Map 等开源地理信息软件中获取。在线过程主要是基于云端的移动机器人定位过程，通过道路网信息计算机器人倾斜模型 (RTI model)，采用粒子滤波融合 RTI 模型信息、传感器数据以及云端视觉补

充结果得出准确的定位结果。同时，对于机器人和云端通信所产生的时间延迟而导致的定位误差，本文也采用了有效的误差补偿算法对其进行了补偿。

本课题主要的贡献在于以下三个方面

(1) 本文提出的方法能够使得移动机器人在不同的复杂户外大场景中进行实时精准的定位，并且在定位过程中不依赖于 GPS 信号。

(2) 本文提出的方法允许移动机器人将大量的道路信息存储在云端的服务器中，从而大大降低了对移动机器人平台自身嵌入式系统的硬件要求。同时，针对由于移动机器人和云端服务器交互数据产生的网络时间延迟也进行了补偿，消除其在定位过程中产生的不良影响。

(3) 当机器人行进在相对较长、角度变化较小且平坦的道路上时，由于姿态变化较小以及传感器的累计误差，常常会导致定位结果不准确，产生模糊的定位结果。本文提出的视觉补充和感知复位方法能够有效地解决这一问题。

1.4 本文结构

第一章，介绍了课题的研究背景和意义，并对当前主流的户外移动机器人定位方法进行了总结和归纳，分析了其优点与缺点。之后对国内外的研究学者对于该领域所做的研究进行了简要分析和介绍。最后介绍了本课题的主要研究内容及主要的贡献。

第二章，本章主要从宏观的角度阐述了整个系统的构成，系统主要分为离线过程和在线过程两个部分。并对各个部分的主要功能以及系统的运作模式进行了详细的说明。

第三章，主要阐述了运行于云端服务器部分的算法和相应的功能。

第四章，主要阐述了运行于移动机器人端的算法和相应的功能。

第五章，本章主要对本课题的主要研究内容进行了综合实验设计。并对实验结果进行了详细地分析和讨论，同时和其他方法也做了详细的对比。

第 2 章 系统总体结构

本文提出的基于云端的移动机器人户外大场景定位系统结构主要包含两个组成部分：离线过程和在线过程，如图 2-1 所示。本章将分别对这两个部分进行系统地说明。

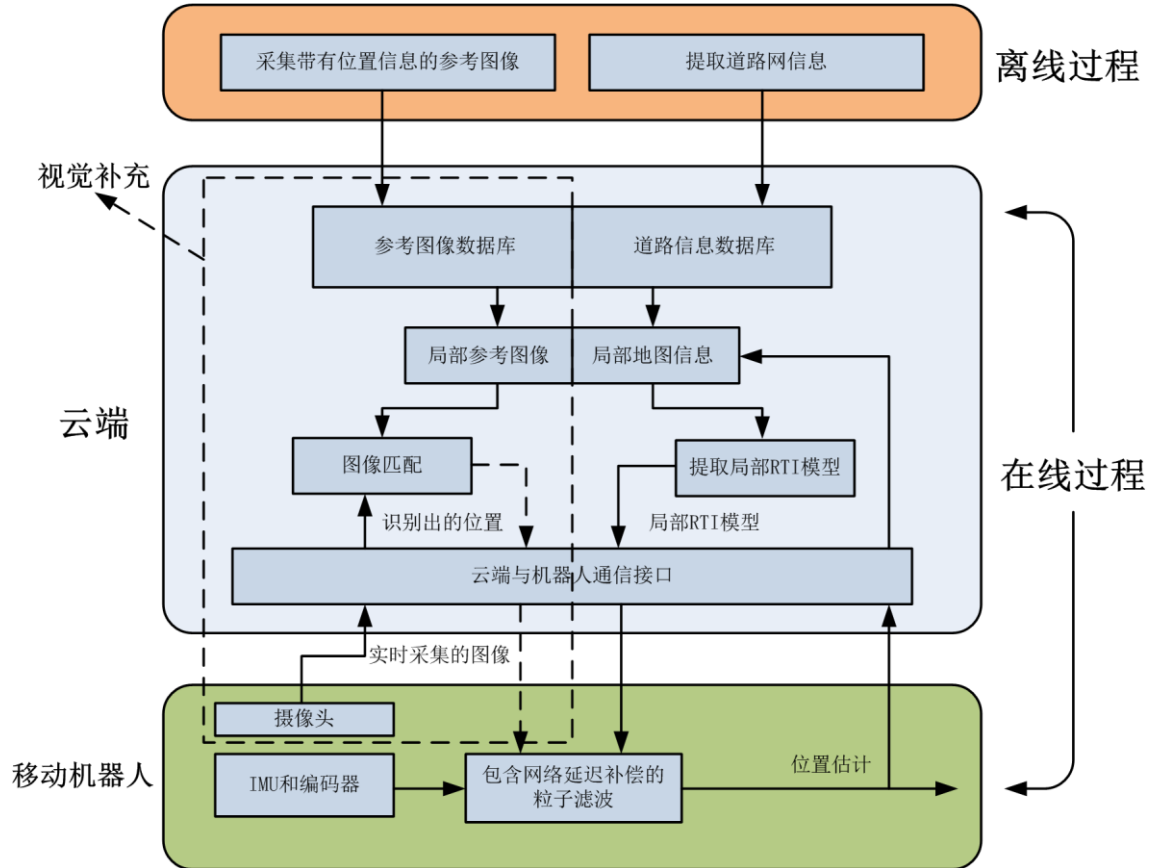


图 2-1 系统结构框图

2.1 离线过程

离线过程主要是在实验之前需要做的准备工作。如图 2-1 所示主要包含两部分内容：提取道路网信息以及采集带有位置信息的参考图像。

2.1.1 道路网信息的提取

本文的定位算法需要融合道路相关信息。所以需要在离线过程先提取道路网信息。提取道路网信息的过程如下。

首先，从谷歌地球或者其他开源地理信息系统(GIS)中将道路信息提取出来，

通常这些软件中已经整合好并包含主要的道路信息，可以将这些信息以 KML 形式的格式导出。KML 是标记语言（Keyhole Markup Language）的缩写，主要基于 XML 语法与格式，适用于用于存储和描述地理信息（包括点、线、图像、多边形和模型等）。KML 文件包含道路信息的经纬度、高程等信息。对于部分软件数据库中不包含的次干道和支路信息，如果有需要的话可以采用手动提取出来并以相同的形式导出。之后将导出的 KML 文件导入到地理信息处理软件当中（比如 ArcGIS）将之前提取的道路信息细分为道路参考点的形式，参考点的密度越大，间隔越小，所能达到的定位精度也会越高。如图 2-2 所示为后文中实验一的道路网参考点示意图，这里为了清楚形象的表示道路参考点而刻意降低了参考点密度，实际密度要大于图示中的参考点密度。经过处理过后的道路网信息将存储在云端服务器上以供在线定位过程使用。当道路信息发生改变时，可以在云端服务器上方便地更新道路网信息。

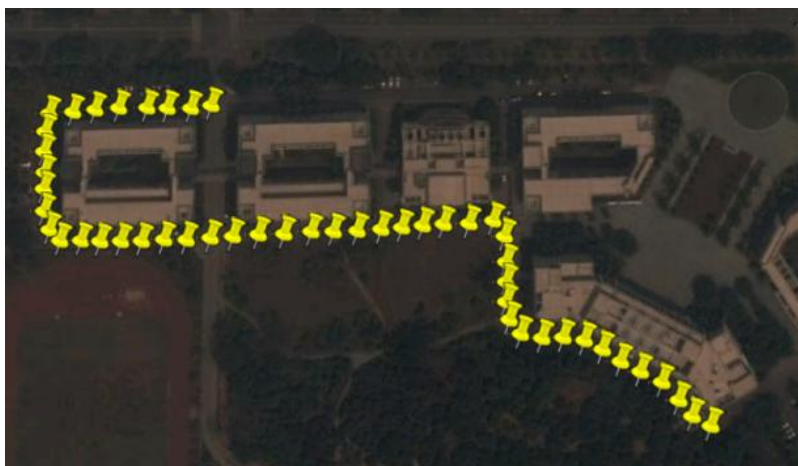


图 2-2 道路网参考点示意图

2.1.2 采集带有位置信息的参考图像

离线过程中的另一部分重要内容是采集带有位置信息的参考图像。为了提高移动机器人的定位精度，并且在整个实验过程都保证相对较低的位置估计误差，在本系统中引入了视觉补充，即通过将移动机器人实时采集的图像信息同事先采集的参考图像信息进行匹配以确定机器人的实时位置。关于视觉补充的详细过程和原理将在下一章节进行说明。本节主要说明用于视觉补充过程的参考图像数据库的构建。

参考图像数据库由在道路网上特定位置采集的图像组成。每一幅参考图像都有一个位置信息与之相对应。为了提高图像匹配的可靠性以及精度，同时易于明确参考图像的位置信息，参考图像都具有明显且易于识别的特征，比如建筑物的

边缘，地面上有明显标志物以及色差等等，如图 2-3 所示，图 a)为包含建筑物的边缘，图 b)在地面上有明显的颜色变化，图 c)为具有特征的易识别的物体（雕像），图 d)为交叉路口。拍摄这些图像的位置信息都便于在地理信息软件中确定。对于现有的地理信息软件图像数据库没有覆盖的区域，可以采用人工采集图像的方法收集参考图像。为了获取具有足够特征的图像，图像的拍摄角度最好朝向路的侧面并且保证摄像头的俯仰角在距水平方向 10-20 度之间。收集的图像及其对应的位置信息将会上传至云端服务器的参考图像数据库中以备在线定位过程使用。当场景信息存在较大变化时，可以在云端服务器上方便地更新参考图像信息。



a) 建筑物边缘



b) 地面有明显的色差



c) 具有特征的易识别的物体



d) 具有多种纹理的交叉路口

图 2-3 参考图像示意图

2.2 在线过程

在线过程是指基于云端的移动机器人定位过程。根据图 2-1 所示，在线过程主要包含两个端子：云端服务器与移动机器人。下面小节分别说明两个端子的功能组成部分以及在线过程的运行方式。

2.2.1 云端服务器

云端服务器主要包含三个部分的内容：视觉补充部分以及提取和计算机器人地形倾斜度模型（RTI）^[19, 20]部分以及和移动机器人通信接口部分。

通信接口部分主要负责和移动机器人之间的信息交互。提取和计算 RTI 模型部分主要负责从道路信息数据库中检索移动机器人需要的局部地图信息，并根据

这些局部地图信息提取和计算局部机器人倾斜模型 (RTI)。在视觉补充部分, 首先从参考图像数据库中根据移动机器人的位置检索出位置信息在移动机器人附近的局部参考图像, 然后将移动机器人实时采集的图像与局部参考图像进行匹配, 并将匹配结果 (位置坐标信息) 返回给移动机器人。

2.2.2 移动机器人

移动机器人部分主要包含摄像头、IMU 和编码器以及包含网络延迟补偿的粒子滤波部分。摄像头会实时采集图像并发送给云端服务器, IMU 和编码器用于测量移动机器人的姿态和线速度。包含网络延迟补偿的粒子滤波用于融合局部 RTI 模型、云端反馈视觉补充信息以及 IMU 和编码器测量数据来得出位置估计结果。

2.2.3 在线过程运行方式

在线过程的运行方式如下所述。本文假设移动机器人只在道路网覆盖区域运行。当机器人行进在道路上时, 首先将机器人的初始位置发送至云端服务器, 确定这个初始位置的方法很多, 比较直接的是采用 GPS 信号得出初始位置, 也可以通过地理信息软件测量机器人的初始位置。这个初始位置不需要非常精确, 云端服务器在接收到这个初始位置之后将会自动检索该位置周围半径为 δ 区域的局部道路网信息以及参考图像信息。相应路段的局部 RTI 模型将被提取计算出来并发送给移动机器人, 移动机器人将采用粒子滤波算法融合局部 RTI 模型信息和机载 IMU 以及编码器数据进行定位。同时, 移动机器人通过安装的摄像头实时采集周围环境的图像信息并将其传送至云端服务器用于视觉补充, 云端服务器接收图像并将其与之前提取的局部参考图像进行匹配。当图像匹配成功时, 相应参考图像的位置信息将会被发送至移动机器人端, 该信息也将作为测量值通过粒子滤波和局部 RTI 模型以及传感器数据一起估计机器人当前的位置。

在本系统中, 数据量较大的参考图像信息以及道路网信息都存储在云端服务器, 视觉补充以及 RTI 模型的检索和计算都运行于云端服务器。从而大幅地降低了移动机器人本体的存储和计算负荷, 同时也使得机器人无论运行在多大的户外环境中都不会受到这两个方面对于机器人本身硬件的限制。

2.3 本章小结

本章主要从宏观的角度阐述了整个系统的构成、各个部分的主要功能以及系统的运作模式。整个系统分为离线过程和在线过程两个部分。离线过程主要包含提取道路网信息以及采集带有位置信息的参考图像两个部分。在线过程主要是基

于云端的移动机器人定位过程。接下来又对整个系统的运行过程进行了阐述。后面的章节将具体说明每个部分采用的具体算法和实现方法。

第 3 章 云端服务器部分算法

在本系统中，大量的计算和存储负荷将会由云端服务器承担从而使得移动机器人在无论多大的环境中运行都不会承受过大的计算和存储压力并可以保证其轻量化。本章将对云端服务器在整个系统中的作用和运行的算法进行详细的阐述，主要包含坐标转换算法，地形倾斜模型（RTI model）的提取和计算以及图像匹配算法。

3.1 道路网信息的处理

3.1.1 坐标转换算法

在上一章介绍的离线过程中提取的道路网信息是在大地坐标系下的，是用经纬度和高程来表示位置信息的。高程 h 定义为以椭球面为基准面，由地面点沿其法线到椭球表面的距离。经度 λ 定义为地球表面上一点与两极的连线与本初子午线所在平面的夹角。纬度 φ 定义为地球表面上一点与地球球心的连线和赤道平面之间的夹角，取值范围在 0 到 90 度之间。位于赤道以南的所有点的纬度记为南纬，位于赤道北部的所有点的纬度记为北纬。但是由于经纬度的尺度过大，经纬度变换一度通常对应相当大的一段距离，所以大地坐标系不适用于较精确的定位算法当中，这就需要进行坐标转换。本文定位算法选择在室外定位较为常用的东北天坐标系下进行。两个坐标系的示意图如图 3-1 所示。

在大地坐标系和东北天坐标系的转换过程中需要经过一个中间坐标系的过渡——地心空间直角坐标系。在将大地坐标转换为东北天坐标时，需要先将大地坐标转换为地心空间直角坐标系，然后将转换过来的地心空间直角坐标系转换为东北天坐标系。反之亦然。

该坐标系是以地球的质心作为原点的坐标系。它的 X 轴指向格林尼治子午面与地球赤道的交点， Z 轴朝向地球北极， Y 轴垂直于 X - O - Z 平面构成右手坐标系，如图 3-1 所示。

本地东北天坐标系定义为以地球表面的某一点作为原点，过该点的切面为本地东北天坐标系的 XOY 平面，其中 X 轴正方向定义为朝向东方， Y 轴正方向定义为朝向北方，切面的朝上的法向量方向定义为 Z 轴正方向。

大地坐标（经度 λ ，纬度 φ ，高程 h ）转换为地心空间直角坐标 (X,Y,Z) 的方式如下^[19]：

$$X = (N(\varphi) + h) \cos \varphi \cos \lambda \quad (3-1)$$

$$Y = (N(\varphi) + h) \cos \varphi \sin \lambda \quad (3-2)$$

$$Z = [N(\varphi)(1 - e^2) + h] \sin \varphi \quad (3-3)$$

$$N(\varphi) = \frac{a}{\sqrt{1 - e^2 \sin^2 \varphi}} \quad (3-4)$$

其中, X, Y, Z 为地心空间坐标对应的坐标值, λ 为经度, φ 为纬度, h 为高程, e 为椭球第一偏心率, $e^2 = 0.00669437999013$, a 为椭球长半径, $a = 6378137m$ 。

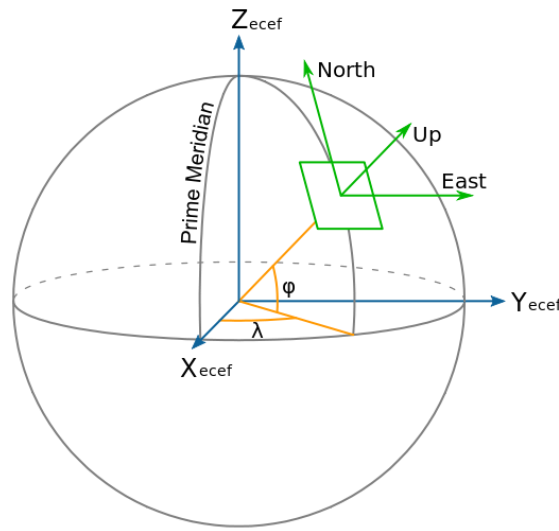


图 3-1 坐标转换示意图

之后再将转换得出的地心空间坐标系转换为东北天坐标系, 首先选择一个参考点 (X_r, Y_r, Z_r) 作为原点, 则空间坐标系下的任一点 (X_p, Y_p, Z_p) 转换以 (X_r, Y_r, Z_r) 为原点的东北天坐标系的坐标 (x, y, z) , 如下所示:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -\sin \lambda_r & \cos \lambda_r & 0 \\ -\sin \varphi_r \cos \lambda_r & -\sin \varphi_r \sin \lambda_r & \cos \varphi_r \\ \cos \varphi_r \cos \lambda_r & \cos \varphi_r \sin \lambda_r & \sin \varphi_r \end{bmatrix} \begin{bmatrix} X_p - X_r \\ Y_p - Y_r \\ Z_p - Z_r \end{bmatrix} \quad (3-5)$$

其中, λ_r 和 φ_r 分别为参考点的经纬度。

将东北天坐标转换回大地坐标只需经过一个以上过程的逆过程, 如下所示:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} -\sin \lambda_r & -\sin \varphi_r \cos \lambda_r & \cos \varphi_r \cos \lambda_r \\ \cos \lambda_r & -\sin \varphi_r \sin \lambda_r & \cos \varphi_r \sin \lambda_r \\ 0 & \cos \varphi_r \sin \lambda_r & \sin \varphi_r \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} \quad (3-6)$$

在将空间直角坐标转换为大地坐标时, 计算方法如下:

$$\lambda = \arctan(Y / X) \quad (3-7)$$

$$\varphi = \arctan\left\{\frac{Z + be' \sin^3 U}{\sqrt{X^2 + Y^2 - ae^2 \cos^3 U}}\right\} \quad (3-8)$$

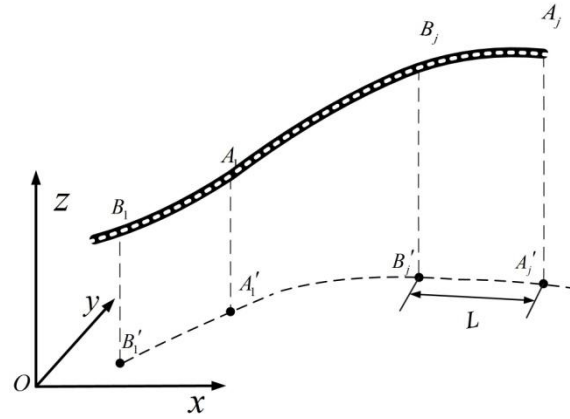
$$h = \frac{Z}{\sin \varphi} - N(\varphi)(1 - e^2) \quad (3-9)$$

$$U = \arctan\left(\frac{Za}{\sqrt{X^2 + Y^2}b}\right) \quad (3-10)$$

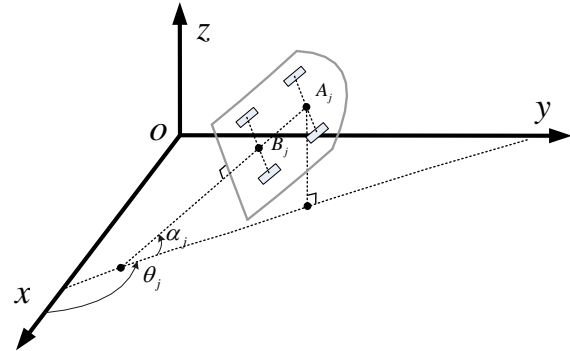
$$b = \sqrt{(1 - e^2)a^2} \quad (3-11)$$

其中, e' 为椭球第二偏心率, $e'^2 = 0.00669437999013$ 。

3.1.2 机器人地形倾斜模型 (RTI model) 的计算



a) 机器人行走的一段路径



b) 计算一个位置的 RTI 模型

图 3-2. 地形倾斜模型 (RTI model) 的计算

机器人地形倾斜度模型^[19, 20]是利用道路网信息来建立行进在道路网上的移动机器人的姿态信息和机器人所在的道路信息之间的联系的工具模型。假设 $B_1A_1B_2A_2$ 是道路网信息中的一段, 如图 3-2a)所示。道路网信息的提取在上一章的离线过程中有所阐述, 首先用上一章提到的坐标转换方法将道路点信息从大地坐标系 (经

度，纬度，高程）转换到东北天坐标系下 x, y, z 。然后做这些点在 $x-O-y$ 平面上的投影 $B'_j A'_j B'_j A'_j$ 。这些投影点之间的间隔为 L/n ，其中 L 表示为移动机器人的长度，参数 n 可以根据定位精度的要求进行调整，其取值越大则投影点之间间隔越密集，模型越精确。 $B'_j A'_j$ 表示第 j 个线段。点 B'_j 和点 A'_j 分别是道路点 B_j 和 A_j 的投影。这些点的 z 坐标值刻印通过均匀插值的方法从道路网信息中获取^[39]。当机器人行进在第 j 个路段上时，点 B_j 和 A_j 表示机器人两个前轮与地面接触的中点以及两个后轮与地面接触的中点。

图 3.2b) 表示了一个位置的 RTI 模型的提取。 $\overline{B_j A_j}$ 表示了机器人运动的方向。朝向角 θ_j 被定义为 $\overline{B_j A_j}$ 和 x 轴的夹角。朝向角由路径唯一确定。角 α_j 定义为 $\overline{B_j A_j}$ 方向与 $x-O-y$ 平面的夹角。所以 α_j 可以通过下面的公式获得。

$$\alpha_j = \sin^{-1}((z_A - z_B) / |\overline{B_j A_j}|) \quad (3-12)$$

$$|\overline{B_j A_j}| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2} \quad (3-13)$$

图3.2b)表示了某一个位置的RTI模型的计算。 $\overline{B_j A_j}$ 表示了移动机器人运动的方向。朝向角 θ_j 被定义为 $\overline{B'_j A'_j}$ 和 x 轴的夹角。朝向角由路径唯一确定。角 α_j 定义为 $\overline{B_j A_j}$ 方向与 $x-O-y$ 平面的夹角。所以 α_j 可以通过下面的公式获得。

$$\alpha_j = \sin^{-1}((z_A - z_B) / |\overline{B_j A_j}|) \quad (3-14)$$

$$|\overline{B_j A_j}| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2} \quad (3-15)$$

式中 $B_j = (x_B, y_B, z_B)$ ， $A_j = (x_A, y_A, z_A)$ 。因此，一系列的角度 (θ_j, α_j) 可以从路段 $B'_j A'_j$ 中计算出来。因为移动机器人在某一个路段上行进的过程中自身的朝向角应该和道路的方向角度一致，所以移动机器人在每个道路线段的位置 (x_j, y_j, z_j) 都会与角度 (θ_j, α_j) 相匹配。通过上述的线性插值关系，可以计算得出 $[\theta_k \ \alpha_k]^T = R T I \ M o(d \ \ell, l \ \kappa x, \ k=1, 2, \dots, N$ ，其中 N 可以根据所需定位精度进行调整。这样就建立起了移动机器人自身的姿态信息和道路信息之间的联系。

3.2 视觉补充

在参考图像数据库中，每一幅图像都对应着一个道路网地图中的位置信息（经度，纬度，高程）。移动机器人实时采集的图像会与存储在云端的参考图像进行匹配。当出现成功的匹配时，相应的参考图像的坐标将会发送给机器人用于更新位置估计。图像匹配算法是基于加速鲁棒特征的（SURF）^[40]。从图像中提取基于

U-SURF 的非旋转不变性特征及其 64 维描述子, 采用 U-SURF 的原因在于移动机器人每次经过采样点时的相对姿态是几乎不变的, 机体本身并不会出现翻转等特殊的情况。另一方面, 采用 U-SURF 也会显著减少匹配时间, 提高系统实时性。

本文的匹配过程采用了双向匹配的方法。首先分别采用基于 U-SURF 的方法提取和计算移动机器人实时采集的图像(下文中以实时图像简称)和云端服务器中的参考图像(下文中简称参考图像)的特征点和描述子。然后对于实时图像中的每一个 64 维特征点的描述子, 都在参考图像中找到与其欧氏距离最近的两个特征描述子, 计算这两个欧氏距离中较小的一个距离和较大的一个距离的比值记为 p , 我们定义一个阈值 r , 如果 $p \leq r$ 则说明两个距离相差较大, 所以我们保留较小的那个距离对应的匹配, 剔除较大的那个距离对应的匹配, 因为较大距离对应的匹配次于较小距离对应的匹配, 并且两者差距较大, 说明较大距离对应匹配有极大的概率是错误的匹配; 类似的道理, 如果 $p > r$, 这说明两个距离非常的接近, 此时, 无论我们接受或者拒绝其中任何一对匹配, 都有可能是错误的, 产生错误的匹配, 所以, 在此情况之下, 为了保证匹配的质量, 剔除这两个距离相近的匹配。在进行完这一步操作之后, 我们得到了一组相对优质的从实时图像到参考图像的匹配集记为 S_1 。接下来采用同样的方法, 计算出一组从参考图像到实时图像的匹配集 S_2 。接下来我们只保留两个匹配集中一一对应的匹配作为最终的匹配集 S_f , 即 S_f 中的匹配既是从实时图像到参考图像的最佳匹配同时也是参考图像到实时图像中的最佳匹配。

经过以上的双向匹配以及滤波操作, 大量的错误匹配都会被过滤掉。最后, 需要根据这个优质匹配集 S_f 计算基础矩阵用于检验这些匹配是否满足极性约束, 由于计算基础矩阵需要至少 8 对匹配, 但是最佳匹配往往可能多余 8 个, 此时采用随机采样一致(RANSAC)方法计算基础矩阵。如果通过基础矩阵的计算得出结论匹配满足极性约束, 则说明两幅图像匹配成功, 这个图像所代表的位置被识别出来了从而相应的地理坐标将会发送至移动机器人端用于修正位置估计。

3.3 本章小结

本章主要说明了云端服务器部分的算法, 主要包含道路网信息的处理和视觉补充两个部分。道路网信息的处理主要包括坐标转换算法和地形倾斜度模型(RTI model), 由于经纬度不适于用作定位算法的坐标表示, 坐标转换算法主要是为了完成由经纬度和高程表示的大地坐标与东北天坐标之间的转换; 接着在东北天坐标系下计算地形倾斜度(RTI model)模型。视觉补充主要包含基于 SURF 图像匹配

算法。下一章将对移动机器人端的算法进行说明。

第 4 章 移动机器人部分算法

移动机器人部分主要包括摄像头、IMU（Inertial Measurement Unit, 惯性测量单元）和编码器。摄像头会实时采集图像并发送给云端服务器，IMU 和编码器用于测量移动机器人的姿态和线速度。包含网络延迟补偿的粒子滤波用于融合局部 RTI 模型、云端反馈视觉补充信息以及 IMU 和编码器测量数据来得出位置估计结果。本章将详细说明各个部分。

4.1 粒子滤波算法

粒子滤波算法来源于蒙特卡洛方法（Monte Carlo Method），它利用粒子集来表示概率，并且可以运用在任何形式的状态空间模型上，无论该状态空间是线性的或者非线性的。与卡尔曼滤波和扩展卡尔曼滤波方法相比，粒子滤波表示的概率分布范围更大。其核心思想是通过从后验概率中抽取的随机状态粒子来表示其分布情况，是一种顺序重要性采样法。粒子滤波经常应用在户外移动机器人定位中^[41-43]。

粒子滤波算法的伪代码如表 4-1 所示。其中粒子集表示如下：

$$\chi_t := x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]} \quad (4-1)$$

M 表示粒子集中粒子的个数。粒子集中的一个粒子 $x_t^{[m]}$ 表示物体在 t 这一时刻的一个实例，如果是应用在移动机器人定位领域，可以用来表示机器人在 t 时刻的一个可能的位置。对于每一个粒子 $x_t^{[m]}$ ，都可以依据当前时刻 t 计算出一个对应的权重，记为 $w_t^{[m]} = p(z_t | x_t^{[m]})$ 。如果粒子 $x_t^{[m]}$ 的值和机器人的真实位置非常接近，则对应的权重 $w_t^{[m]}$ 就会比较大，反之，权重就会相对较小。

重采样（见表 4-1 第 7-10 行）是粒子滤波算法非常重要的一个步骤，核心思想就是通过迭代计算，淘汰权重较小的粒子，并将其用权重较大的粒子代替，重采样过后粒子的总数不变。也就是说越是接近于移动机器人真实位置的粒子越容易保留在粒子集中。

表 4-1 粒子滤波算法流程^[44]

算法流程	
1	$\bar{\chi}_t = \chi_t = \emptyset$
2	from $m = 1$ to M do
3	sample $x_t^{[m]} \sim p(x_t \mu_t, x_t^{[m-1]})$

表 4-1 (续表)

算法流程	
4	$\omega_t^{[m]} = p(z_t x_t^{[m]})$
5	$\bar{\chi} = \chi_t + \langle x_t^{[m]}, \omega_t^{[m]} \rangle$
6	endfor
7	from $m = 1$ to M do
8	draw i with probability $\propto \omega_t^{[i]}$
9	add $\omega_t^{[i]}$ to χ_t
10	endfor
11	return χ_t

4.2 基于粒子滤波的定位算法

本课题利用存储在云端的道路网信息以及参考图像信息来辅助判断机器人的位置。当移动机器人在道路网上行驶时，采用粒子滤波融合存储在云端的信息以及实时测得的机器人的姿态（yaw 角，pitch 角）和速度估计出机器人的位置。

4.2.1 运动模型

当移动机器人行进在道路网上时，它的姿态和路面的地形是存在一个相关关系的。移动机器人的 yaw 角和道路的 yaw 角是相等的，这里为了简化，我们假设移动机器人并不会在道路上翻转，而是一直沿着路面行进，这种假设也是基本符合实际情况的，pitch 角会随着路面的坡度而变化。如图 4-1 所示：

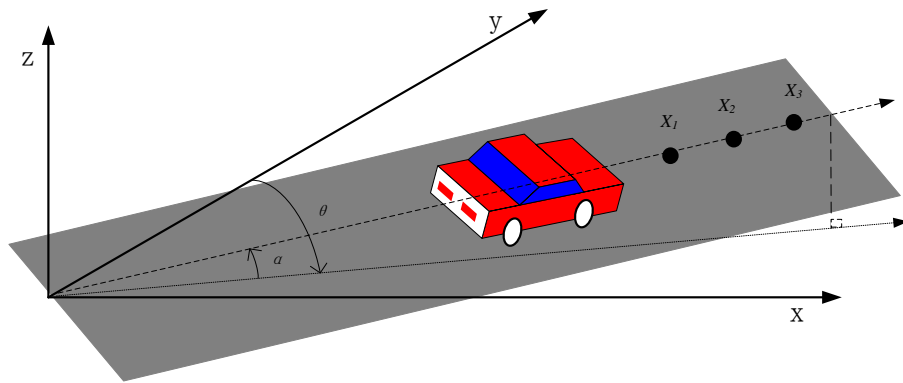


图 4-1 车体沿路行驶的位姿

在移动机器人的起始位置建立东北天坐标系，用 x, y, z 表示移动机器人在东北天坐标系中的位置。则 $\chi_t^{[m]} = [x_t^{[m]}, y_t^{[m]}, z_t^{[m]}]^T$ 表示第 m 个粒子在 t 时刻的位置。本文

假设移动机器人紧贴路面行驶并且本身不存在翻转和侧移，则运动模型如下：

$$\chi_t^{[m]} = \begin{bmatrix} x_t^{[m]} \\ y_t^{[m]} \\ z_t^{[m]} \end{bmatrix} = \begin{bmatrix} x_{t-1}^{[m]} + \cos \theta_t \cdot \cos \alpha_t \cdot v_t \cdot T \\ y_{t-1}^{[m]} + \sin \theta_t \cdot \cos \alpha_t \cdot v_t \cdot T \\ z_{t-1}^{[m]} + \sin \alpha_t \cdot v_t \cdot T \end{bmatrix} \quad (4-2)$$

其中， $x_{t-1}^{[m]}, y_{t-1}^{[m]}, z_{t-1}^{[m]}$ 表示上一时刻 $t-1$ 时移动机器人的位置， T 为系统采样时间， θ_t 为 t 时刻移动机器人上的 IMU 测得的 yaw 角， α_t 为 IMU 测得的 pitch 角， v_t 为车体前进方向上的速度。

4.2.2 测量值模型

在上一章阐述的地形倾斜模型（RTI model）建立起了移动机器人的姿态和道路网信息之间的联系，在这里采用 RTI 模型作为移动机器人测量值的估计。对于第 m 个粒子在 t 时刻的测量预测值为：

$$\hat{Z}_t^{[m]} = \begin{bmatrix} \hat{\theta}_t^{[m]} \\ \hat{\alpha}_t^{[m]} \\ \hat{d}_t^{[m]} \\ \hat{V}_t^{[m]} \end{bmatrix} = \begin{bmatrix} RTI_{\theta}(x_t^{[m]}, y_t^{[m]}, z_t^{[m]}) \\ RTI_{\alpha}(x_t^{[m]}, y_t^{[m]}, z_t^{[m]}) \\ dist[p(x, y, z), p(x_t^{[m]}, y_t^{[m]}, z_t^{[m]})] \\ \mathbf{0} \end{bmatrix} \quad (4-3)$$

RTI_{θ} 和 RTI_{α} 代表从云端服务器下载的局部地形倾斜度 RTI 模型。 $\hat{d}_t^{[m]}$ 表示粒子 m 和点 $p(x, y, z)$ 之间的欧式距离， $p(x_t^{[m]}, y_t^{[m]}, z_t^{[m]})$ 表示粒子 m 的位置， $p(x, y, z)$ 表示道路网中距离粒子 m 最近的道路点的位置。 \hat{V}_t 表示云端视觉补充过程成功之后，云端服务器返回的移动机器人在大地坐标系下位置的测量预测值，其他时刻返回零向量。由于视觉补充成功的过程涉及网络延时，所以关于这部分详细将在 4.3 节网络延迟补偿部分阐述。

传感器测量模型为：

$$Z_t = \begin{bmatrix} \theta_t \\ \alpha_t \\ d_t \\ V_t \end{bmatrix} \quad (4-4)$$

其中， θ_t 和 α_t 分别表示移动机器人机体上安装的传感器测量的 yaw 角和 pitch 角， d_t 是一个虚拟测量值用于保证所有的粒子都是位于道路点附近^[18]。 V_t 表示云端服务器传送至移动机器人端的视觉补充位置识别结果。

4.2.3 权重计算及重采样

在明确了运动模型和测量模型之后，可以根据其计算每个粒子的权重，计算的步骤分以下两步进行：

$$Z_t - \hat{Z}_t^{[m]} = \begin{bmatrix} \theta_t - \hat{\theta}_t^{[m]} \\ \alpha_t - \hat{\alpha}_t^{[m]} \\ d_t - \hat{d}_t^{[m]} \\ V_t - \hat{V}_t^{[m]} \end{bmatrix} \quad (4-5)$$

$$w_t^{[m]} = |2\pi Q_t|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (Z_t - \hat{Z}_t^{[m]})^T Q_t^{-1} (Z_t - \hat{Z}_t^{[m]}) \right\} \quad (4-6)$$

首先计算测量预测值和实际测量值之差，然后根据正态分布计算每个粒子的权重。重采样是粒子滤波算法非常重要的一个步骤，核心思想就是通过迭代计算，淘汰权重较小的粒子，并将其用权重较大的粒子代替，重采样过后粒子的总数不变。也就是说越是接近于移动机器人真实位置的粒子越容易保留在粒子集中。

本文在重采样阶段采用 Low Variance Sampler 算法^[44]。如图 4-2 所示。对于一个包含有 M 个粒子的粒子集，每个长方形的宽度表示粒子的权值，该算法首先选取一个大于 0 且小于等于 M^{-1} 的数 r ，然后以 M^{-1} 为步长向前选取，结果保存在 U 中。从图中可以直观的观察，对应长方形的宽度越大，权值越大，越容易被选中，相反，如果对应长方形的宽度越小，权值越小，则越不容易被选中。能够加入新的粒子集的粒子编号应满足以下条件：

$$i = \arg \min_j \sum_{m=1}^j w_t^{[m]} \geq U \quad (4-7)$$

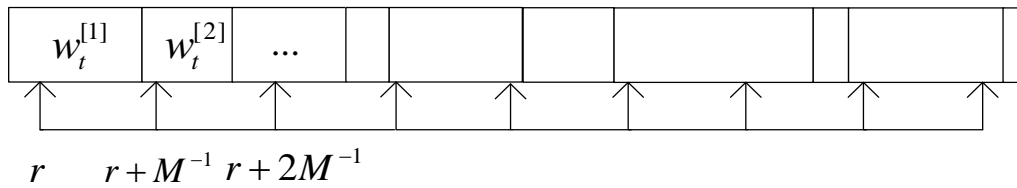


图 4-2 Low Variance Sampler 算法原理^[44]

算法的伪代码如表 4-2 所示：

表 4-2 Low Variance Sampler 算法伪代码^[44]

算法流程	
1	$\bar{\mathcal{X}}_t = \emptyset$
2	$r = \text{rand}(0; M^{-1})$

表 4-2 (续表)

算法流程	
3	$c = w_t^{[1]}$
4	$i = 1$
5	for $m = 1$ to M do
6	$U = r + (m - 1) \cdot M^{-1}$
7	while $U > c$
8	$i = i + 1$
9	$c = c + w_t^{[i]}$
10	endwhile
11	add $x_t^{[i]}$ to $\bar{\chi}_t$
12	endfor
13	return $\bar{\chi}_t$

4.2.4 感知复位

如果由 GPS 估计的初始位置距离真实位置较远，机器人必须重新进行定位。此外，当机器人行进在一段相对较长，平坦且道路特征变化较小并且不具备视觉补充校正的条件时，所有的粒子将会在错误的位置产生聚集。这种情形非常类似于绑架问题而且会导致粒子滤波失效^[45]。因此，为了因对上述状况，基于传感器测量值的感知复位被引入系统来解决这个问题（表 4-3 第 16 行）。如果所有的粒子权重和 $\sum w_i$ 高于某个阈值 K ，所有的粒子将会被保留。否则，视为位置估计错误，通过将传感器数据与局部 RTI 模型进行结合，找到道路网中与机器人姿态最匹配的位置^[46]。在重采样过程中将以此位置为基准添加新的粒子。

4.2.5 定位算法伪代码

基于粒子滤波的定位算法的整体伪代码如表 4-2 所示：

表 4-3 基于粒子滤波的定位算法伪代码

算法流程	
1	$(X_{t-1}, Z_t, v_t, X_{t-k-1}, Z_{\text{backup}}, v_{\text{backup}})$

表4-3 (续表)

算法流程	
2	$X_{t-1} = \langle \chi_{t-1}^{[1]}, \chi_{t-1}^{[2]}, \dots, \chi_{t-1}^{[M]} \rangle, Z_t = \{\theta_t, \alpha_t, d_t, V_t\}, Q_t, \bar{X}_t = X_t = \emptyset,$ $X_{t-k-1} = \langle \chi_{t-k-1}^{[1]}, \chi_{t-k-1}^{[2]}, \dots, \chi_{t-k-1}^{[M]} \rangle, Z_{\text{backup}} = \{Z_{t-k}, Z_{t-k+1}, \dots, Z_t\},$ $V_{\text{backup}} = \{v_{t-k}, v_{t-k+1}, \dots, v_t\}.$
3	if $VI=0$ do
4	for $m=1$ to M do
5	$\chi_t^{[m]} = \begin{bmatrix} x_t^{[m]} \\ y_t^{[m]} \\ z_t^{[m]} \end{bmatrix} = \begin{bmatrix} x_{t-1}^{[m]} + \cos \theta_t \cdot \cos \alpha_t \cdot v_t \cdot T \\ y_{t-1}^{[m]} + \sin \theta_t \cdot \cos \alpha_t \cdot v_t \cdot T \\ z_{t-1}^{[m]} + \sin \alpha_t \cdot v_t \cdot T \end{bmatrix}$
6	$\hat{Z}_t^{[m]} = \begin{bmatrix} \hat{\theta}_t^{[m]} \\ \hat{\alpha}_t^{[m]} \\ \hat{d}_t^{[m]} \\ \hat{V}_t^{[m]} \end{bmatrix} = \begin{bmatrix} RTI_{\theta}(x_t^{[m]}, y_t^{[m]}, z_t^{[m]}) \\ RTI_{\alpha}(x_t^{[m]}, y_t^{[m]}, z_t^{[m]}) \\ dist[p(x, y, z), p(x_t^{[m]}, y_t^{[m]}, z_t^{[m]})] \\ \mathbf{0} \end{bmatrix}$
7	$Z_t - \hat{Z}_t^{[m]} = \begin{bmatrix} \theta_t - \hat{\theta}_t^{[m]} \\ \alpha_t - \hat{\alpha}_t^{[m]} \\ d_t - \hat{d}_t^{[m]} \\ V_t - \hat{V}_t^{[m]} \end{bmatrix}$
8	$w_t^{[m]} = 2\pi Q_t ^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (Z_t - \hat{Z}_t^{[m]})^T \right\} Q_t^{-1} ((Z_t - \hat{Z}_t^{[m]})),$ //权值计算
9	add $\chi_t^{[m]}$ and $w_t^{[m]}$ to \bar{X}_t
10	endfor
11	if $\sum w_i > K$, for $m=1$ to M do
12	draw i with probability $\propto w_t^{[i]}$
13	add $\chi_t^{[i]}$ to X_t
14	endif
15	else
16	replace particles with more reliable ones //感知复位
17	endif
18	return $X_t = \langle \chi_t^{[1]}, \chi_t^{[2]}, \dots, \chi_t^{[M]} \rangle$
19	if $VI=1$ do
20	network delay compensation //网络延迟补偿算法
21	endif

系统状态 X_{t-1} 表示机器人在 $t-1$ 时刻笛卡尔坐标系下的三维位置 (x, y, z) 。 Q_t 是

系统的协方差矩阵。 Z_t 是系统的测量向量。从时刻 $t-k$ 到时刻 t 的 IMU 和里程计的测量值分别存储在 Z_{backup} 和 v_{backup} 中, k 表示机器人和云端通信的网络延迟时间。 Z_{backup} , v_{backup} 以及 X_{t-k-1} 用于进行网络延迟补偿, 这部分内容将在 4-3 小节进行阐述。系数 VI (表 4-3 第 3 行) 表示机器人是否收到云端的视觉补充位置匹配反馈信息。当 VI 为 0 时表示未收到云端反馈信息, 从第 4 行到第 17 行用于进行移动机器人的位置估计。其中第 5 行表示运动模型, 第 6 行表示测量值估计值, 第 7、8 行表示粒子权值的计算。第 9-17 行表示重采样以及感知复位。当 VI 为 1 时表示收到云端反馈, 因为视觉校正补充过程所引入的网络延迟无法被忽略, 网络延迟补偿算法将被执行。

4.3 网络延迟补偿算法

在本文所描述的架构中, 机器人和云端服务器需要交换两种数据, 一种是局部 RTI 模型, 另一种是用于视觉补充的图像信息。其中传输局部 RTI 模型所带来延迟是可以忽略的, 因为移动机器人在行进的过程中, 只要移动机器人本体存有周围环境的局部 RTI 模型信息就可以正常完成定位, 在本文架构中, 移动机器人每行进 l 米, 将会向云端请求以机器人位置为中心, 半径 $2l$ 米范围的 RTI 模型, 所以在这段传输数据的时间内, 机器人本体存储有局部用于定位的 RTI 模型, 所以这种网络延迟并不会影响定位结果。但是另一方面, 在视觉补充过程中, 由于需要实时向云端传送图像信息, 这部分的传输延迟时间将会对系统运行产生显著的影响。因此, 当 VI 等于 1 的时候 (表 4-3 第 19-20 行), 将运行一个网络延迟的补偿算法。该算法的具体内容见表 4-4。

表 4-4 算法: 网络延迟补偿算法伪代码

算法流程	
1	$(X_{t-k-1}, Z_{backup}, v_{backup})$
2	for $i=t-k$ to t do //重新更新网络延迟间隔的系统状态
3	for $m=1$ to M do
4	$\chi_i^{[m]} = \begin{bmatrix} x_i^{[m]} \\ y_i^{[m]} \\ z_i^{[m]} \end{bmatrix} = \begin{bmatrix} x_{i-1}^{[m]} + \cos \theta_i \cdot \cos \alpha_i \cdot v_i \cdot T \\ y_{i-1}^{[m]} + \sin \theta_i \cdot \cos \alpha_i \cdot v_i \cdot T \\ z_{i-1}^{[m]} + \sin \alpha_i \cdot v_i \cdot T \end{bmatrix}$
5	$\hat{Z}_i^{[m]} = \begin{bmatrix} \hat{\theta}_i^{[m]} \\ \hat{\alpha}_i^{[m]} \\ \hat{d}_i^{[m]} \\ \hat{V}_i^{[m]} \end{bmatrix} = \begin{bmatrix} RTI_{\theta}(x_i^{[m]}, y_i^{[m]}, z_i^{[m]}) \\ RTI_{\alpha}(x_i^{[m]}, y_i^{[m]}, z_i^{[m]}) \\ dist[p(x, y, z), p(x_i^{[m]}, y_i^{[m]}, z_i^{[m]})] \\ IL[p(x_i^{[m]}, y_i^{[m]}, z_i^{[m]})] \end{bmatrix}$

表 4-4 (续表)

算法流程	
6	$Z_i - \hat{Z}_i^{[m]} = \begin{bmatrix} \theta_i - \hat{\theta}_i^{[m]} \\ \alpha_i - \hat{\alpha}_i^{[m]} \\ d_i - \hat{d}_i^{[m]} \\ V_i - \hat{V}_i^{[m]} \end{bmatrix}$
7	$w_i^{[m]} = 2\pi Q_i ^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} \left(Z_i - \hat{Z}_i^{[m]} \right)^T Q_i^{-1} \left(Z_i - \hat{Z}_i^{[m]} \right) \right\}$
8	add $\chi_i^{[m]}$ and $w_i^{[m]}$ to \bar{X}_i
9	endfor
10	for $m=1$ to M do
11	draw j with probability $\propto w_i^{[j]}$
12	add $\chi_i^{[j]}$ to X_i
13	endfor
14	endfor
15	return $X_t = \langle \chi_t^{[1]}, \chi_t^{[2]}, \dots, \chi_t^{[M]} \rangle$

假设机器人是在 t 时刻接收到云端反馈的视觉补充位置识别结果。匹配成功的图像是机器人于 $t-k$ 时刻采集的。该网络延迟补偿算法的核心思想就是利用云端反馈的位置识别结果和存储下来的延迟间隔时间段的传感器数据来重新估计系统的状态。当机器人收到云端反馈的结果时，另一个粒子滤波将被启动用于融合云端反馈的位置信息以及存储的传感器数据，从而根据这些数据重新估计系统状态，见表 4-4。算法的更新的过程类似于表 4-3。主要的差别在于引入了测量函数 $IL[p(x_i^{[m]}, y_i^{[m]}, z_i^{[m]})]$ 用于视觉补充， $IL[p(x_i^{[m]}, y_i^{[m]}, z_i^{[m]})]$ 是一个将坐标由东北天坐标系转换到大地坐标系下的坐标转换函数，具体过程在第二章已经阐述。

该补偿算法的运行时间相对于延迟时间是可以忽略不计的，因为所需要的前几个采样周期的传感器数据都已经被存储下来了。所以，本网络延迟补偿算法可以有效的消除视觉补充带来的网络延迟而且算法的执行不会再引入延迟。

4.4 本章小结

本章主要阐述了运行于移动机器人本体的相关定位算法。首先介绍了粒子滤波的基本原理，接着从运动模型、测量值模型、重采样及感知复位几个角度做了重点说明。接着将所有部分组合在一起列出了整个算法的伪代码。最后介绍了网

络延迟补偿算法。

第 5 章 实验设计与结果分析

本文前面章节详细介绍了整个基于云端的移动机器人户外大场景定位系统的设计与相关算法，接下来将进行综合实验对系统的性能进行验证和评估。本文选择了三个实验场景，分别有针对性地重点对系统的各个部分以及整体性能进行了验证和测评。

5.1 实验设计

本课题选择了三条路径进行实验来验证本文提出方法的性能。第一段为深圳大学城哈工大园区校园路，第二段为哈工大-大学城图书馆-清华的路线，第三段为从哈工大出发途经大学城清华及北大校区，经过留仙大道丽水路最后到北部郊区的麒麟山庄。

5.1.1 实验平台



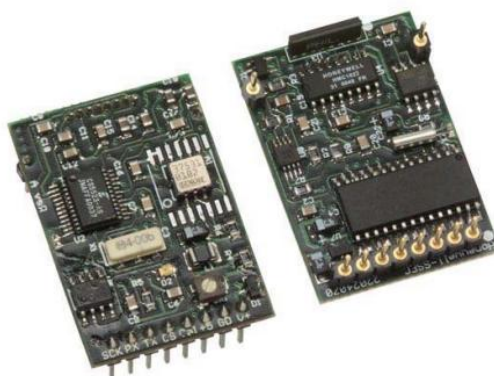
图 5-1 Summit XL 型移动机器人

实验选择 Summit XL 移动机器人作为移动平台，如图 5-1 所示。云端服务运行在由深圳国家超级计算机中心提供的服务器上（Intel Xeon CPU X5650 2.67GHz, 4GB RAM）。网络选择中国电信 4G 网络，理论上行速度（上传至网络）峰值 50Mbps，理论下行速度（从网络下载）峰值 100Mbps。惯性传感器采用 ADI 公司生产的高精度三轴惯性传感器 ADIS16405，磁力计采用霍尼韦尔公司生产的 HMR3300 数字磁力计。通过 ST 公司的 32 位嵌入式处理器 STM32F207 对二者的

数据进行融合，构成 IMU 单元，如图 5-2 所示。IMU 模块对于 roll 方向和 pitch 方向测量误差为 0.5 度，yaw 方向为 1 度。安装在移动机器人顶部的摄像头选择常规的网络摄像头，采集图像的尺寸为 640x480，采集频率为 30fps。一个常规的 GPS 模块被安装在了移动机器人上用来获取机器人的初始位置（定位过程是无需 GPS 的，如果存在其他方式获取机器人的初始位置，则无需安装 GPS 模块）。采用安装在机器人轮子上的光电编码器来采集移动机器人的速度信息，编码器采集的脉冲信号传输给 DSP 进行处理，如图 5-3 所示。所有传感器的采样周期设定为 0.1 秒。机器人的行进速度不超过 3 米每秒，实验场景选择在深圳大学城附近。



a) ADIS16405 模块



b) HMR3300 模块



c) IMU 模块

图 5-2 IMU 模块及其主要组成部分

5.1.2 实验方案及场景选择

本课题选择了三条路径进行实验来验证本文提出方法的性能。第一段为深圳大学城哈工大园区校园路，第二段为哈工大-大学城图书馆-清华的路线，第三段为

从哈工大出发途经大学城清华及北大校区，经过留仙大道丽水路最后到北部郊区的麒麟山庄。实验中选择了两种方法用于和本文提出的方法进行对比，一种方法是运用 GPS 信号定位，另一种是传统的不利用云端的粒子滤波融合惯性传感器的组合定位方式（下文中简称为“粒子滤波融合 GPS 和 INS 方法”）。下面分别对三个实验具体设计和目的进行说明。

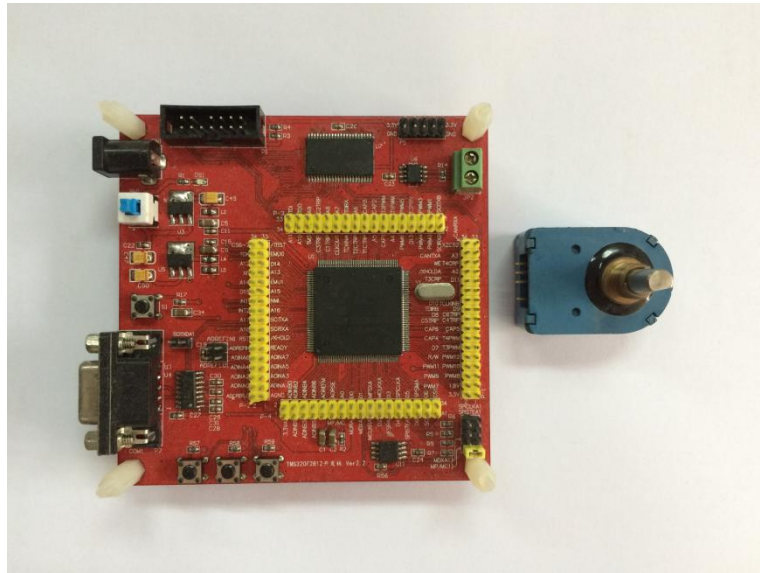


图 5-3 基于 DSP 的测速模块

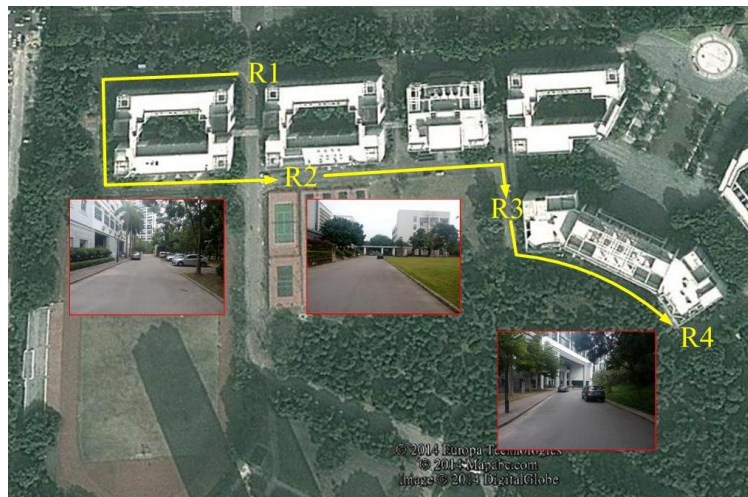


图 5-4 实验场景一路线

实验一：该场景包含区域大小为 400 米×200 米，如图 5-4 所示（ $R1 \rightarrow R2 \rightarrow R3 \rightarrow R4$ ）。首先在该场景测试整体的系统运作情况。运用 GPS 确定初始位置的误差有时可能会达到 10 米以上，所以一系列的实验被设计用于测试不同初始误差（从 5 米到 20 米）对系统运行的影响。在该场景的实验中，每隔 20 米做一个标记，接着在离线过程提取的道路网信息中找到每个标记所对应的道路点作为参考点。此

外，由于户外环境当中网络中断也是可能会发生的现象，在实验中还对网络连接中断对系统的影响也进行了测试。



图 5-5 实验场景二路线

实验二：该场景包含区域大小为 700 米×500 米，如图 5-5 所示。机器人的路线从 P1 点到 P8 点，全长 1.8 千米，包括如城市峡谷等各种不同的路段。实验二用于检验本文提出的方法在不同种类道路的定位效果，尤其是对于城市峡谷以及周围存在树木和建筑物遮挡，GPS 信号质量被严重影响的条件下。同时视觉补充过程的图像匹配部分也将在该实验中进行评估。

实验三：该场景包含区域大小为 2500 米×6700 米，全称超过 13 千米，包含绝大部分的道路种类如图 5-6 所示。该实验作为一个综合实验，主要是为了检验本文提出的方法可以应用于户外大范围的环境探索中。

5.2 实验结果分析

5.2.1 实验一结果分析

首先在实验开始之前从谷歌地球或者其他开源的地理信息软件中提取实验区域的道路网信息。通过和云端原有的道路网信息进行比较，如图 5-7 所示，道路#1 和道路#2 属于之前未保存的道路信息，所以根据最新的道路网信息更新云端服务器原有的道路网信息数据库。

在实验进行时，首先将由 GPS 提供的初始位置发送至云端服务器，这里需要说明的是，定位过程是不需要 GPS 信号的，只有在确定初始位置的时候需要 GPS

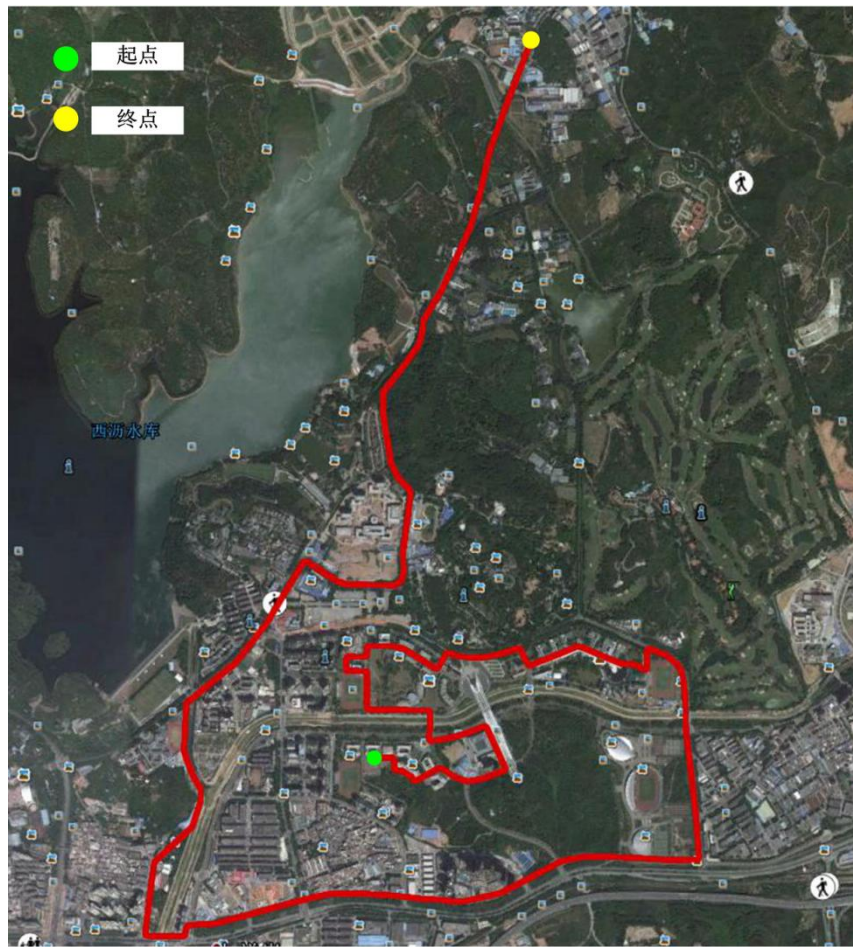


图 5-6 实验场景三路线

信号，确定初始位置的方法并不是单一的，本文选择采用 GPS。以 $E1$ 为原点， δ 为半径的局部道路网信息将被检索，如图 5-7b)所示。由于 GPS 信号的误差，所得出的初始位置可能存在较大偏差，并不在道路网上。在这种情况下，道路网上距离 $E1$ 最近的点将作为 RTI 模型的初始位置。以该位置为初始位置的 RTI 模型将在

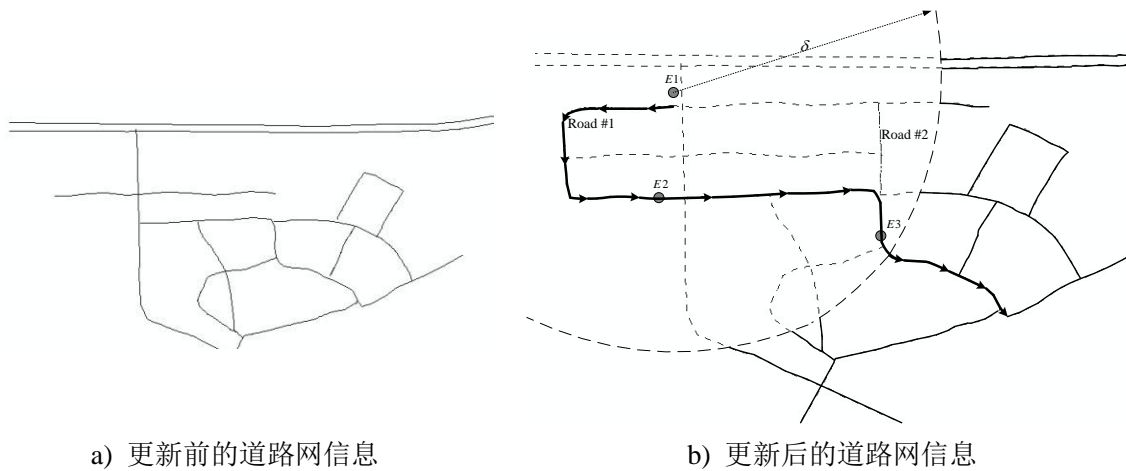


图 5-7. 更新实验区域的道路网信息（带箭头的道路表示实验路线）

云端被提取并计算出来传送给移动机器人。移动机器人运用第四章阐述的算法（表 4-2）进行定位。当机器人行进一段特定距离（200 米）之后，机器人将自身估计的位置发送至云端服务器（图 5-7b 中 E2 以及 E3），云端服务器将返回该位置周围的局部 RTI 模型，如此往复。本场景一共设置了 25 个位置参考点，参考点之间的间隔为 20 米。

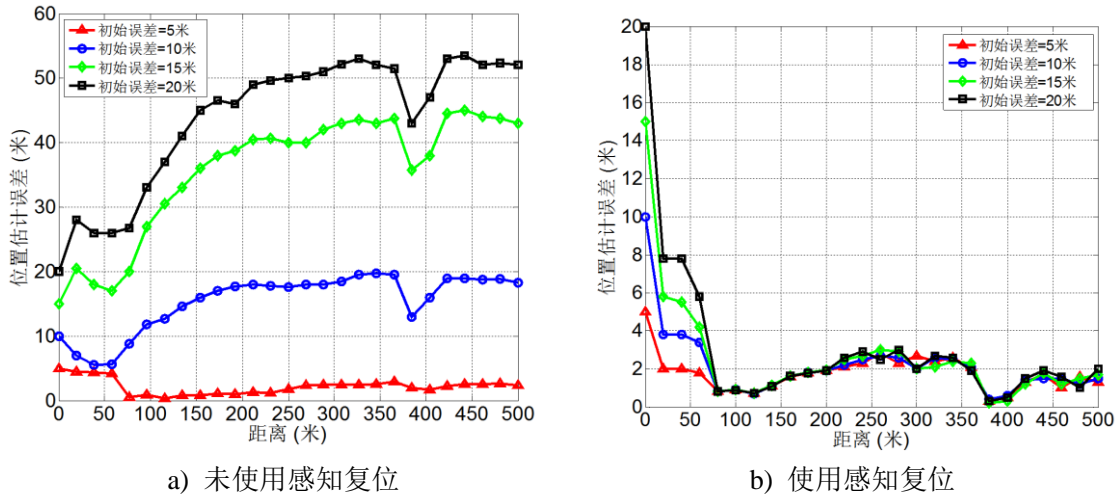


图 5-8 不同初始误差条件下的位置估计误差

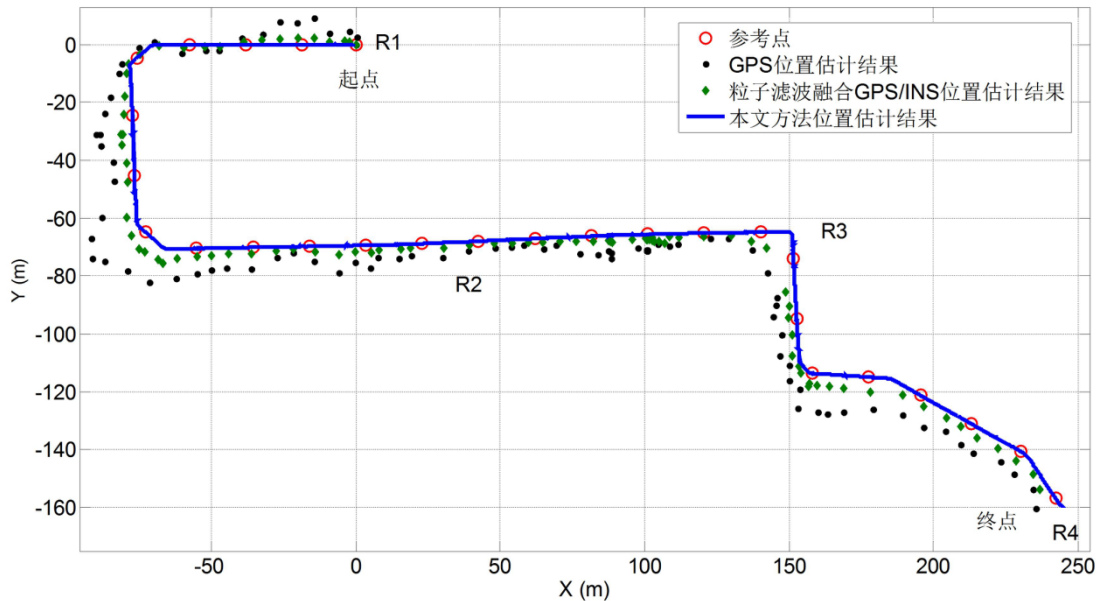


图 5-9 本文定位方法与其他定位方法的比较（实验场景一）

图 5-8 展示了由不同精度的 GPS 获得的不同初始误差条件下的位置估计误差。位置估计误差 Δd 定义为估计位置和参考位置之间的三维欧式距离。图 5-8a) 和 b) 分别表示了未采用以及采用感知复位系统位置估计误差和移动机器人行进距离的关系，红色的三角符号表示初始位置误差为 5 米时位置估计误差和移动机器人行进距离的关系，蓝色圆形符号表示初始位置误差为 10 米的情况，绿色菱形符号表

示初始位置误差为 15 米的情况，黑色正方形符号表示初始位置误差为 20 米的情况。从图 5-8a) 中信息可以看出在未采用感知复位时，系统的位置估计误差只有在初始误差为 5 米时是低于 5 米的，是可以接受的，其他情况位置估计误差都在 10 米以上。图 5-8b) 中的信息表明采用感知复位将会显著的减少由初始位置所带来的位置估计误差，所以验证了本文算法具有针对大初始误差的鲁棒性。

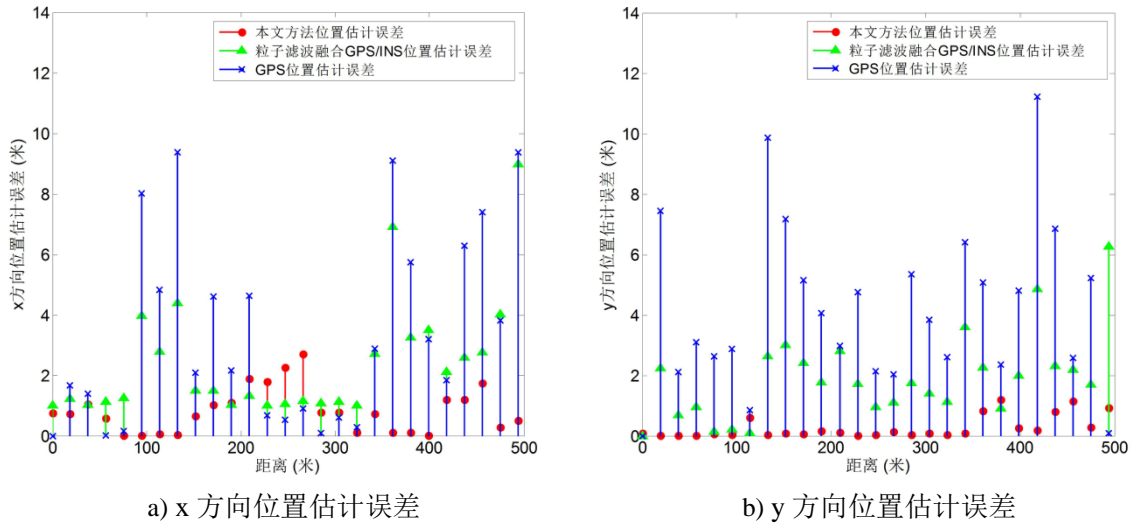


图 5-10. 本文定位方法与其他定位方法在东北天坐标系下位置估计误差的比较 (实验场景一)

图 5-9 展示了本文算法位置估计结果和 GPS 位置估计结果的比较，红色的圆圈表示参考点，黑色的圆点表示采用 GPS 定位的结果，绿色的菱形表示粒子滤波融合 GPS 和 INS 位置估计结果，蓝色的线表示本文方法的位置估计结果。可以明显的看出，本文算法的位置估计结果更靠近参考点，定位精度更高。路段 $R1 \rightarrow R2$ 以及路段 $R3 \rightarrow R4$ 由于被建筑包围，GPS 信号的接收很不稳定，采用 GPS 定位精度较差。图 5-10 展示了在东北天坐标系下本文定位方法与其他定位方法在东北天坐标系下位置估计误差的比较，红色的圆点表示本文定位方法的位置估计误差，绿色的三角形表示采用粒子滤波融合 GPS 和 INS 定位的位置估计误差，蓝色的标记表示采用 GPS 定位的位置估计误差。从图 5-10 中可以看出，由于周围建筑物和树木对 GPS 信号的遮挡和反射影响，在移动机器人行进 360 米处时采用 GPS 定位和粒子滤波融合 GPS 和 INS 定位的位置估计误差分别达到了 (9 米, 6 米) 和 (6 米, 2.3 米)，这已经严重偏离了道路，而本文定位方法的位置估计误差为 (0.2 米, 0.3 米)，和真实位置更为接近。

由于室外环境探索中可能发生网络连接中断的情况，所以网络连接中断对系统的影响也在实验一中进行了评估。在移动机器人每次和云端服务器进行通讯之后，移动机器人端会存储以其位置为圆心，方圆 200 米的 RTI 模型信息用于辅助定位。如果移动机器人在丢失网络的过程中未驶出该区域，则网络丢失不会对定

位产生任何影响，如果移动机器人行进至该范围之外，由于网络丢失，无法更新局部 RTI 模型，移动机器人端并不包含局部 RTI 模型区域以外的地图信息。所以移动机器人此时只能依靠机载的 IMU 以及编码器采用航迹推算法进行定位。由于移动机器人行驶过程的机械震动以及传感器的累计误差，定位误差将会随着行驶距离增加而显著增大，如图 5-11 所示，不同颜色和符号表示的曲线为不同组实验数据，图中共列举了五组实验结果。当机器人和云端重新连接之后，机器人会将当前时刻的位置估计发送至云端服务器，之前航迹推测导致的位置估计误差可以等效于前面阐述过的初始误差，由于采用了感知复位，该误差会被迅速补偿，保证系统之后的正常运行。

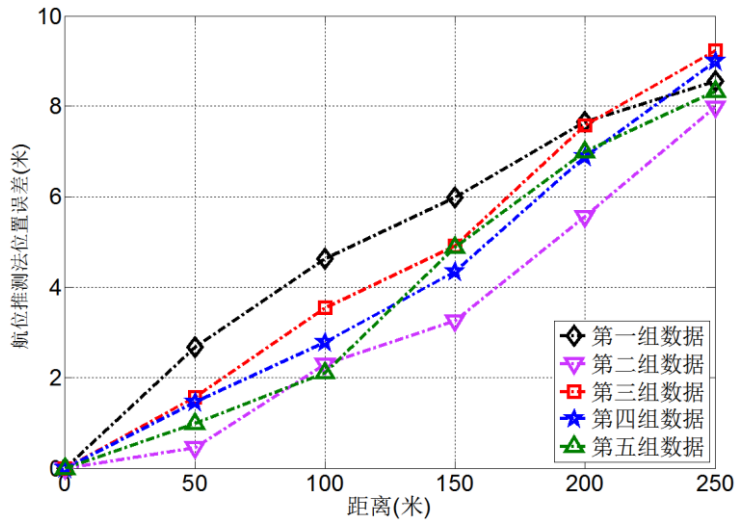


图 5-11 在网络丢失时采用航迹推测法产生的位置估计误差与行进距离的关系

5.2.2 实验二结果分析

实验场景二选择的路径为从深圳大学成哈工大园区出发途径图书馆，北大汇丰商学院，清华校区然后回到哈工大园区，全长 1.8 千米，包括如城市峡谷等各种不同的路段。实验二主要用于检验本文方法在不同种类道路的定位效果，尤其是对于城市峡谷以及周围存在树木和建筑物遮挡，GPS 信号质量被严重影响的条件下。同时视觉补充过程的图像匹配部分也将在该实验中进行评估。

本文定位方法与其他定位方法的位置估计结果比较如图 5-12 所示，其中红色的线表示本文定位方法的位置估计结果，黄色的点表示 GPS 位置估计结果，黄色的线表示粒子滤波融合 GPS 和 INS 位置估计结果。移动机器人途径多种不同 GPS 信号质量较差的路段。从图中可以明显的看出，在次干道（图中 C 点到 D 点，长度约 100 米）和城市峡谷（图中 G 点到 H 点，长度约 230 米）这两个路段，采用 GPS 和粒子滤波融合 GPS 和 INS 定位方法定位效果较差，位置估计结果严重偏离

了实际行走的道路。采用 GPS 定位在这两个路段的定位误差最大时超过了 30 米，但是同样情况下本文定位算法的误差仅为 2 米。本实验选择的路线起始点和终点为同一位置。本文定位算法的终点位置估计结果距离实际位置 0.5 米，而采用 GPS 和粒子滤波融合 GPS 和 INS 定位方法的终点位置估计结果距离实际位置分别为 8 米和 4 米。

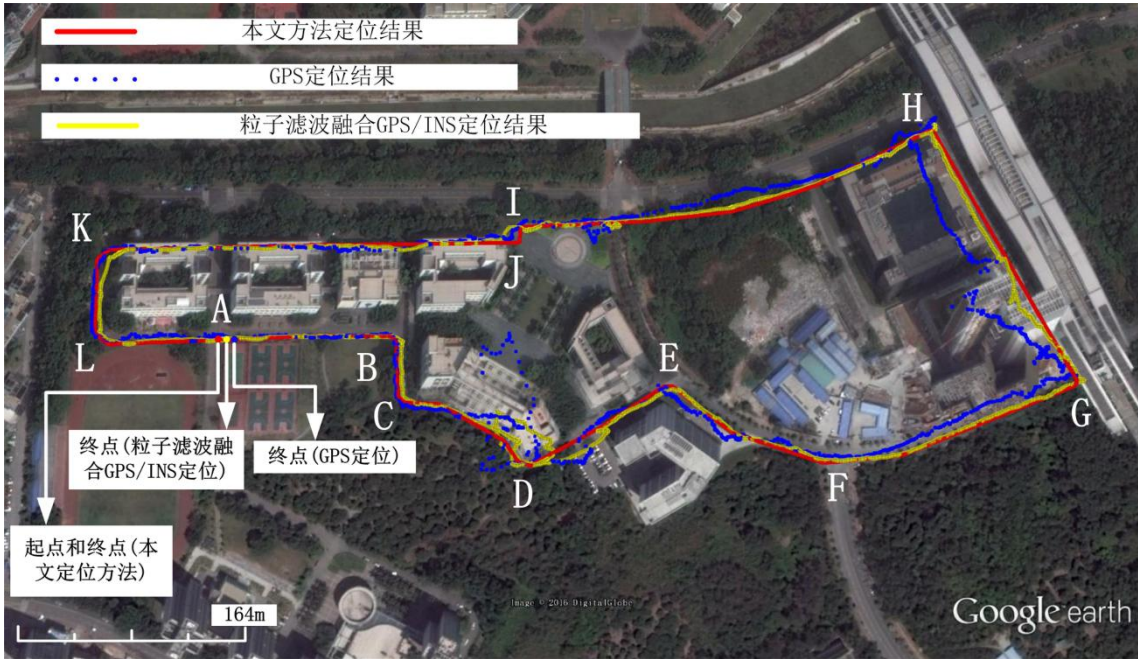


图 5-12. 本文定位方法与其他定位方法的比较（实验场景二）

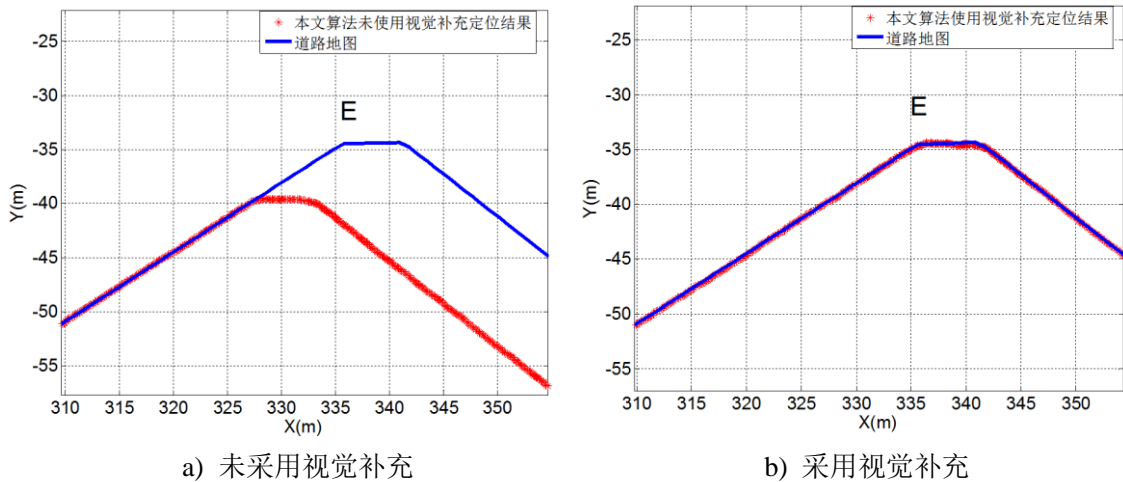


图 5-13 图 5-12 中 E 处附近位置估计结果

当移动机器人匀速行进在一段相对较长、平坦且道路地形特征变化较少的路面上时，由于机器人姿态变化较小，粒子滤波由于在累计误差的影响下产生不稳定的定位效果甚至导致定位失败。这可能导致位置估计结果落后于移动机器人的实际位置。然而，视觉补充的引入可以有效地解决这个问题，如图 5-13 所示。图

中蓝色线表示道路网 a)图中红色标记表示未使用视觉补充的位置估计结果,b)图中红色标记表示采用了视觉补充的位置估计结果。可以明显的看出,加入视觉补充能有效的解决由于移动机器人姿态变化较小以及传感器累计误差导致的位置估计结果落后于真实位置的问题。

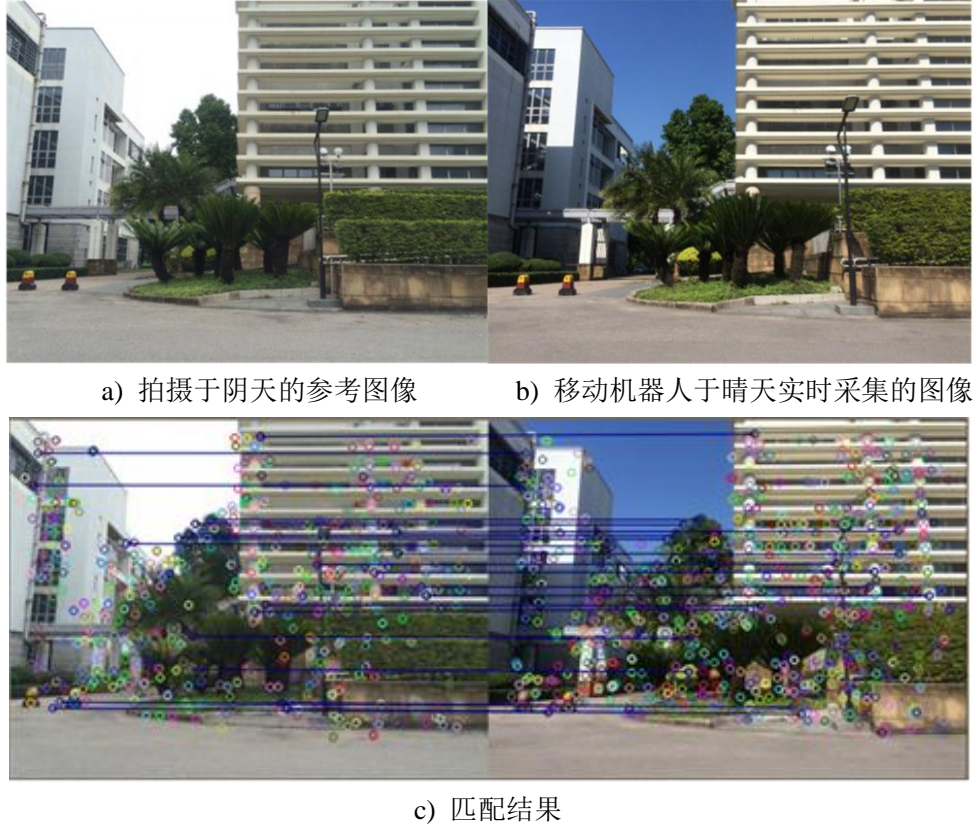


图 5-14 在不同的光照条件下的图像匹配结果

视觉补充部分的图像匹配过程也在实验二中进行了测评。对于室外环境来讲,光照和阴影情况会随着天气的不同而产生变化。比如图 5-14a)中的参考图像拍摄于阴天,而图 5-14b)是移动机器人在晴天时实时采集的图像,从中可以明显的看出光照和阴影条件的不同。5-14c)是匹配结果,其中图中的小圆圈表示 SURF 算法检测出的特征点,连线表示两者之间匹配成功的特征点。从匹配结果可以明显的看出,可靠的匹配主要产生于图像中的建筑物部分, SURF 算法属于基于梯度变化的特征描述算法,而建筑物部分由于天气变化引起的主体梯度变化较小,所以不会影响匹配结果。但是当存在特殊颜色的光以某种角度直射到建筑物上时可能会引起建筑物主体部分在图像中的梯度变化较大,此时,基于 SURF 的图像匹配算法可能会受到影响。

图像中存在部分场景变化的情况也在实验二中进行了测评,见图 5-15。图 5-15a)中是参考图像,图 5-15b)是移动机器人实时采集的图像,可以看出,参考图像中的

舞台、地面上的标志以及建筑物上的横幅都不存在了。两幅图像中存在明显的场景变化，图 5-15c)表示匹配结果，其中图中的小圆圈表示 SURF 算法检测出的特征点，连线表示两者之间匹配成功的特征点。因为稳定的匹配成功的特征点主要位于图中的建筑物本体部分，而建筑物主体并未发生明显的改变，所以并没有影响匹配结果。然而，如果在一些特殊的场合，一些物体（比如停靠在机器人侧面的车辆或者过往的行人）完全遮挡住了移动机器人机载摄像头的视野，遮挡住了图像中产生主要稳定匹配特征的部分，将会导致匹配的失败。如果当前的匹配失败了，则当前的匹配将被丢弃，云端服务器不会向移动机器人发送参考图像对应的位置信息。少量的匹配失败不会对定位结果产生任何负面影响。

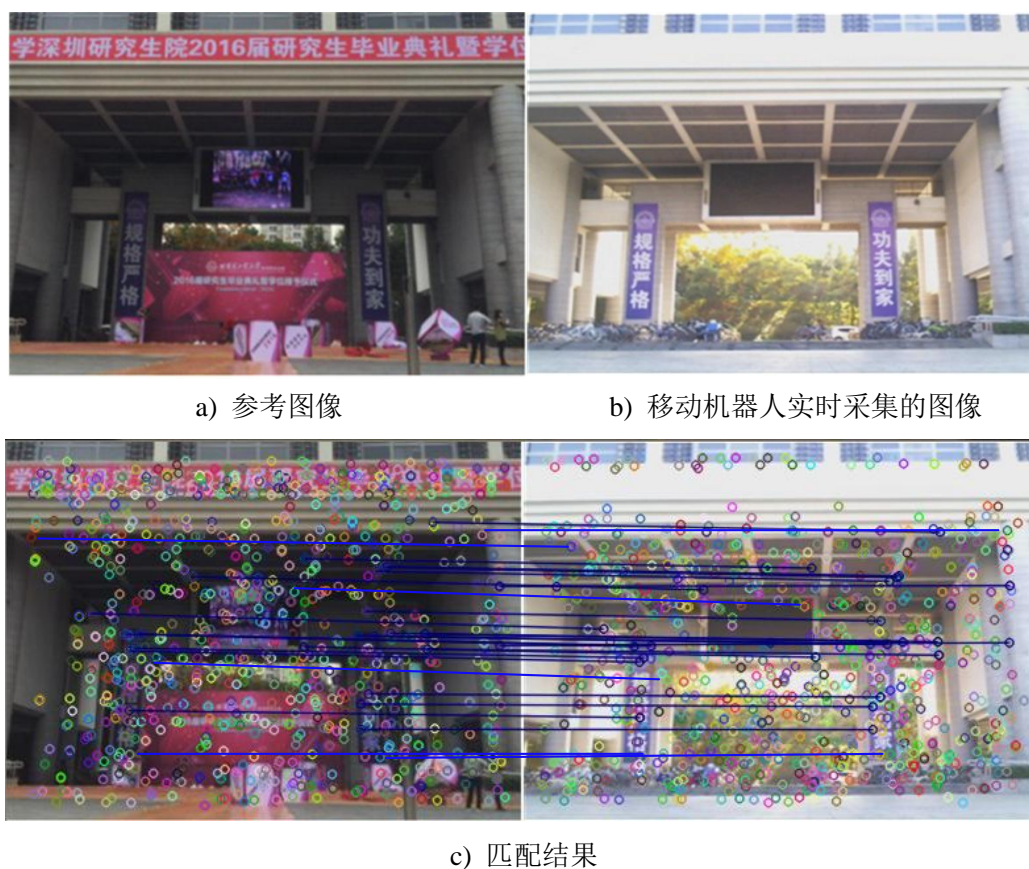


图 5-15 图像中存在部分场景变化时的匹配结果

关于安装在移动机器人上的摄像头的位置、高度以及朝向在本文方法中并没有绝对严格的要求。只要将摄像头朝向移动机器人侧面并与水平方向呈 10 到 20 度即可，原则上只要保证能够将侧面物体主要的算法可识别特征拍摄到即可。对于摄像头安装的不同位置、高度以及俯仰角对于图像匹配的影响也在实验二中进行了测试。图 5-16 中展示了摄像头处于不同高度和俯仰角时采集的图像，图 5-16a)为参考图像，图 5-16b)和 c)分别表示摄像头安放在不同俯仰角且距离地面 0.5 米时

拍摄的图像，图 5-16d)和 e)分别表示摄像头处于不同俯仰角且距离地面 1.2 米时拍摄的图像。实验结果表明，由于这四幅图像中建筑物的主体特征没有发生明显变化，所以这些图像都能够与参考图像匹配成功。



a) 参考图像



b)



c)



d)



e)

图 5-16 摄像头处于不同高度和俯仰角时采集的图像，a)为参考图像，b)-e)为摄像头处于不同高度和俯仰角时采集的图像

尽管由于本文采取了非常严格的匹配机制，大部分的错误匹配是可以避免的，但是仍然存在部分特殊情况会导致错误匹配，比如在两个不同的位置存在两个几乎完全相同的建筑物或其他物体。在出现错误匹配的情况时，本文方法体系下会有相应的策略来解决错误匹配的问题。具体方法如下：第一点，只有位于局部本地 RTI 模型范围内参考图像数据库中的参考图像会被检索并试图与移动机器人实

际采集的图像进行匹配，这样就避免了与距离机器人实际位置较远的错误位置的参考图像匹配成功。第二点，如果匹配成功的参考图像所对应的位置信息确实位于局部本地 RTI 模型范围之内，移动机器人存贮的 IMU 测量值和编码器的测量值将用于向前推测之前机器人采集实时图像时的位置，如果这个位置信息和参考图像的位置信息相距过大，则说明匹配结果错误，视觉补充的辅助位置估计结果将不会反馈给移动机器人。同时，图像匹配进程会重新启动并不会对整个定位过程产生不良影响。

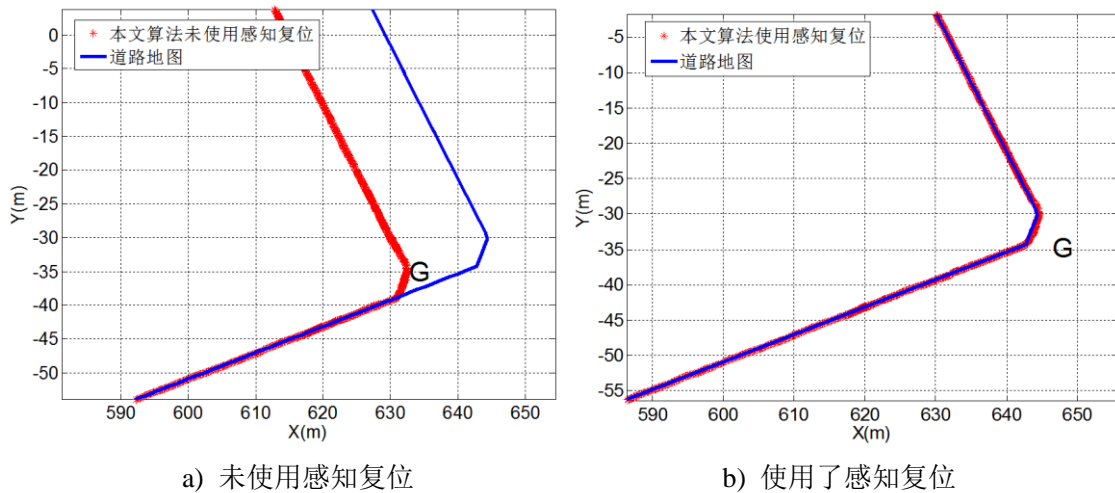


图 5-17 在具有大量相似特征的路段的位置估计结果

另一方面，在类似的条件下如果机器人行进在路边充满相似特征从而不适合做图像匹配的特征的路段时，视觉补充部分将无法完成。比如道路两旁栽满了树木或其他植被，由于可识别的特征太少将导致视觉补充会失败。在这种情况下，感知复位将会解决这个问题。图 5-17 展示了实验场景二中图 5-12 中 G 点附近使用以及未使用感知复位的位置估计结果，其中蓝色线表示道路网信息，图 5-17a) 图中红色标记表示未采用感知复位的位置估计结果，图 5-17b) 图中红色标记表示采用了感知复位的位置估计结果。F 到 G 路段路旁栽满了茂密的植被，由于植物的可用于匹配的特征较少，不利于图像匹配，从而使得视觉补充部分失效。从图 5-17a) 中可以看出，移动机器人在行进至拐点 G 处附近时产生了较大的定位误差，图 5-17b) 展示了采用感知复位后解决了在相同位置定位误差较大的问题。

5.2.3 实验三结果分析

实验三：该场景包含区域大小为 2500 米×6700 米，全程超过 13 千米，包含绝大部分的道路种类如图 5-18 所示。整个实验持续 3 小时 10 分钟，移动机器人的平均速度在 1.21 米每秒，整个实验过程中系统运行稳定，没有系统崩溃的情况发生。该实验作为一个综合实验，主要是为了检验本文提出的方法可以应用于户外大范

围的环境探索当中。

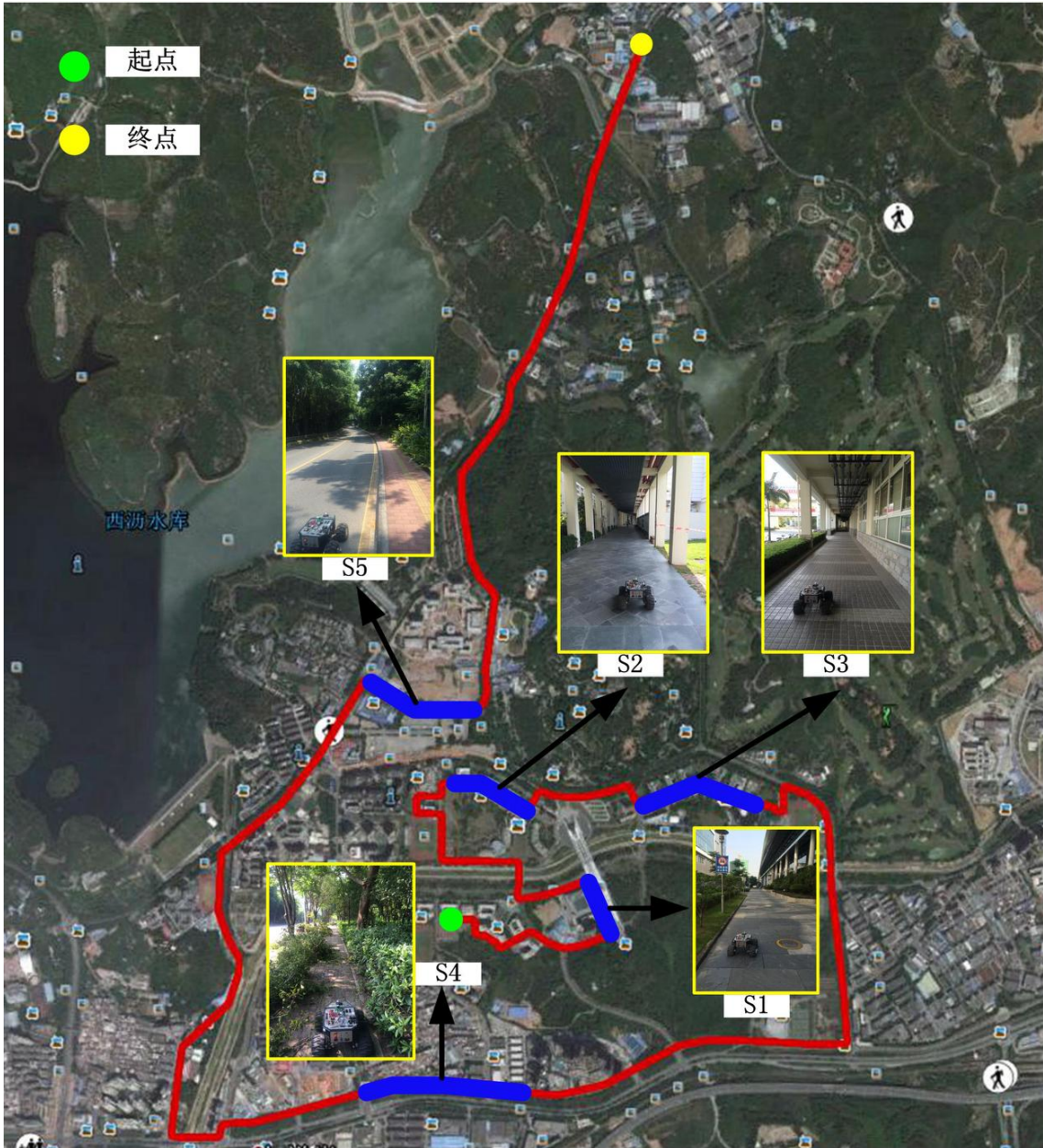


图 5-18 本文方法在实验场景三中的位置估计结果

图 5-18 展示了本文定位方法在实验场景三当中的位置估计结果，图中红色线代表本文定位方法的位置估计结果，蓝色线部分为五段(S1-S5)具有代表性的 GPS 信号被完全屏蔽或质量较差严重影响传统方法定位的路段，传统的基于 GPS 的定位方法及组合定位方法在这些路段的位置估计结果是比较差的，这样的路段对于户外定位而言也一直是具有挑战性的，绿色的圆点代表实验中移动机器人的出发点，黄色的圆点为终点。图中 S1 路段属于一段城市峡谷类型的支路，该路段长约 230 米，道路两旁分别是大学城图书馆以及其他教学楼，由于这些建筑物对 GPS

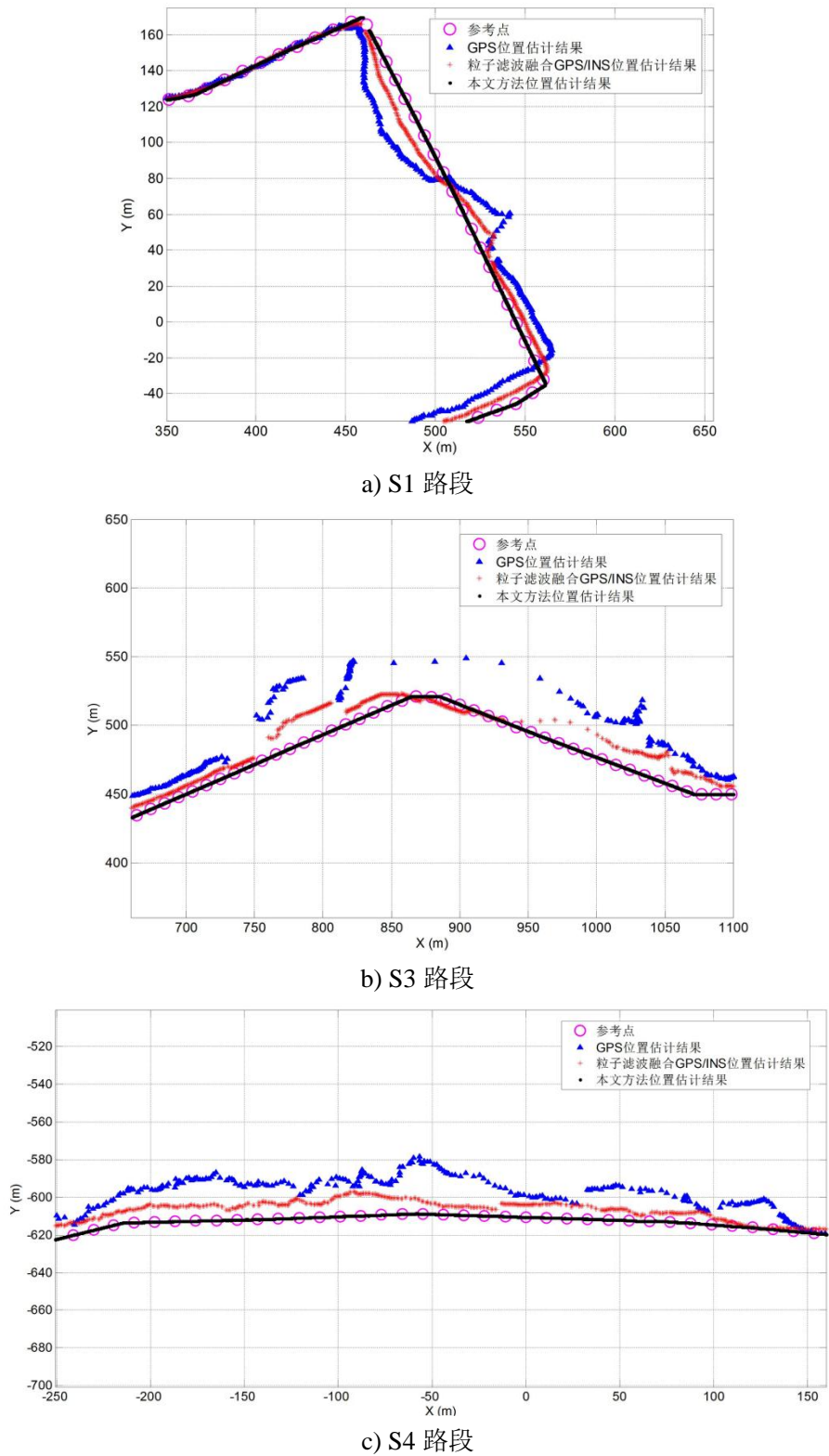
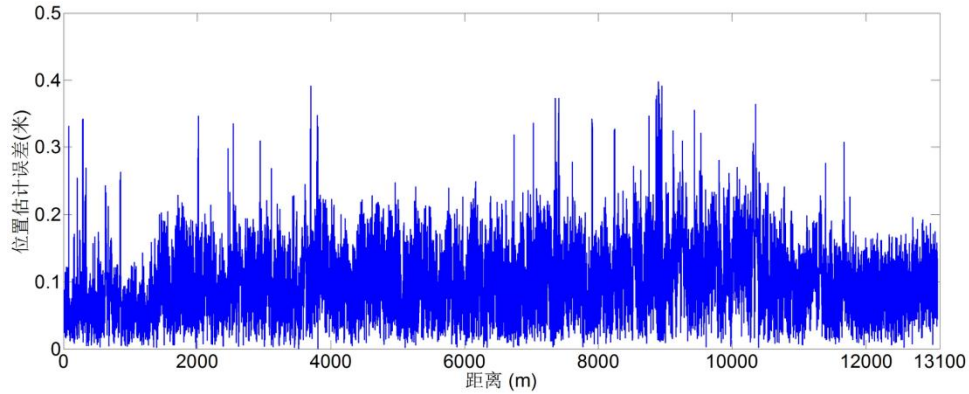
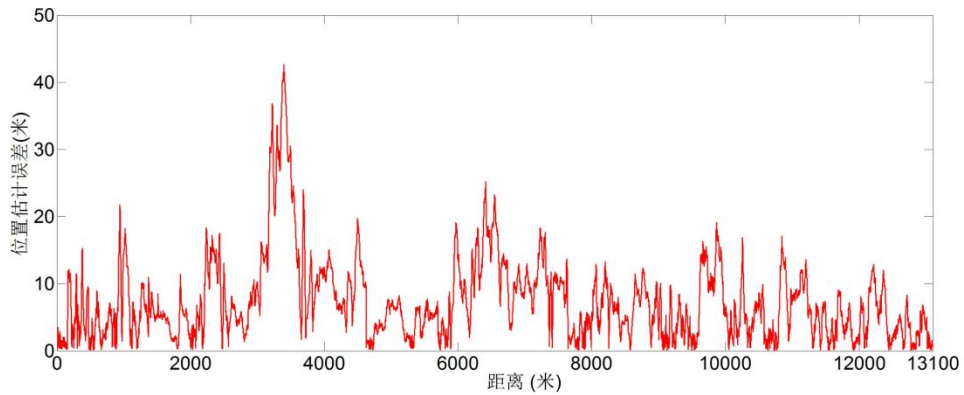


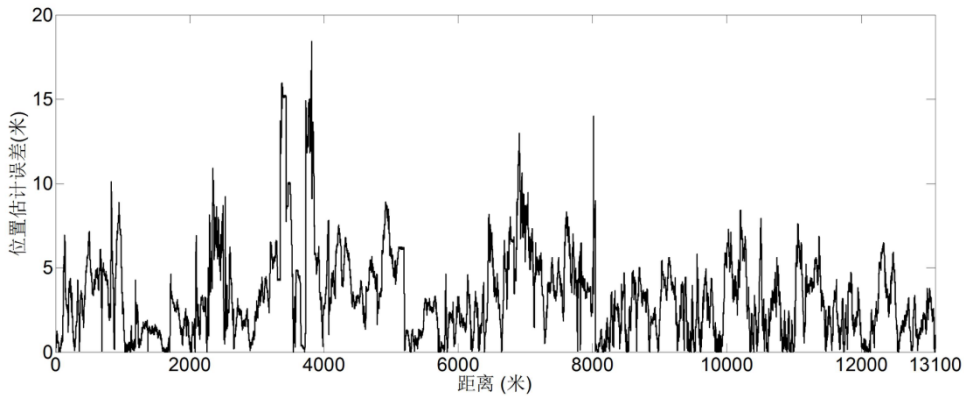
图 5-19 三个典型路段的不同定位方法位置估计的比较



a) 本文定位方法



b) GPS 定位方法



c) 粒子滤波融合 GPS 和 INS 定位方法

图 5-20 实验三中不同定位方法位置估计误差

信号的遮挡以及反射效应，GPS 信号常常会偏离于道路很多。图中另外两个路段 S2 和 S3 都属于隧道类型，长度分别为 250 米和 450 米，GPS 信号在这样条件的路段上是被完全屏蔽的。图中的 S4 和 S5 路段属于郊区的次干道，道路两旁栽满了茂密的树木从而部分遮挡了 GPS 信号。

图 5-19 分别表示了在三种不同路段 S1,S3,S4 的不同定位方法的比较，图中粉色的圆圈表示作为参考的道路网信息，这些道路网信息是在第二章介绍的离线过

程中从开源的地理信息软件当中提取出来的。蓝色的三角型符号表示 GPS 位置估计结果，红色的圆点表示粒子滤波融合 GPS 和 INS 位置估计结果，黑色的圆点表示本文方法位置估计结果。整个实验过程三种定位方法的位置估计误差如图 5-20 所示，位置估计误差定义为位置估计误差和道路网信息之间的欧式距离，图中蓝色曲线表示本文方法的位置估计误差，红色曲线表示 GPS 定位位置估计误差，黑色曲线表示粒子滤波融合 GPS 和 INS 定位位置估计误差。从图中能够明显地看出本文提出的位置估计误差远小于其他两种方法，而且由于本文定位方法不像另两种方法依赖于 GPS，即使在 GPS 信号被屏蔽或严重干扰的区域也能保证精确的定位结果。

值得注意的一个问题是，由于本文定位过程采用的东北天坐标系是以地球表面一个点的切面作为坐标系的 XOY 平面，由于地球的尺寸相对较大，在方圆 10 千米范围内的地球表面是可以近似视为平面的，但是当移动机器人朝着一个方向前进超过 10 千米时，由于球面所引起的误差就需要考虑进来了。本文定位方法的应对办法如下：当机器人行进超过 10 千米时，会以当前移动机器人的位置为原点，按照第二章中描述的方法重新构建一个新的东北天坐标系，并将相关的道路网信息和 RTI 模型转换到新的东北天坐标系下，这样就解决了由于将球面近似为平面所带来的定位误差。

整个系统运行的实时性也在本实验中进行了测试和验证。实验中对比了本文基于云端的移动机器人架构和定位方法相同但是不采用云端的架构，以此突出采用云端架构的优点和必要性。实验三场景的尺寸为 2500 米 × 6700 米，该区域的道路网信息数据量为 1GB，包含 1.087 百万条数据记录，覆盖该区域的全部参考图像的数据信息约为 4GB。这些数据对于存储在低成本的嵌入式处理上是相对比较困难的。整个系统的计算负荷主要可以分为两个部分：第一部分是通过道路网信息计算 RTI 模型，第二部分是视觉补充。计算 RTI 模型又分为从大量的道路网信息检索得到局部道路网信息和通过局部道路网信息计算 RTI 模型两个部分。对于本文所使用的搭载 MySQL 的云端服务器而言，检索局部道路网信息耗时 1 毫秒，然而对于智能手机级别的低成本嵌入式系统（搭载 SQLite），需要 10 秒左右才能完成同样的局部数据检索工作。云端服务器在计算 RTI 模型过程耗时少于 20 毫秒，同样的工作如果在机载嵌入式处理器中完成的话需要 3.5 秒。类似的，视觉补充部分的耗时采用云端架构只需要少于 450 毫秒的时间，即使包含移动机器人和云端服务器之间的数据传输时间也不会超过 1.5 秒。但是对于智能手机级别的嵌入式处理器相同的工作需要耗时约 6 秒。显然，在没有云端服务器的帮助下，单独依靠机器人本身的嵌入式处理器是很难将本文的方法实时运行的。另一方面，随着移

动机器人运行环境范围的扩大，比如包含整个城市区域，对于硬件和处理时间的要求会增加得更加显著。而由于云端服务器的存在，这些问题都得到了有效地解决。

5.3 本章小结

本章主要对本课题的主要研究内容进行了综合实验设计。整个实验过程总共包含三个户外场景，场景尺寸逐渐递增，每个实验分别重点验证和测试了系统的不同功能模块，并对实验结果进行了详细的分析和讨论，同时和其他方法也做了详细的对比。

结 论

本课题主要提出了一种基于云端的移动机器人户外大场景定位方法。同时，对于机器人和云端通信所产生的时间延迟而导致的定位误差，本文也采用了有效的误差补偿算法对其进行了补偿。

本课题主要的贡献在于以下三个方面：

（1）本文提出的方法能够使得移动机器人在不同的复杂户外大场景中进行实时定位，并且在定位过程中不依赖于 GPS 信号。

（2）本文提出的方法允许移动机器人将大量的道路信息存储在云端的服务器中，从而大大降低了对移动机器人平台自身嵌入式系统的硬件要求。同时，对于移动机器人和云端服务器交互数据产生的网络延迟也进行了补偿，消除其在定位过程中产生的不良影响。

（3）当机器人行进在相对较长、角度变化较小且平坦的道路上时，由于姿态变化较小以及传感器的累计误差，将会导致定位结果不准确，产生模糊的定位结果。本文提出的视觉补充和感知复位方法能够有效地解决这一问题。

然而由于研究时间有限且相关科研经验的欠缺，本课题仍然有许多可以改进的方面：

（1）部分道路网信息的提取仍然需要手动操作，这个过程难免会引入少量误差。未来可以通过更智能且自动化的方法获得更为精准的道路网信息，从而提高定位精度。

（2）虽然本文实验包含大部分道路情况并且场景规模较大，但仍然存在对于定位而言更具挑战性和复杂的环境，未来应当进行更多的实验以检验和提升系统性能。

参考文献

- [1] Quintas J M, Menezes P J, Dias J M. Cloud Robotics: Toward Context Aware Robotic Networks[C]// Iasted International Conference Robotics, 2011:87-101.
- [2] Hu G, Tay W P, Wen Y. Cloud robotics: architecture, challenges and applications[J]. IEEE Network, 2012,26(3):21-28.
- [3] Kamei K, Nishio S, Hagita N, et al. Cloud networked robotics[J]. Network IEEE, 2012,26(3):28-34.
- [4] 徐浩. GPS车辆导航与定位系统的地图匹配算法研究[D]. 中国科学技术大学, 2007:
- [5] Parent M. Automated Vehicles: Autonomous or Connected[C]//IEEE International Conference on Mobile Data Management, 2013:2.
- [6] Bonnifait P, Bouron P, Crubille P, et al. Data fusion of four ABS sensors and GPS for an enhanced localization of car-like vehicles[C]//IEEE International Conference on Robotics & Automation, 2001:1597-1602.
- [7] Bouvet D, Garcia G. Civil-engineering articulated vehicle localization: solutions to deal with GPS masking phases[C]//2000:3499-3504.
- [8] 罗雨. 基于多信息融合的轮式移动机器人定位导航技术研究[D]. 北京理工大学, 2011:
- [9] Semeniuk L, Noureldin A. Bridging GPS outages using neural network estimates of INS position and velocity errors[J]. Measurement Science & Technology, 2006,17(10):2783-2798.
- [10] Drawil N M, Amar H M, Basir O A. GPS Localization Accuracy Classification: A Context-Based Approach[J]. Intelligent Transportation Systems IEEE Transactions on, 2013,14(1):262-273.
- [11] Brenneke C, Wulf O, Wagner B. Using 3D laser range data for SLAM in outdoor environments[C]//Ieee/rsj International Conference on Intelligent Robots and Systems, 2003:188-193.
- [12] Nuchter A, Surmann H, Lingemann K, et al. 6D SLAM with an application in autonomous mine mapping[J]. 2004,2(2):1998-2003.
- [13] Martin M C, Moravec H P. Robot Evidence Grids[J]. Robot Evidence Grids,

- 1996.
- [14] Fairfield N, Kantor G, Wettergreen D. Real - Time SLAM with Octree Evidence Grids for Exploration in Underwater Tunnels[J]. Journal of Field Robotics, 2007,24(1-2):3-21.
 - [15] Blaer P, Allen P. Topological mobile robot localization using fast vision techniques[C]//IEEE International Conference on Robotics and Automation, 2002. Proceedings. ICRA, 2002:1031-1036.
 - [16] Kawewong A, Tongprasit N, Tangruamsub S, et al. Online and Incremental Appearance-based SLAM in Highly Dynamic Environments[J]. International Journal of Robotics Research, 2011,30(1):33-55.
 - [17] Valgren C, Lilienthal A J. SIFT, SURF and seasons: Long-term outdoor localization using local features[C]//European Conference on Mobile Robots ,2007:
 - [18] Mandel C, Laue T. Particle filter-based position estimation in road networks using digital elevation models[C]//IEEE/RSJ International Conference on Intelligent Robots & Systems, 2010:5744-5749.
 - [19] Zhu X, Qiu C, Mark A. Terrain Inclination Aided Three- Dimensional Localization and Mapping for an Outdoor Mobile Robot[J]. International Journal of Advanced Robotic Systems, 2013,10(1):257-271.
 - [20] Zhu X, Qiu C, Minor M A. Terrain-inclination-based Three-dimensional Localization for Mobile Robots in Outdoor Environments[J]. Journal of Field Robotics, 2014,31(3):477-492.
 - [21] Majdik A L, Verda D, Albers-Schoenberg Y, et al. Air-ground Matching: Appearance-based GPS-denied Urban Localization of Micro Aerial Vehicles[J]. Journal of Field Robotics, 2015,32(7):1015-1039.
 - [22] Vandapel N, Donamukkala R, Hebert M. Experimental Results in Using Aerial LADAR Data for Mobile Robot Navigation[M].Springer Berlin Heidelberg, 2006: 103-112.
 - [23] Thrun S, Montemerlo M, Dahlkamp H, et al. Stanley: The robot that won the DARPA Grand Challenge[J]. Journal of Field Robotics, 2004,23(9):1-43.
 - [24] Urmson C, Anhalt J, Bagnell D, et al. Autonomous driving in urban environments:

- Boss and the Urban Challenge.[J]. Journal of Field Robotics, 2009,25(8):425-466.
- [25] Lankenau A, Rofer T. Mobile robot self-localization in large-scale environments[C]//IEEE International Conference on Robotics and Automation, 2002. Proceedings. ICRA, 2002:1359-1364.
- [26] Cazorla M, Bonev B. Large Scale Environment Partitioning in Mobile Robotics Recognition Tasks[J]. Journal of Physical Agents, 2010,4(2):11-18.
- [27] Bradley D M, Patel R, Vandapel N, et al. Real-time image-based topological localization in large outdoor environments[C]//Ieee/rsj International Conference on Intelligent Robots and Systems, 2005:3670-3677.
- [28] Valgren C, Lilienthal A J. SIFT, SURF and seasons: Long-term outdoor localization using local features[C]//European Conference on Mobile Robots, 2007:
- [29] Badino H, Huber D, Kanade T. Real-time topometric localization[C]// ieee International Conference on Robotics & Automation, 2012:1635-1642.
- [30] Goldberg K. Cloud Robotics and Automation:A Survey of Related Work[J]. Eecs Department, 2013.
- [31] Berenson D, Chestnutt J, Srinivasa S S, et al. Pose-constrained whole-body planning using Task Space Region Chains[C]//Ieee-Ras International Conference on Humanoid Robots, 2009. Humanoids, 2010:181-187.
- [32] Arumugam R, Enti V R, Bingbing L, et al. DAVinCi: A cloud computing framework for service robots[C]//IEEE International Conference on Robotics & Automation, 2010:3084-3089.
- [33] Riazuelo L, Civera J, Montiel J M M. C2TAM: A cloud framework for cooperative tracking and mapping[J]. Robotics & Autonomous Systems, 2013,62(4):401-413.
- [34] Kehoe B, Matsukawa A, Candido S, et al. Cloud-based robot grasping with the google object recognition engine[C]//IEEE International Conference on Robotics and Automation, 2013:4263-4270.
- [35] Hu G, Tay W P, Wen Y. Cloud robotics: architecture, challenges and applications[J]. IEEE Network, 2012,26(3):21-28.
- [36] Wang L, Liu M, Meng Q H. Towards cloud robotic system: A case study of

- online co-localization for fair resource competence[C]//IEEE International Conference on Robotics and Biomimetics, 2012:2132-2137.
- [37] Mohanarajah G, Usenko V, Singh M, et al. Cloud-Based Collaborative 3D Mapping in Real-Time With Low-Cost Robots[J]. IEEE Transactions on Automation Science & Engineering, 2015,12(2):1-9.
- [38] Barbosa J P D A, Lima F D P D, Coutinho L D S, et al. ROS, Android and cloud robotics: How to make a powerful low cost robot[C]//Advanced Robotics (ICAR), 2015 International Conference on IEEE, 2015:158-163.
- [39] Chen H, Xing L, Ding W. Twelve Kinds of Gridding Methods of Surfer 8.0 in Isoline Drawing[J]. Chinese Journal of Engineering Geophysics, 2007.
- [40] Bay H, Ess A, Tuytelaars T, et al. Speeded-Up Robust Features[J]. Computer Vision & Image Understanding, 2006,110(3):404-417.
- [41] Thrun S, Fox D, Burgard W, et al. Robust Monte Carlo localization for mobile robots[J]. Artificial Intelligence, 2001,128(1 - 2):99-141.
- [42] Ascher C, Kessler C, Wanknerl M, et al. Dual IMU Indoor Navigation with particle filter based map-matching on a smartphone[C]//Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference, 2010:1-5.
- [43] Mmerle R, Triebel R, Pfaff P, et al. Monte Carlo localization in outdoor terrains using multilevel surface maps[J]. Journal of Field Robotics, 2008,25(6 - 7):346-359.
- [44] Thrun S. Probabilistic robotics[J]. 2005,45(3):52-57.
- [45] Engelson S P, Mcdermott D V. Error correction in mobile robot map learning[C]//IEEE International Conference on Robotics and Automation, 1992. Proceedings, 1992:2555-2560.
- [46] Lenser S, Veloso M. Sensor resetting localization for poorly modelled mobile robots[J]. 2002,2:1225-1232.

哈尔滨工业大学学位论文原创性声明和使用权限

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《基于云端的移动机器人室外环境定位系统的设计与实现》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名： 日期： 年 月 日

学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

（1）学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；（2）学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；（3）研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名： 日期： 年 月 日

导师签名： 日期： 年 月 日

致 谢

转眼间，两年半的研究生生涯就要完结。首先需要感谢的就是我的导师朱晓蕊教授。自我踏入深研院以来，朱老师渊博的专业知识、务实严谨的治学态度、富有创造性和启发性思维方式、精益求精的工作作风不断地激励着我，在我慵懒懈怠的时候为我敲响警钟。同时生活中他也以包容的心态、平易近人的态度润物无声般地影响着我。在我完成课题的整个过程中他总是能把握大局、深入细节、切中要害、耐心指导，给我莫大的鼓励和帮助。

感谢深研院控制科学与工程专业的各位老师，尤其是专业课课程设计中给予我指导与灵感的老师们，是你们让我找到了作为一名研究生最初的乐趣。

感谢博士后师兄邓富成、邱纯鑫，在我的课题研究和生活中给予了我大量的帮助和启发。感谢同级的曾泽峰、张振操、任家敏、李玉洁、李蒙、李俭，大家一起努力，无论生活还是学习上都互相关怀。你们对于学习追求完美和对于生活的热爱的态度深深影响着我，让我受益匪浅。感谢实验室的师弟杨浩、何超、梁辰、樊建崑、尹路，感谢你们在我的实验中提供的帮助，没有你们我很难独立完成数十千米的大场景定位实验。

最后还要感谢我的家人。他们为作者的成长付出了无数的心血和艰辛的劳动。感谢你们一直以来对于作者的我条件支持和鼓励。是你们给了作者前进的动力。