

第十三届中国研究生电子设计竞赛

技术论文

论文题目：面向室内场馆的智能定位系统

参赛单位：重庆邮电大学

队伍名称：诗成笑傲凌沧海

指导老师：杨小龙、田增山

参赛队员：任浩良、金潇潇、从海峰、卢帅、陈益

完成时间：2018 年 6 月

目录

摘要	1
引言	1
第 1 章 作品介绍	1
1.1 作品内容概括	1
1.2 作品难点	2
1.2.1 实现自适应 AP 布局	2
1.2.2 高效率地构建位置指纹数据库	2
1.2.3 保持指纹库定位精度困难	2
1.3 作品创新	3
1.3.1 提出自适应 AP 布局算法	3
1.3.2 快速构建位置指纹数据库	3
1.3.3 指纹库自适应更新	3
1.3.4 室内高精度融合定位	3
第 2 章 方案论证与设计	4
2.1 系统框架设计	4
2.2 速度及航向角解算模块	4
2.2.1 速度解算	4
2.2.2 航向角解算	5
2.3 蓝牙指纹定位模块	7
2.3.1 自适应 AP 部署	8
2.3.2 快速构建指纹数据库	9
2.4 融合定位技术	13
2.5 指纹库更新技术	14
2.5.1 算法概述	14
2.5.2 数据预处理	15
2.5.3 轨迹修正	15
2.5.4 指纹库更新	18
第 3 章 原理分析与硬件电路图	20
3.1 原理分析	20
3.1.1 Tag 板	20
3.1.2 Anchor	21
3.1.3 服务器	21
3.2 硬件实物图	21
3.2.1 Tag	21
3.2.2 Anchor	21
3.2.3 服务器	22
第 4 章 软件设计及调试	23

4.1 Tag 的设计和调差.....	23
4.2 自适应 AP 布局软件.....	23
4.3 快速建库辅助 APP 设计.....	24
4.4 服务器设计.....	25
第 5 章 系统测试与误差分析.....	27
5.1 测试环境介绍.....	27
5.2 自适应 AP 部署性能测试.....	27
5.3 快速建库定位性能测试.....	28
5.3.1 优化参数.....	28
5.3.2 定位结果.....	32
5.3.3 时间开销.....	32
5.4 自适应更新指纹库定位性能测试.....	32
第 6 章 总结.....	34
第 7 章 参考文献.....	35

摘要

随着移动设备便携化的快速发展,人们对于室内基于位置信息的服务需求与日俱增。全球定位系统(Global Positioning System, GPS)和蜂窝定位系统虽然在室外环境中能达到较高的定位精度,但在室内环境中信号容易受到建筑物或设施的遮挡,无法提供精确的位置服务。同时,定位精度、可扩展性和设备成本等因素限制了传统射频识别(Radio Frequency Identification, RFID)、无线局域网(Wireless Local Area Network, WLAN)、传感器(Micro-Electro-Mechanical System, MEMS)等单一定位技术的进一步发展。因此,提出一种高精度、低功耗、低成本、可扩展性强的室内定位系统势在必行。

“面向室内场馆的智能定位系统”是一个能够针对多种室内场景,采用低功耗蓝牙(Bluetooth Low Energy, BLE)/MEMS 融合技术,达到低功耗性能,实现高精度定位功能的室内定位系统。该系统的创新点在于自适应布局信号接入点(Access Point, AP)、快速构建位置指纹数据库、指纹库自动更新和精准定位算法等,避免了传统指纹定位系统的指纹库构建工作繁重、保持高定位精度困难等缺点。

实验结果表明,本作品设计的室内定位系统,利用自主设计的服务器及手机终端 APP,能够对多种室内环境自适应快速部署定位系统并保持较高的定位精度。相比传统的指纹定位系统,本套定位系统的建库工作量减小 80% 以上,定位精度达到 1.5 米,实时响应时间为 1 秒。综上,本作品积极响应国家对导航与位置服务的新要求和节能减排的战略需求,成功研制了一整套智能定位系统,实现低功耗性能,同时突破了多项室内高精度定位关键技术问题,具有较大的应用前景。

关键词: 低功耗; AP 布局; 位置指纹库; 指纹库更新; 高精度融合定位

Abstract: With the rapid development of portability of mobile devices, people's demand for indoor location based services is increasing. Although the Global Positioning System (GPS) and cellular positioning system can achieve high positioning accuracy in outdoor environment, the signals are easily blocked by buildings or facilities in the indoor environment, so as can not provide accurate location services. At the same time, the further development of single positioning technologies, such as Radio Frequency Identification (RFID), Wireless Local Area Network (WLAN), Micro-Electro-Mechanical System (MEMS), etc., are limited by the positioning accuracy, extensibility and equipment cost. Therefore, it is imperative to propose an indoor positioning system with high accuracy, low power consumption, low cost and high scalability.

The "intelligent positioning system for indoor venues" is an indoor positioning system that can achieve low power consumption performance and achieve high precision positioning for a variety of indoor scenes, using Bluetooth Low Energy (BLE)/MEMS fusion technology. The innovation of the system lies in self-adaptive

layout Access Point (AP), the fast construction of the location fingerprint database, the automatic updating of fingerprint database and the precision positioning algorithm, so as to avoid the shortcomings of traditional fingerprint localization system, such as heavy construction of fingerprint database, difficulty in keeping high positioning accuracy and so on.

The experimental results show that the proposed indoor positioning system, using self-designed server and mobile terminal APP, can adaptively and quickly deploy positioning system for many indoor environments and maintain high positioning accuracy. Compared with the traditional fingerprint positioning system, the workload of constructing database is reduced by more than 80%, the positioning accuracy is 1.5 m, and the real-time response time is 1 second. To sum up, this work actively responds to the national new requirements for navigation and location services and the strategic needs of energy saving and emission reduction. A set of intelligent positioning systems have been successfully developed to achieve low power consumption performance. At the same time, many key technical problems in indoor high precision positioning have been broken through.

Keywords: Low power consumption; APs deployment; location fingerprint database; fingerprint database updating; high precision fusion positioning

引言

随着智能终端设备的普及和无线传感器网络技术的蓬勃发展,人们对位置感知的需求越来越大。商场游客指引、医院病患跟踪、企业员工管理等基于位置的服务均离不开定位系统,尤其是室内定位系统。GPS[1]、蜂窝定位[2]等室外定位技术已经非常成熟,但在室内环境,无线信号的多径效应、人员的流动以及各种设备的位置变换因素都会对信号造成影响,使上述室外定位技术无法应用于室内场景。针对室内定位领域,现有的定位技术[3]主要包括 RFID、BLE、WLAN、MEMS 等,其中 BLE 定位技术[4]具有低功耗、信号传输距离远,自适应跳频机制、设备体积小等优势,近年来在室内定位领域应用广泛。

近年来,我国相继出台的政策表示了政府对我国导航与位置服务的大力推动工作。随着国务院《导航与位置服务科技发展“十二五”专项规划》、《国家“十二五”科学和技术发展规划》和《国务院关于加强培育和发展战略性新兴产业的决定》等相关政策的出台,为我国室内定位行业的发展孕育了一份沃土。2015 年 10 月,中央政治局委《中共中央关于制定国民经济和社会发展第十三个五年规划的建议》文件中明确指出将节能环保产业发展规划作为国家加快培育和发展的 7 个战略性新兴产业之一。所以,我们应该积极响应国家政策,将节能环保与高精度定位技术相结合,设计高精度、低功耗的室内定位服务系统,推动社会服务

业的发展。

目前存在的多种室内定位技术，定位精度达到一定的需求，设备成本与产品功耗也能达到一定的指标。但是，通过对现有的多种定位系统进行分析，还存在一些待解决的技术问题，包括构建位置指纹数据库工作量大[5]、未实现自适应布局 AP [6]、无法自动更新指纹库[7]和保持定位精度难等。

针对上述目前室内定位系统存在的问题，我们设计了一套“面向室内场馆的智能定位系统”，实现低功耗的性能和高精度的定位指标，同时解决了多项室内高精度定位关键技术问题。本文的贡献为：

1、设计自适应布局 AP 算法。根据真实定位场景的平面地图，利用该算法可以给出建议部署 AP 的位置坐标，避免了传统 AP 布局方法耗时、耗力、准确性低的缺陷；

2、设计快速构建位置指纹数据库算法。通过较少数量的指纹样本建立稀疏指纹库，随后根据 Multi-quadric[8]径向基函数（Radial Basis Function, RBF）[9]插值算法扩充稀疏指纹库并建立完整的位置指纹数据库，相比于传统的指纹库构建方法，工作量减少 80% 以上且保持较高的定位性能；

3、设计指纹库自适应更新算法。依据后期用户在同一定位环境中的定位数据更新当前的位置指纹数据库，使定位系统一直保持较高的定位性能；

4、提出一种 BLE 指纹/MEMS 数据融合定位算法，综合利用指纹定位和 MEMS 定位两种技术的优势，实现室内高精度定位。

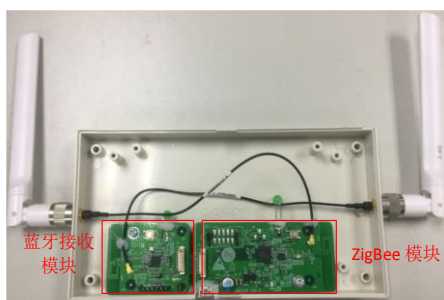
第 1 章 作品介绍

1.1 作品内容概括

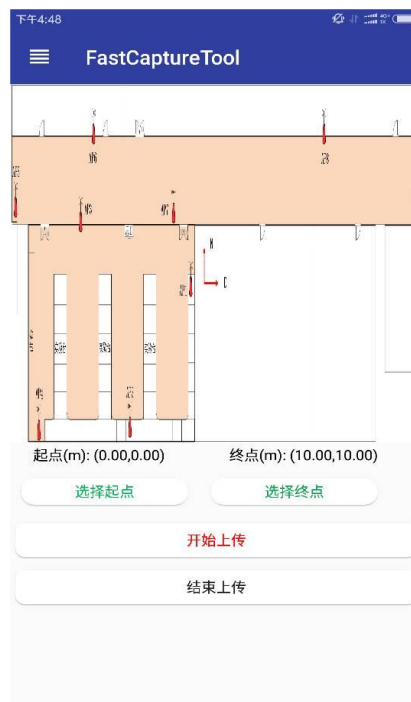
本文提出一套高精度、低成本智能室内定位系统，能够应用于多种类型的大型室内场景，包括会展中心、博物馆和学校等，满足用户对室内定位导航的需求。另外，我们研发有辅助指纹采集终端 APP、自适应 AP 部署软件、系统各模块算法代码，设计有定位解算服务器和网页显示服务器等，具有一整套的软硬件设施，如图 1-1 所示。系统各部分的功能内容如表 1-1 所示。



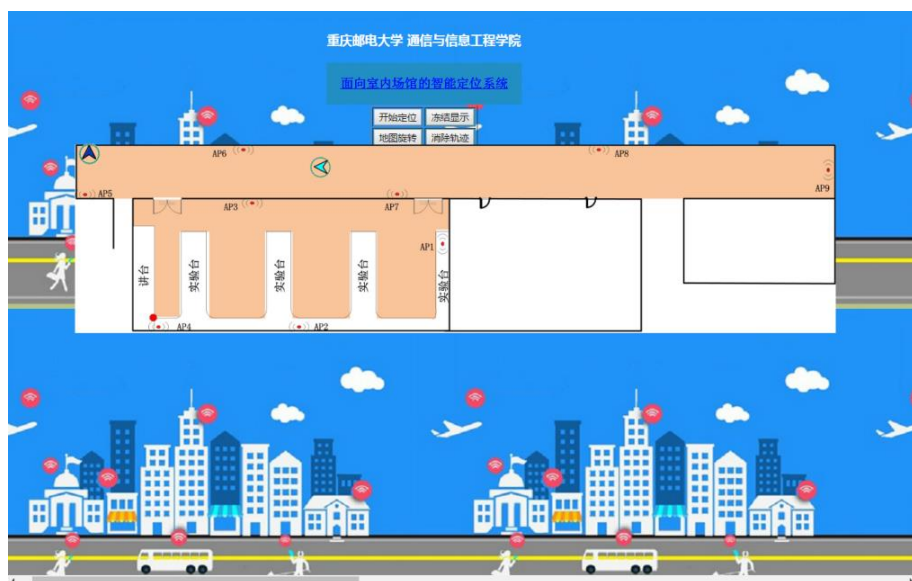
(a) Tag 板



(b) Anchor



(c) 辅助指纹采集 APP 界面



(d) 网页显示服务器

图 1-1：系统软硬件图片

本文目前拥有的成果为：国家发明专利（申请专利号：CN201610011331.4）；中文核心期刊“通信学报”收录论文《室内 BLE/MEMS 跨楼层融合定位算法》；国际顶会 IEEE GLOBECOM 收录论文“A Case Study of Cross-floor Localization System Using Hybrid Wireless Sensing”；发表期刊“Hardware and Software Design of BMW System for Multi-floor Localization ”（SCI 检索）。

表 1-1：作品主要内容

名称	主要组成部分	功能
Anchor	CC2640 芯片、BLE 信号发射天线	作为信号接入点，像服务器上传信号
Tag 板	CC2640 芯片、BLE 模块、传感器模块	向 Anchor 发送 BLE 信号，加速度计等传感器模块可以测试传感器数据
指纹采集 APP	Android 代码	安装于普通手机，用于采集并保存标记路径上的指纹样本
定位系统服务器	RBF 插值、速度与航向解算、融合定位、指纹库更新算法代码	根据 MEMS 数据、BLE 数据构建和更新位置指纹数据库、解算行人实时位置信息
网页服务器	Apache	通过网页地图显示行人实时位置坐标

1.2 作品难点

1.2.1 实现自适应 AP 布局

在传统的 AP 布局算法中，往往依靠人工经验布局，该方法耗费大量的人力和时间，布局结果的准确性不稳定。基于此，本文提出了一种基于室内定位系统的自适应 AP 布局算法。核心思想是利用贪心算法，以仿真指纹库的区分度作为目标函数，经过多次迭代优化实现最优布局或者接近于最优布局。

1.2.2 高效率地构建位置指纹数据库

传统的位置指纹库构建方法多采用逐点采集法，该方法的工作量过于繁重，并且当环境发生改变时，又需要数小时的时间重新采集指纹样本建立指纹库，不利于推广至实际的工程应用中。基于此，本文提出一种动态采集指纹样本联合径向基函数插值方法的高效指纹库构建方法。首先，利用动态采库的方法采集数条路径上的两方向指纹数据，随后利用径向基函数对稀疏指纹库进行插值，从而建立完整的位置指纹数据库。

1.2.3 保持指纹库定位精度困难

初期构建指纹库后，设备位置的移动、人群的流动等诸多环境变化因素均会

导致位置指纹库“失效”，即定位精度大大降低导致不能满足人们的定位需求。本文提出一种指纹库更新算法，利用有效的融合定位结果实时更新指纹库，保持指纹定位精度。

1.3 作品创新

1.3.1 提出自适应 AP 布局算法

本文提出一种基于贪心算法的自适应 AP 布局算法，以仿真指纹库的区分度为目标函数优化 AP 布局，经过多次迭代达到最优布局或者接近最优布局，较传统布局方法准确性更高且节约大量成本。

1.3.2 快速构建位置指纹数据库

本文提出一种快速采集指纹联合径向基函数插值的建库方法。在目标环境中动态采集部分指纹样本，然后利用径向基函数插值算法对指纹库进行扩充建立完整指纹库，建库时间开销较普通建库方法减小 80% 以上。

1.3.3 指纹库自适应更新

本文提出一种指纹数据库更新算法，使初期建立的指纹库一直保持较高的定位精度。

1.3.4 室内高精度融合定位

本文基于 BLE 指纹定位结果和 MEMS 定位结果，选择扩展卡尔曼滤波器对两者进行融合定位，可达到 1.5 米以内的定位精度。

第2章 方案论证与设计

2.1 系统框架设计

本系统主要由四个模块组成，分别为速度与航向角解算模块、蓝牙指纹定位模块、融合定位模块和指纹库自适应更新模块。

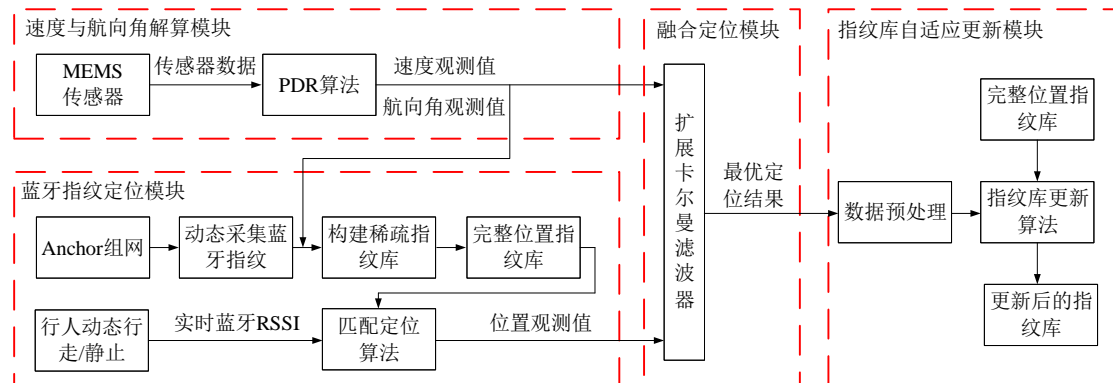


图 2-1: 系统框架图

2.2 速度及航向角解算模块

2.2.1 速度解算

本系统中 MEMS 与 BLE 指纹技术的融合以时间为单位，即每秒实现一次融合，因此需要计算每一秒的平均速度(简称秒速)。秒速是一秒内 50 个时刻(采样率 50Hz)速度的平均值。根据行人在行走时的特征分析可知，满足一定条件的加速度峰值代表一次跨步，因此根据当前这一秒数据的加速度峰值情况得到每一个时刻合理的速度，从而得到秒速度。为了充分利用加速度的数据特征，实现前后两秒之间速度求解的连续性，我们利用上一秒的最后一个波峰辅助当前的速度解算。秒速度的求解流程如图 2-2 所示，其中 $N(i)$ ，表示第 i 个波峰出现在一秒内的哪个时刻； $L(i)$ 表示第 i 步的步长， $L(0)$ 表示步长是利用上一秒最后一个波峰与当前这一秒的第一个波峰求解得到的步长； $M(i)$ 表示第 i 个波峰与第 $i+1$ 个波峰之间的间隔，即两次跨步之间的时间间隔， $M(0)$ 表示上一秒最后一个波峰与当前这一秒第一个波峰的时间间隔。

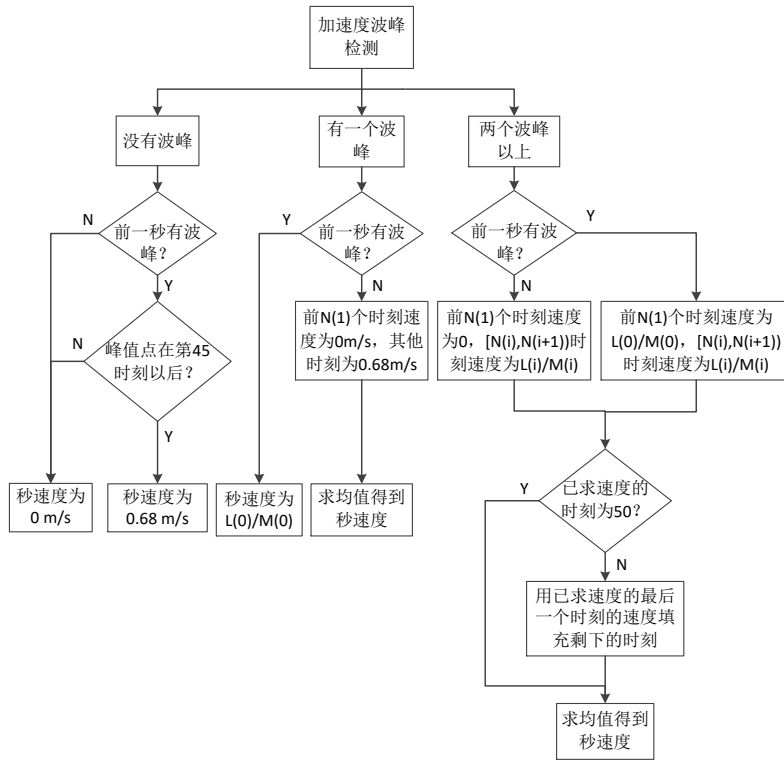


图 2-2：速度解算

2.2.2 航向角解算

解算航向角的原理如图 2-3 所示。

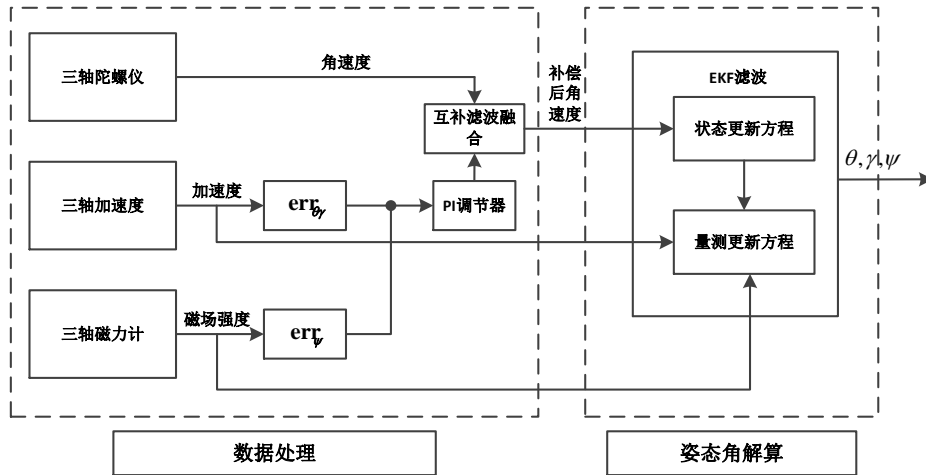


图 2-3：航向角的估计

本文采用三轴加速度计、磁强计辅助陀螺仪的形式，应用扩展卡尔曼（Extended Kalman Filter, EKF）[10]滤波器实现各传感器数据融合，计算载体航向、姿态。具体过程描述如下。

在初始静止状态下，分别测得重力加速度沿三轴的分量为 a_x ， a_y ， a_z ，则加速度矢量在载体系下表示为：

$$a^b = [a_x \ a_y \ a_z]^T \quad (2-1)$$

加速度矢量在导航坐标系下表示为：

$$a^n = [0 \ 0 \ g]^T \quad (2-2)$$

建立重力加速度在两坐标系间的转换关系式：

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = C_n^b \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = (C_b^n)^{-1} \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (2-3)$$

由于在水平方向上，加速度计输出并不对航向角产生影响，航向角可以视为 $\psi=0$ ，此时

$$C_n^b = \begin{pmatrix} \cos \gamma & 0 & \sin \gamma \\ \sin \theta \sin \gamma & \cos \theta & -\sin \theta \cos \gamma \\ -\cos \theta \sin \gamma & \sin \theta & \cos \theta \cos \gamma \end{pmatrix} \quad (2-4)$$

将上式代入式（2-3），得到

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{pmatrix} \cos \gamma & 0 & \sin \gamma \\ \sin \theta \sin \gamma & \cos \theta & -\sin \theta \cos \gamma \\ -\cos \theta \sin \gamma & \sin \theta & \cos \theta \cos \gamma \end{pmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} \sin \gamma \\ -\sin \theta \cos \gamma \\ \cos \theta \cos \gamma \end{bmatrix} \cdot g \quad (2-5)$$

对上式作矢量展开，得到初始水平姿态角：

$$\begin{cases} \gamma = \arctan \left(\frac{a_x}{\sqrt{a_y^2 + a_z^2}} \right) \\ \theta = \arctan \left(-\frac{a_y}{a_z} \right) \end{cases} \quad (2-6)$$

类似地，考察初始航向角的确定。定义地磁强度矢量在沿载体三轴的分量为 m_x, m_y, m_z 。建立载体系下矢量表达式

$$m^b = [m_x \ m_y \ m_z]^T \quad (2-7)$$

定义导航坐标系下地磁场强度矢量

$$m^n = [m_x^n \ m_y^n \ m_z^n]^T \quad (2-8)$$

地磁场强度在导航系下投影矢量与在载体系的投影矢量满足反变化关系

$$\begin{bmatrix} m_x^n \\ m_y^n \\ m_z^n \end{bmatrix} = \begin{pmatrix} \cos \gamma & 0 & \sin \gamma \\ \sin \theta \sin \gamma & \cos \theta & -\sin \theta \cos \gamma \\ -\cos \theta \sin \gamma & \sin \theta & \cos \theta \cos \gamma \end{pmatrix}^T \cdot \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} \quad (2-9)$$

由于水平姿态角 (θ, γ) 已知，则其三角函数值已知。将上式作矢量展开，水平方向的磁场分量为：

$$\begin{cases} m_x^n = m_x \cos \gamma + m_y \sin \gamma \sin \theta - m_z \cos \theta \sin \gamma \\ m_y^n = m_y \cos \theta + m_z \sin \theta \end{cases} \quad (2-10)$$

这里， m^b 中各元素可以通过磁力传感器输出获得，则磁航向角的初值估计可通过下式获得

$$\psi_{mag} = \arctan \left(\frac{m_x^n}{m_y^n} \right) \quad (2-11)$$

上述求解过程得到行人的初始姿态角，将四元素[11]作为状态变量，可求出实时的姿态角：

$$X = [q_0 \quad q_1 \quad q_2 \quad q_3]^T \quad (2-12)$$

根据捷连惯导的四元素原理，状态方程的定义如下：

$$\dot{X} = \frac{1}{2} \begin{bmatrix} 0 & -(\omega_x - w_x) & -(\omega_y - w_y) & -(\omega_z - w_z) \\ (\omega_x - w_x) & 0 & (\omega_z - w_z) & -(\omega_y - w_y) \\ (\omega_y - w_y) & -(\omega_z - w_z) & 0 & (\omega_x - w_x) \\ (\omega_z - w_z) & (\omega_y - w_y) & -(\omega_x - w_x) & 0 \end{bmatrix} X \quad (2-13)$$

其中， $\omega_x, \omega_y, \omega_z$ 为陀螺仪测量值， w_x, w_y, w_z 是陀螺仪测量噪声。把归一化的三轴加速度计和磁力计测量值作为观测量：

$$Z = [a_x \quad a_y \quad a_z \quad m_x \quad m_y \quad m_z]^T \quad (2-14)$$

对以上的状态变量、状态方程和观测量进行 EKF 循环迭代，得到最优的状态变量四元素，再根据四元素与欧拉角的转换关系输出实时最优航向角，如下式所示：

$$\psi = \arctan \left(-\frac{2(q_1 q_2 + q_0 q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2} \right) \quad (2-15)$$

2.3 蓝牙指纹定位模块

蓝牙位置指纹定位算法实施过程如图 2-4 所示，包含离线阶段和在线阶段[12]。在离线阶段，主要完成定位区域内的锚节点（Anchor）布局、坐标系的建立、指纹样本采集及构建位置指纹数据库等工作。在 Anchor 布局部分，根据蓝牙信号的传输特性及真实定位场景合理布放 Anchor，保证不同定位区域的指纹样本具有差异性。在构建位置指纹数据库部分，首先，动态采集标定路径上指纹样本，经过数据处理建立稀疏指纹库。随后利用 RBF 插值算法扩充指纹库，得到指纹颗粒度较小的位置指纹数据库。在线阶段需要完成对目标位置的估计，本文利用的指纹匹配算法为加权 K 邻近（Weighted K-nearest Neighbor, WKNN）[13] 算法。

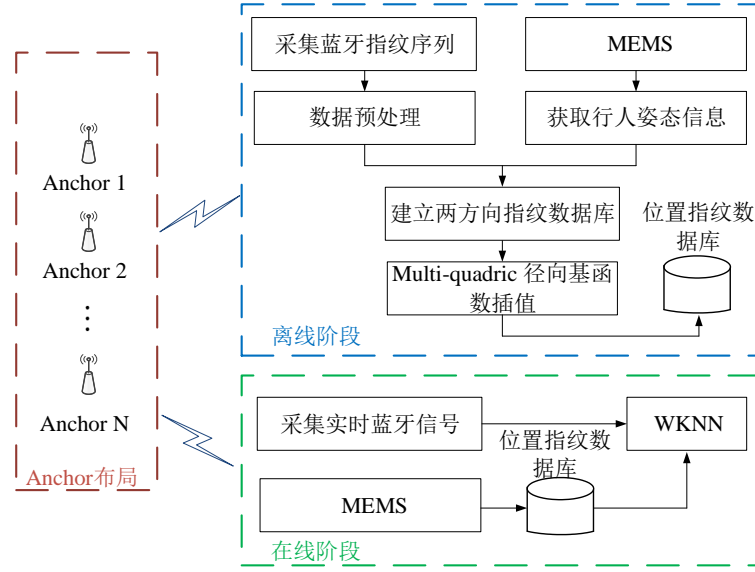


图 2-4: 基于蓝牙的位置指纹定位算法框图

2.3.1 自适应 AP 部署

在定位前期，需要对目标环境部署 AP。AP 部署算法以提高定位精度、部署效率为准则，主要包括获取需部署 AP 数目、优化 AP 的部署位置坐标。AP 数目的获取采用完全覆盖的方法。首先，根据环境信息，建立环境信息数据矩阵。然后，测试单个 AP 的有效定位区域。最后，利用完全覆盖的方法获取环境中需要布放的 AP 数目。AP 部署位置坐标的优化采用以指纹库区分度为目标函数的贪心算法。首先，在目标环境中随机部署 AP，使用信号的衰减模型建立仿真指纹库，模型表示如下：

$$PL = PLr[dB] + 10k \log(d_0) + \xi \quad (2-16)$$

其中， $PLr[dB]$ 表示距离 AP 一米处的接收信号强度（RSS，Received Signal Strength）； k 表示该环境下的无线信号的衰减因子； d_0 表示参考点和 AP 之间最短步行可达距离； ξ 表示遮蔽因子，服从均值为 0、方差为 σ 的正态分布。

依据仿真指纹库构建指纹库区分度函数，目标函数表示为：

$$\begin{cases} \alpha = \{(X_1, Y_1), (X_2, Y_2), \dots, (X_p, Y_p)\}; \\ \bar{l}_i = \sum_{j \in O(i)} \|\overline{RSS_i} - \overline{RSS_j}\| / \text{length}(O); \\ F_1(p) = \frac{1}{n} \sum_{i=1}^n \bar{l}_i; \\ F_2 = F_1 - \left(\sum_{i=1}^n (\bar{l}_i - F_1)^2 \right)^{1/2}; \end{cases} \quad (2-17)$$

其中， n 是参考点总数； p 是 AP 的数目，即信号的维； $\overline{RSS_i}$ 是第 i 个参考点的 RSS 均值向量； $O(i)$ 是满足和第 i 个参考点的物理距离小于 Len （预先确定的值，

这里取 $2m$ 为经验值) 的所有参考点的集合; $length(O)$ 是集合 O 的元素个数; l_i 是第 i 个参考点的 P 维信号强度欧氏距离平均值, $(X_1, Y_1), (X_2, Y_2), \dots, (X_p, Y_p)$ 是 P 个 AP 的位置坐标, α 是 P 个 AP 的一种放置方式, F_1 是所有参考点的 P 维信号强度欧氏距离平均值, F_2 是所有参考点信号平均欧氏距离与其标准差的差值。

最后, 以 F_2 为目标函数, 使用动态规划算法对目标环境下的 AP 位置坐标进行迭代优化, 实现最优 AP 部署。算法具体流程如下:

步骤一: 分别对 P 个 AP 进行编号 $\{AP_1, AP_2, AP_3, \dots, AP_p\}$, 并将 AP 均布放在参考点上并初始化数据: $i=1$ (i 表示第 i 个 AP); $L=1$ (L 表示 AP 移动的距离); k 表示 AP 移动的四个方向 (k 值可取 1, 2, 3, 4); ΔF_2 表示最大区分度, $\max F_2 = -\infty$ 。

步骤二: 计算当前布局下目标函数值 F_2 , 并对 AP_i 在参考点上的位置进行上下左右四个方向 (1, 2, 3, 4 四个方向) 的移动, 如图 2-5 所示为移动操作, 其中移动距离为 L 。如果第 k 个方向移动后超出定位区域, 那么该方向上计算的目标函数中 $F_{2k} = -\infty$, 那么移动后四方向的值为 $\{F_{21}, F_{22}, F_{23}, F_{24}\}$ 。

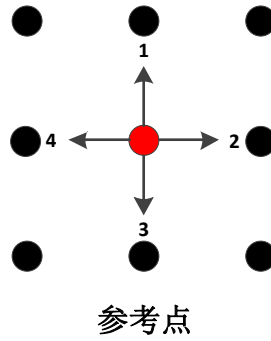


图 2-5: 移动操作

步骤三: 当 $\{F_{21}, F_{22}, F_{23}, F_{24}\}$ 的值均等于 $-\infty$ 时, 如果 i 等于 P 那么跳到步骤五, 否则 $i=i+1$, $L=1$ 并从步骤二开始。

步骤四: 当 F_2 等于 $\max\{F_{21}, F_{22}, F_{23}, F_{24}, F_2\}$ 时, 那么移动距离 $L=L+1$ 并从步骤二开始, 否则更新 F_2 , $F_2 = \max\{F_{21}, F_{22}, F_{23}, F_{24}, F_2\}$ 并将 AP_i 放置 F_2 所对应的位置上, $L=1$ 然后从步骤二开始。

步骤五: 如果 $\max F_2$ 等于 F_2 , 那么循环结束并输出此时 AP 的布局坐标, 否则 $\max F_2 = F_2$, $i=1$, $L=1$ 并从步骤二开始。

寻找到 $\max F_2$, 在此种情况下的 AP 布局可以到达最优布局或者接近最优布局。另外, 在此种 AP 布局下采集的指纹库, 其参考点间的信号欧式距离可以达到一定程度的最大化。

2.3.2 快速构建指纹数据库

本系统采用基于 MEMS 传感器辅助蓝牙的动态指纹数据库构建算法。用户

持有包含 MEMS 传感器和 BLE 模块的 Tag 板，在预先确定的路径上收集指纹，并按时间标签同步 MEMS 数据和 BLE 数据。通过计算速度和航向角等行人运动参数获得位置信息，结合参考点（Reference Point, RP）获得的蓝牙 RSS 序列，构造稀疏位置指纹数据库。

(1) 构建稀疏指纹数据库

PDR 算法[14]在长时间定位的情况下累积误差会导致定位精度下降，所以有必要修正 PDR 递推轨迹，将修正后的坐标与标准指纹库坐标匹配，从而提高稀疏指纹库的定位性能。具体过程为：

步骤一：根据标记路径数据生成标准指纹库坐标：

$$\begin{cases} X_i = x_0 + i * L_{step_x} \\ Y_i = y_0 + i * L_{step_y} \end{cases} (i \in 1, 2, \dots, \frac{|x_{end} - x_0|}{L_{step_x}}) \quad (2-18)$$

式中， (X_i, Y_i) 表示标准指纹库中第 i 个参考点的坐标， (x_0, y_0) 和 (x_{end}, y_{end}) 分别为一条标记路径的起点和终点， L_{step_x} 和 L_{step_y} 为标准指纹库参考点坐标间隔常数。

步骤二：在 PDR 递推算法中加入误差修正因子，表示如下：

$$\begin{cases} x_j = x_0 + \sum_{n=1}^j vx_n + \sum_{n=1}^j \varepsilon x_n \\ y_j = y_0 + \sum_{n=1}^j vy_n + \sum_{n=1}^j \varepsilon y_n \end{cases} \quad (2-19)$$

式中， (x_j, y_j) 为推算的行人在第 j 秒的坐标， (x_0, y_0) 为标记路径的起点坐标。 vx_j 和 vy_j 为行人在第 j 秒的速度，如式 (2-20) 所示。 ε_{xj} 和 ε_{yj} 分别为对行人在第 j 秒的坐标的修正因子，如式 (2-21) 所示。

$$\begin{cases} vx_j = v_j \sin(head_j) \\ vy_j = v_j \cos(head_j) \end{cases} \quad (2-20)$$

$$\begin{cases} \varepsilon_{xj} = \frac{vx_j}{\sum_{n=1}^{end} vx_n} (x_{end} - L_{xpd}) \\ \varepsilon_{yj} = \frac{vy_j}{\sum_{n=1}^{end} vy_n} (y_{end} - L_{ypd}) \end{cases} \quad (2-21)$$

式中， $L_{xpd} = \sum_{n=1}^{end} vx_n$ 为 PDR 递推轨迹在 X 轴方向的投影长度， $L_{ypd} = \sum_{n=1}^{end} vy_n$ 为 PDR 递推轨迹在 Y 轴方向的投影长度。

步骤三：根据步骤二，我们获得若干组参考点和相应的蓝牙 RSS 序列。接下来，将数据与标准位置指纹数据库的坐标结合起来，得到稀疏指纹数据库。

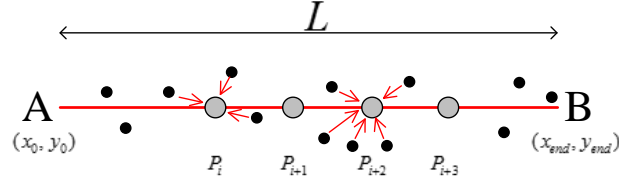


图 2-6: 位置估计

如图 2-6 所示，路径附近有多个估计位置。对于每个位置，计算位置和所有标准坐标之间的欧几里得距离，并根据最近邻（NearestNeighbor, NN）算法与相应的标准坐标匹配。以参考点 P_i 为例，该点上的蓝牙 RSS 值为：

$$RSS_n = \frac{\sum_{i=1}^M rss_i}{M} \quad (2-22)$$

式中， RSS_n 为参考点 P_i 上相应的 RSS 值， M 是匹配到该参考点上定位位置点的总个数。

步骤四：由前面的步骤，根据行人位置信息和每秒的 RSS 可以构建稀疏指纹数据库。行人的步行速度约为 1.2m/s，所以从每条标记路径指纹数据提取的指纹点存在“空洞”，本文选择基于距离的空间指纹插值算法，即计算相邻 RPs 的平均 RSS，并将其分配给没有分配指纹序列的 RPs。

在构建数据库的过程中，我们考虑了人体对蓝牙信号的遮挡因素。同时，我们考虑了不同路径的位置指纹，包括航向角信息，因此我们构建了一个多方向的位置指纹数据库。例如，如果路径的标题角是 45 度到 135 度，则设置 90 度方向数据库，相应地设置 0 度、180 度和 270 度方向数据库。RSS 矢量表示为

$$\psi_i^k = [\phi_{i,1}^k, \dots, \phi_{i,j}^k, \dots, \phi_{i,M}^k]^T \quad (2-23)$$

其中 $\phi_{i,j}^k (1 \leq i \leq P^k, 1 \leq j \leq M, 1 \leq k \leq 4)$ 表示在第 i 个参考点接收的来自第 j 个 Anchor 的 RSS 值， P^k 是 k 方向指纹库 RPs 的总数量， M 是定位区域中 Anchor 的数量。 k 方向指纹库表示为

$$\Lambda^k = \begin{bmatrix} \psi_1^k & \psi_2^k & \dots & \psi_{P^k}^k \\ x_1 & x_2 & \dots & x_{P^k} \\ y_1 & y_2 & \dots & y_{P^k} \end{bmatrix}^T \quad (2-24)$$

(2) 指纹库插值

从上一节获得了稀疏指纹数据库，参考点的颗粒度为 (1.6*0.8) 平方米/个。[15]表明，当指纹库的参考点间隔是 1 米时，定位性能和指纹库建设成本达到较好的效果。因此，我们根据插值算法对新参考点的蓝牙指纹序列进行估计，从而扩展原始稀疏位置指纹数据库。

1) 径向基函数插值模型

RBF 插值算法因其设置参数较少、应用程序简单，受到了广泛的关注。本系统利用 RBF 插值方法对给定区域内同一 Anchor 的 RSS 序列进行估计。此外，我们还特别研究和考虑了形状参数和截止区半径对插值精度的影响，从而提高了估计的准确性。

根据 RBF 构造插值模型的方法表述如下，给定 $\{x_j, y_j\}_{j=1}^n \in R^d \otimes R$ 为一组数据，函数 $\phi: R_+ \rightarrow R$ ，寻找如下形式的函数：

$$f(x) = \sum \lambda_j \phi(\|x - x_j\|), x \in R^d \quad (2-25)$$

满足 $\sum \lambda_j \phi(\|x_k - x_j\|) = y_k, k=1, 2, \dots, n$ 。其中， $\lambda_j, j=1, 2, \dots, n$ 是待定系数，采用欧式距离计算两点间距离。

求解系数矩阵 $(\lambda_1, \lambda_2, \dots, \lambda_n)$ 的线性方程组表示为：

$$\begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \dots \\ \lambda_n \end{pmatrix}^T \begin{pmatrix} \phi(\|x_1 - x_1\|) & \phi(\|x_2 - x_1\|) & \dots & \phi(\|x_n - x_1\|) \\ \phi(\|x_1 - x_2\|) & \phi(\|x_2 - x_2\|) & \dots & \phi(\|x_n - x_2\|) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\|x_1 - x_n\|) & \phi(\|x_2 - x_n\|) & \dots & \phi(\|x_n - x_n\|) \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix}^T \quad (2-26)$$

对 $\{x_j, y_j\}_{j=1}^n \in R^d \otimes R$ 中的任意两点 $x_m, x_n (x_m \neq x_n)$ ，式 (2-26) 有解的充分必要条件是对称矩阵 $\phi(\|x_k - x_j\|)$ 都是非奇异的[16]。当存在解，系数矩阵的求解公式为：

$$\begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \dots \\ \lambda_n \end{pmatrix}^T = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix}^T \begin{pmatrix} \phi(\|x_1 - x_1\|) & \phi(\|x_2 - x_1\|) & \dots & \phi(\|x_n - x_1\|) \\ \phi(\|x_1 - x_2\|) & \phi(\|x_2 - x_2\|) & \dots & \phi(\|x_n - x_2\|) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\|x_1 - x_n\|) & \phi(\|x_2 - x_n\|) & \dots & \phi(\|x_n - x_n\|) \end{pmatrix}^{-1} \quad (2-27)$$

通过求解系数矩阵并带入目标函数 $f(x)$ ，可以估计出对应于任意一点 x 的函数输出值。

2) 优化形状参数

在实际的插值逼近问题中，本文采用 Hardy 提出的 Multi-Quadric 函数[17]作为核函数，即采用函数 $\phi(x) = \sqrt{c^2 + \|x\|^2}$ 的平移作为一组基函数并形成函数空间，其中 c 为自由形状参数，

在 Multi-Quadric RBF 函数中，参数 c 影响基函数的取值，本文提出一种基于遗传算法的训练方法选择最优的 c ，利用均方根误差 (Root Mean Square Error, RMSE) 关于 c 的函数作为目标函数，使用矩阵的条件数作为约束条件。

当采样点 $\{x_j, y_j\}_{j=1}^n \in R^d \otimes R$ 确定，令 s 表示 c 的函数，表示为 $s(x, c)$ ，根据

RMSE 的表达式，有

$$\sqrt{\frac{\sum_{i=1}^n (y(x_i) - s(x_i, c))^2}{n}} = f(c) \quad (2-28)$$

其中 i -th 为对第 i 个采样点的实测值， $s(x_i, c)$ 为对第 i 个采样点的估计值， n 为样本点总数。优化过程表示为

$$\begin{cases} \text{find } c \\ \min f(c) \\ \text{cond}(\phi) < 10^{15} \end{cases} \quad (2-29)$$

3) 插值方法描述

本系统应用 Multi-Quadric RBF 插值方法逼近估计非采样点信号强度的目标函数，从而达到扩充稀疏指纹数据库的目的，流程如图 2-7 所示。

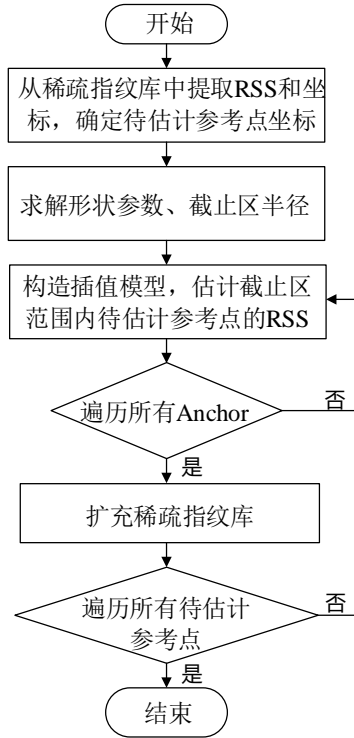


图 2-7：指纹库插值过程

2.4 融合定位技术

BLE 指纹定位结果不存在误差累积，但是受多径及人体遮挡等影响，跳动性较大；MEMS 可实现连续定位，短期内定位精度较高，但是随着时间的推移存在累积定位误差。综合 BLE 指纹定位与 MEMS 定位的特点，本文将 BLE 指纹定位与 MEMS 定位技术结合，采用 EKF 作为组合导航的数据融合算法，实现连续高精度定位。

通过位置、速度和航向角信息可以完整描述目标的运动状态，本文选择本地导航坐标系中东方向和北方向的位置坐标、速度和航向作为系统状态变量，即

$X = [E \ N \ v \ \varphi]^T$ 。将 BLE 指纹定位的位置信息、MEMS 定位中的速度和航向信息作为观测量，即 $Z = [E^{wlan} \ N^{wlan} \ v^{mems} \ \varphi^{mems}]^T$ 。根据目标运动方程，可以建立系统的状态方程为：

$$X_t = f(X_{t-1}, W_{t-1}) = \begin{bmatrix} 1 & 0 & \sin(\hat{\varphi}_{t-1}) & 0 \\ 0 & 1 & \cos(\hat{\varphi}_{t-1}) & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \hat{E}_{t-1} \\ \hat{N}_{t-1} \\ \hat{v}_{t-1} \\ \hat{\varphi}_{t-1} \end{bmatrix} + W_{t-1} \quad (2-30)$$

式中， $X_t = [E_t \ N_t \ v_t \ \varphi_t]^T$ ， E_t 为 t 时刻的东位置坐标， N_t 为 t 时刻的北位置坐标， v_t 为 t 时刻的速度， φ_t 为 t 时刻的航向角， \hat{E}_{t-1} 为 $t-1$ 时刻东位置的最优估计， \hat{N}_{t-1} 为 $t-1$ 时刻北位置的最优估计， \hat{v}_{t-1} 为 $t-1$ 时刻估速度的最优估计， $\hat{\varphi}_{t-1}$ 为 $t-1$ 时刻航向角的最优估计， W_{t-1} 是均值为 0 的高斯白噪声序列，且 $E[W_i W_j^T] = Q(i, j) \delta_{ij}$ ， $i, j = 1, 2, \dots, m$ ， m 为需要估计的变量个数， $Q(Q \geq 0)$ 为 m 维过程噪声协方差矩阵。

观测方程为：

$$Z_t = h(X_t, V_t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} E_t \\ N_t \\ v_t \\ \varphi_t \end{bmatrix} + V_t \quad (2-31)$$

式中， $Z_t = [E_t^{wlan} \ N_t^{wlan} \ v_t^{mems} \ \varphi_t^{mems}]^T$ ， E_t^{wlan} 为 t 时刻 BLE 定位出的东位置， N_t^{wlan} 为 t 时刻 BLE 定位出的北位置， v_t^{mems} 为 t 时刻 PDR 输出的速度， φ_t^{mems} 为 t 时刻 PDR 输出的航向； E_t 为 t 时刻待估计的东位置， N_t 为 t 时刻待估计的北位置， v_t 为 t 时刻待估计的速度， φ_t 为 t 时刻待估计的航向， V_t 是均值为 0 的高斯白噪声序列，且 $E[V(i) V^T(j)] = R(i, j) \delta_{ij}$ ， $i, j = 1, 2, \dots, n$ ， n 为观测变量个数， $R(R > 0)$ 为 n 维观测噪声协方差矩阵。

2.5 指纹库更新技术

2.5.1 算法概述

系统总体框架如图 2-8 所示，包括数据获取模块、数据预处理模块和指纹库更新模块。首先，收集用户上传的 RSS、速度及航向和陀螺仪模值数据[18]。然后，将 RSS 和陀螺仪模值用于信标识别生成波峰信标和拐点信标，将 PDR 算法推算出的航迹与识别出的信标作为轨迹修正及指纹库匹配模块的输入，进行轨迹修正及指纹库参考点临近点匹配生成候选指纹。最后，基于负指数时效性模型、标准指纹、相似性过滤等准则结合新旧指纹库进行指纹库更新。

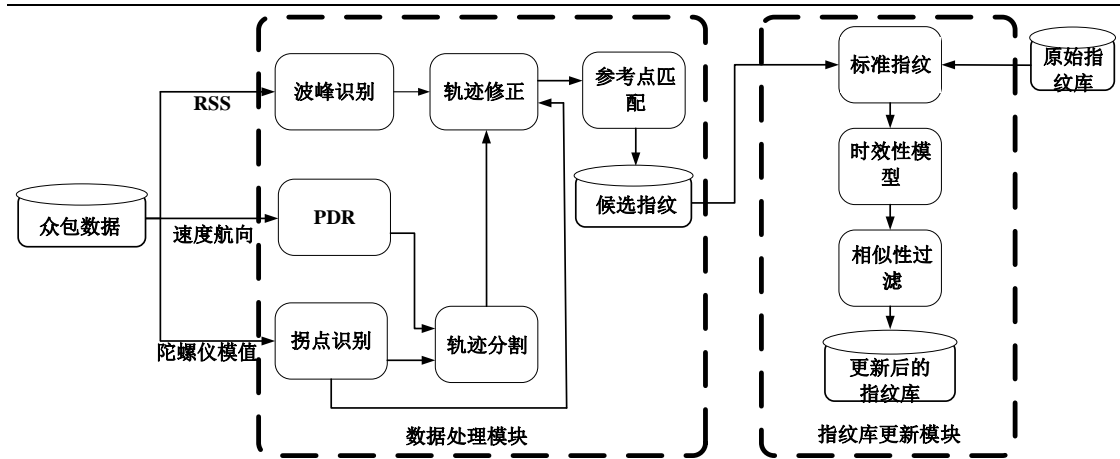


图 2-8: 指纹库自适应更新框架

2.5.2 数据预处理

本系统依赖智能手机终端内置的 BLE 模块上传 RSS 信息，同时利用三轴加速度计、三轴陀螺仪和三轴磁力计对行人速度和航向角进行估计[19]，带入 PDR 算法推导行人航迹。随着时间的推移，PDR 递推结果存在累积误差。如图 2-9 所示，红色轨迹为 PDR 推算结果，黑色实线为真实轨迹。因此，本文提出航迹修正方案对 PDR 轨迹进行修正。

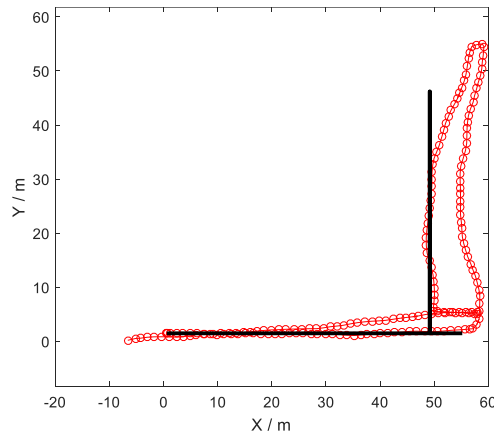


图 2-9: PDR 航迹推算结果

2.5.3 轨迹修正

由信号的传播模型，可知距离 AP 越远 RSS 越低。如下式所示，A 为距离 AP 一米处的 RSS，n 为根据实测数据训练出来的一个比例系数。

$$RSSI = -(10n \log d + A) \quad (2-32)$$

图 2-10 为某 AP 的 RSS 随时间变化趋势，可以看出经过 AP 时 RSS 最大，因此根据 AP 的 RSS 波峰出现的时间，可识别出行人某时刻经过了某个 AP。行人经过了某个 AP 时波峰必大于一个阈值，根据经验值将该阈值设置为-50dBm。此时可将 AP 的坐标作为行人当前位置，本文称其为波峰信标。

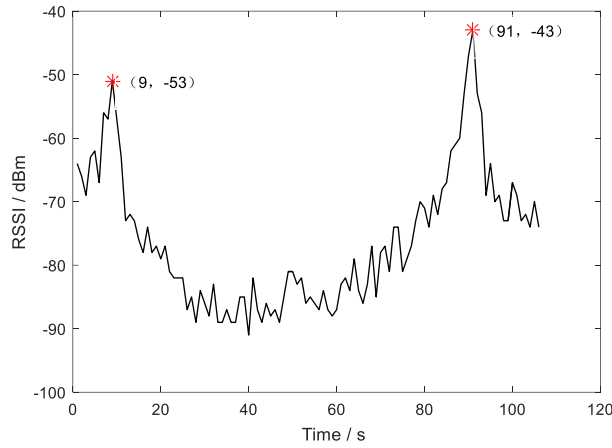


图 2-10: 某 AP 的 RSSI 随时间变化趋势

首先计算三轴陀螺仪的总模值:

$$g_i^{total} = \sqrt{(g_i^x)^2 + (g_i^y)^2 + (g_i^z)^2} \quad (2-33)$$

其中 g_i^{total} 为第 i 个采样点三轴陀螺仪的总模值, (g_i^x, g_i^y, g_i^z) 为陀螺仪第 i 个采样点的三轴输出。

通过陀螺仪的模值变化可判别出行人某时刻进行了拐弯。当识别出某时刻进行拐弯后, 需要进一步识别在哪个拐点进行了拐弯, 在拐点处采集 RSSI 特征信息, 当识别出拐弯后直接用当前 RSSI 信息和事先在拐点处采集的 RSSI 特征信息做对比直接识别出拐点。此时可将拐点的坐标作为行人当前位置, 本文称其为拐点信标。

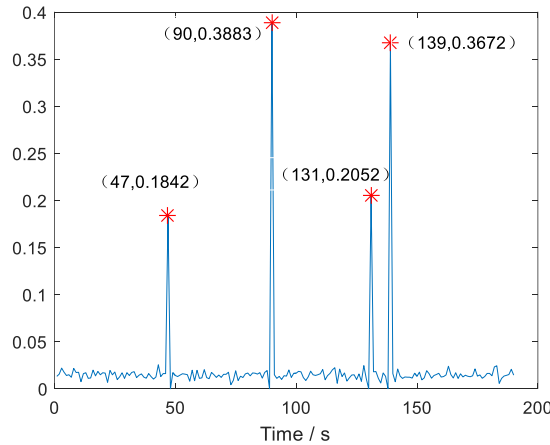


图 2-11: 陀螺仪模值识别拐弯

当完成信标识别后, 根据拐点信标的出现时间顺序对 PDR 轨迹进行分割, 将大段的轨迹分割为小段直线, 便于接下来的轨迹修正。分割后的轨迹如图 2-12 所示, 不同的颜色及符号代表分割后不同的轨迹片段。

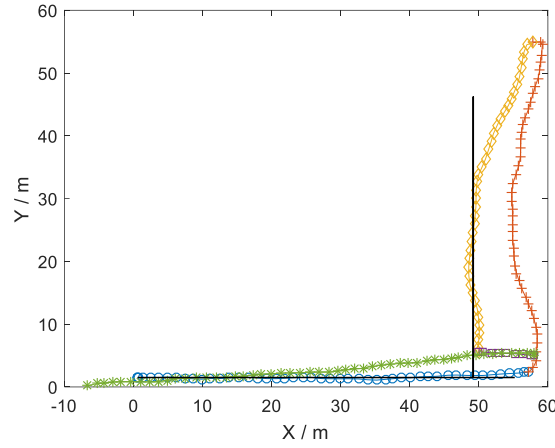


图 2-12：轨迹分割

然后利用波峰信标、拐点信标对应的地理坐标对分割为片段后的 PDR 轨迹进行修正，减小了 PDR 轨迹长时间的累计误差，从而提高利用行人航迹更新指纹库的精度。具体修正方式如公式 2-34 所示。

$$\begin{cases} x_i = x_0 + \sum_{n=1}^i v_n \sin(\theta_n) + \sum_{n=1}^i \varepsilon_{xn} \\ y_i = y_0 + \sum_{n=1}^i v_n \cos(\theta_n) + \sum_{n=1}^i \varepsilon_{yn} \\ \varepsilon_{xi} = \frac{v_i \sin(\theta_i)}{\sum v_i \sin(\theta_i)} (L \sin(\psi_1) - L_{PDR} \sin(\psi_2)) \\ \varepsilon_{yi} = \frac{v_i \cos(\theta_i)}{\sum v_i \cos(\theta_i)} (L \cos(\psi_1) - L_{PDR} \cos(\psi_2)) \end{cases} \quad (2-34)$$

其中 v_n 、 θ_n 分别为速度与航向角， ε_{xi} 、 ε_{yi} 分别为 x、y 方向的误差， L 为前一个信标至后一个信标的真实长度而 $L_{PDR} = \sum_{i=1}^N v_i$ 为 PDR 算法推算出来的轨迹长度。角度 ψ_1 为后一个信标坐标 (x_{end}, y_{end}) 与前一个信标坐标 (x_0, y_0) 的夹角，角度 ψ_2 为 PDR 推算的终点坐标 (x, y) 与起点坐标 (x_0, y_0) 的夹角。PDR 推算结果与经信标修正后的效果对比如图 2-13 所示，显然经修正后的轨迹已较为贴合真实轨迹。

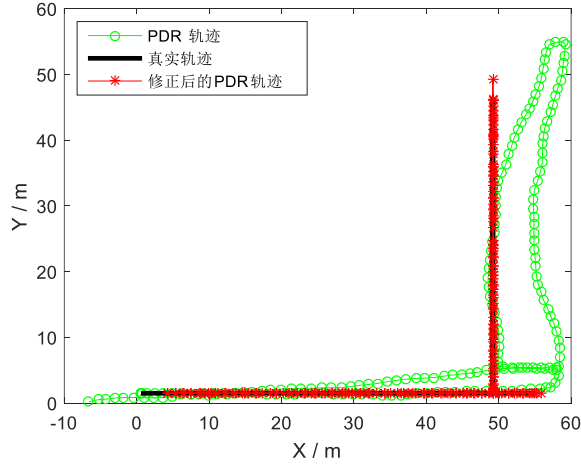


图 2-13: 信标轨迹修正效果

根据修正后的 PDR 航迹推算结果，提取地理坐标并结合 RSS，匹配至距离最近参考点坐标，生成候选指纹。临近点匹配方式如图 2-14 所示，点 a、b、c 为参考点，点 1、2、3、4 为 PDR 定位结果经信标修正后的点。

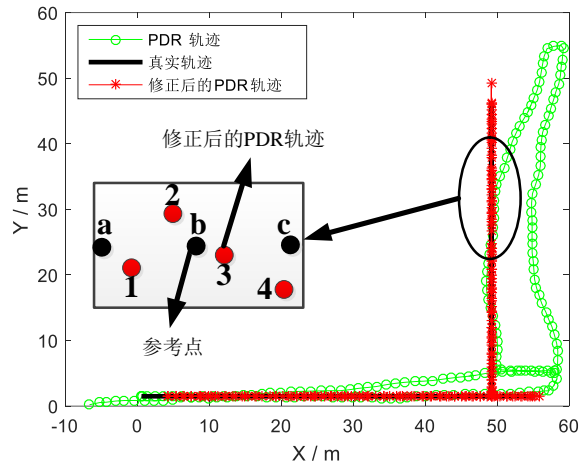


图 2-14: 临近点匹配

$$(x_{RF}, y_{RF}) = \arg \min(\sqrt{(x - x_{RF})^2 + (y - y_{RF})^2}) \quad (2-35)$$

式中， x_{RF} 及 y_{RF} 为参考点，也就是实际存储于指纹库中的坐标。 x 及 y 为 PDR 定位结果经信标修正后的坐标，通过公式 (2-35) 可将指纹信息匹配至相应的参考点上去。最终处理完所有新采集的数据后，便获取到了大量候选指纹。

2.5.4 指纹库更新

当新指纹库生成之后便可基于负指数时效性模型、标准指纹、相似性过滤等更新准则对指纹库进行更新。

首先，设结合新指纹库与原始指纹库计算标准指纹，设原始指纹库中某个参考点的指纹序列为 $S_m = \{RSS_{m1}, RSS_{m2}, \dots, RSS_{mn}\}$ ，更新后该参考点的指纹序列为 $S_k = \{RSS_{k1}, RSS_{k2}, \dots, RSS_{kn}\}$ ，那么该参考点的标准指纹可定义为公式 (2-36)。标准指纹可用于与指纹库及新采集的指纹进行相似度计算，通过滤除低相似度指纹

及保留高相似度指纹，可动态的保持指纹库的准确性与稳定性。

$$S_s = \frac{\sum_{m=1}^M S_m + \sum_{k=1}^K S_k}{M + K} \quad (2-36)$$

指纹间的相似性与指纹间信号强度的欧氏距离相关，定义为公式（2-37），其中 α 为一个常数， $|S_i - S_s|$ 为指纹与标准指纹间信号强度的欧式距离，显然 $|S_i - S_s|$ 越大指纹间相似性越小， $|S_i - S_s|$ 越小指纹间相似性越高

$$\text{Sim}_i = \alpha \times (|S_i - S_s|)^{-1} \quad (2-37)$$

当采集到的更新数据 S_k 量较少时，由公式（2-36）可得出更新数据 S_k 对标准指纹影响较小，导致标准指纹依旧偏相似于指纹库中的原始指纹，很难准确反应环境变化对 RSS 的影响。因此，考虑加入负指数时效性模型，随着时间的增长老旧的指纹信息会逐渐降低权重，减少其对标准指纹库的影响。从而重新定义标准指纹为公式（2-38），其中 $R(t)$ 为负指数时效性模型。

$$S_s = \frac{\sum_{m=1}^M S_m \times R(t_m) + \sum_{k=1}^K S_k \times R(t_k)}{\sum_{m=1}^M R(t_m) + \sum_{k=1}^K R(t_k)} \quad (2-38)$$

$$R(t) = \begin{cases} \exp(-3.725 \times 10^{-6}(t - t_0)) & t - t_0 < 15\text{day} \\ 0 & t - t_0 \geq 15\text{day} \end{cases} \quad (2-39)$$

第 3 章 原理分析与硬件电路图

3.1 原理分析

系统架构如图 3-1 所示。离线阶段，Anchor 使用 BLE 模块扫描 Tag 发送的蓝牙 RSS 及解算得到的速度和航向角，然后使用 WIFI 将数据发送到无线网关，接着使用网络助手保存数据，并将辅助建库 APP 上传的数据保存，最后通过插值算法生成指纹库。在线阶段，采集 Tag 发送的蓝牙 RSS 和 MEMS 数据，在 JAVA 服务器中解算当前点位置坐标并于标准指纹库进行指纹匹配，最终得出定位结果并在网页服务器上显示当前位置坐标。指纹库自适应更新阶段主要利用在线定位的数据，通过机器学习的算法，更新原有指纹库数据，保持指纹库定位精度。

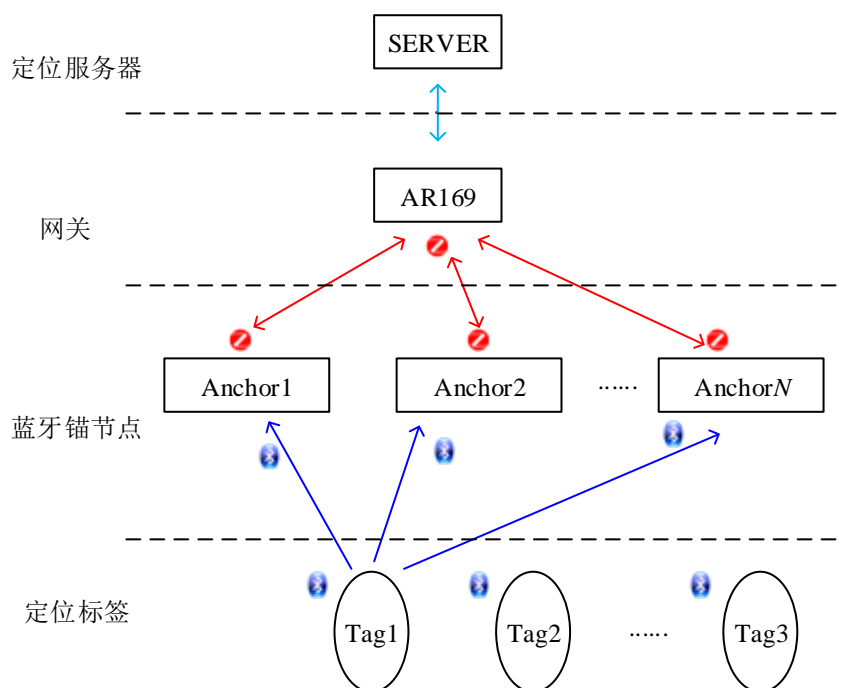


图 3-1: 系统框架图

3.1.1 Tag 板

Tag 板是以 CC2640 芯片为核心集成外围电路，外围电路主要有天线、电源、稳压等模块。CC2640 包含作为主处理器，运行在 48MHz 的 32 位 ARM Cortex-M3 处理器和独特的超低功耗传感器控制器，并且使用 8KB SRAM。CC2640 芯片做为 BLE 标签，主要功能为以 10Hz、0dBm 发送广播报文。CC2640 芯片的原理图如图 3-2 所示。

第十三届中国研究生电子设计竞赛 面向室内场馆的智能定位系统

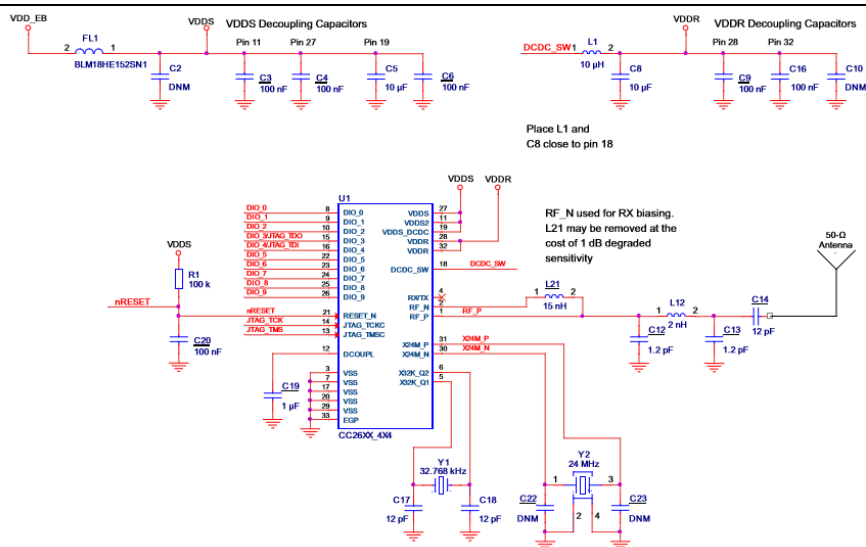


图 3-2: Tag 核心板原理图

3. 1. 2 Anchor

Anchor:包括蓝牙模块和 WiFi 模块。其中蓝牙模块用于和 Tag 进行通信，扫描 Tag 的广播报文，并将报文里面的蓝牙 RSS 以及传感器信息读取出来，然后使用 WiFi 模块将读取出来的信息上传服务器。

3. 1. 3 服务器

服务器采用普通的家用计算机，Windows7 操作系统，CPU 为 i3-3220 处理器，内存为 4GB。

3. 2 硬件实物图

3. 2. 1 Tag

如图 3-3 所示为 Tag 板及整体效果图

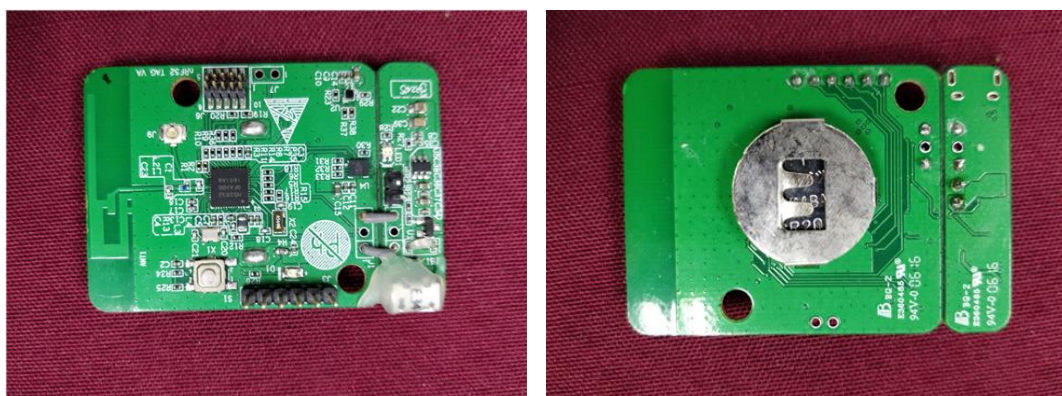


图 3-3: Tag 板实物图

3. 2. 2 Anchor

如图 3-4 所示为 Anchor 板实物图



图 3-4: Anchor 实物图

3.2.3 服务器

如图 3-5 所示为服务器定位界面

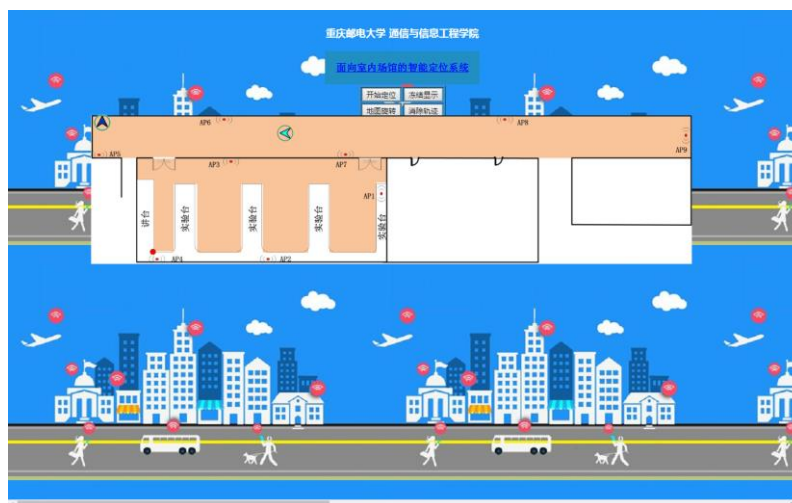


图 3-5: 服务器界面图

第 4 章 软件设计及调试

4.1 Tag 的设计和调差

本系统使用 CC2640 作为主处理器，功率设置为 0dBm，信号的广播频率为 10Hz。

```
HCI_EXT_SetTxPowerCmd(LL_EXT_TX_POWER_0_DBM); //发射功率设置
//Minimun connection interval(units of 1.25ms, 80=100ms)
#define DEFAULT_DESIRED_MIN_CONN_INTERVAL 80 //发射频率设置
```

4.2 自适应 AP 布局软件

该软件在 Eclipse 开发平台上利用 Java 语言编程实现，对目标环境进行自适应 AP 布局，满足信号的覆盖率和指纹区分度最好的指标，主界面如图 4-1 所示。



图 4-1: AP 布局主界面

该软件主要组成部分为：（1）GUI_AP.java：负责软件的 GUI 显示和相应；（2）APInitPosition.java：根据传输模型及地图信息，建立 AP 最初最佳位置，减少算法执行的复杂度；（3）ObjectiveFunction.java：优化目标函数，其值越大，AP 布局效果越好；（4）GreedyAlgorithm.java：以贪心思想搜寻 AP 布局空间；其项目工程如图 4-2。












 AP_Main.java	2018/4/19 20:14	JAVA 文件	3 KB
 APInitPosition.java	2018/4/19 15:47	JAVA 文件	7 KB
 GoalPoint.java	2017/12/19 14:37	JAVA 文件	1 KB
 GreedyAlgorithm.java	2018/1/31 15:25	JAVA 文件	6 KB
 GUI_AP.java	2018/4/24 15:02	JAVA 文件	7 KB
 MapRecognize.java	2018/4/24 16:11	JAVA 文件	10 KB
 ObjectiveFunction.java	2018/4/19 19:19	JAVA 文件	5 KB
 ShowMap.java	2018/3/13 21:03	JAVA 文件	2 KB
 SignalPointMatrix.java	2018/3/5 21:06	JAVA 文件	2 KB
 VirtualSourceFunction.java	2018/3/15 16:11	JAVA 文件	10 KB
 VisioMain.java	2018/4/24 15:23	JAVA 文件	3 KB

图 4-2: AP 布局代码

4.3 快速建库辅助 APP 设计

软件主要分为四个模块，分别实现不同的功能，从而保证快速建库方案（根据定直线建库的方法，完成进行数据的传输、保存）的完成，如图 4-3 所示。

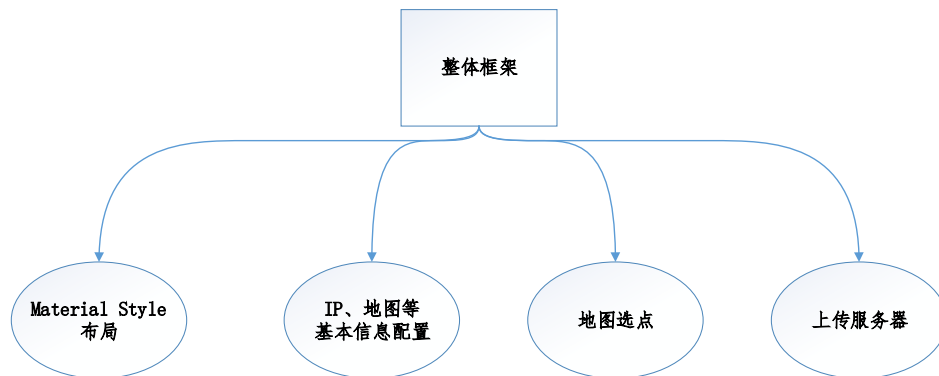


图 4-3: APP 设计框架

通过在地图上选取行走轨迹，然后上传服务器，最终服务器端通过轨迹信息实现快速建库。详细流程如下图所示。

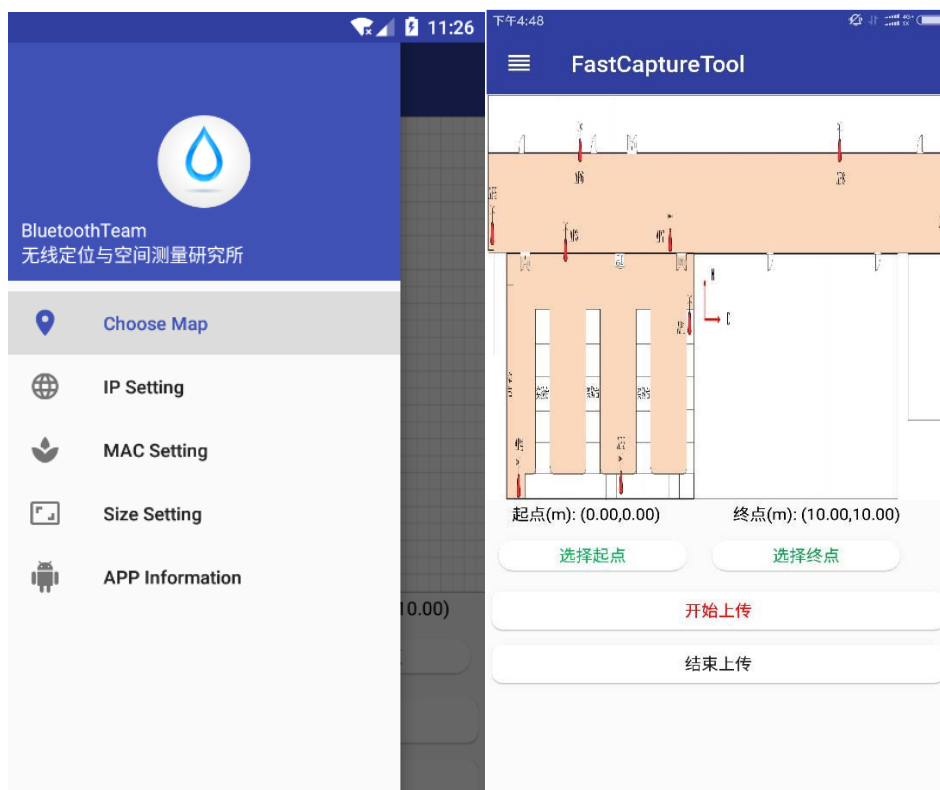


图 4-4: 应用界面

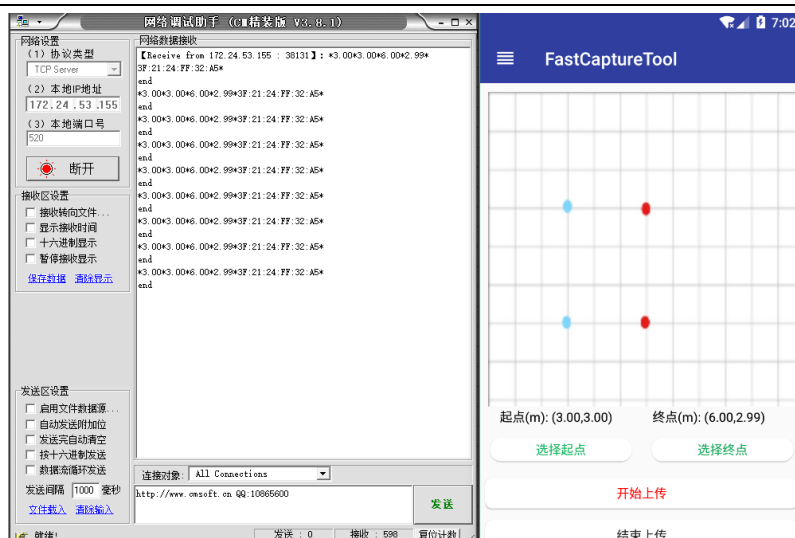


图 4-5: 通信界面

4.4 服务器设计

服务器主要分为 Java 定位服务器、MySQL 数据存储服务器、定位结果显示服务器等几种网页服务器。Java 定位算法和 MySQL 数据存储算法均集中在电脑服务器, 电脑端通过运行算法解算出目标的实时定位结果并将定位结果依次存储入数据库中, 同时在网页服务器界面能够显示出用户的实时坐标点。网页服务器的开发界面如图 4-6、4-7 所示。

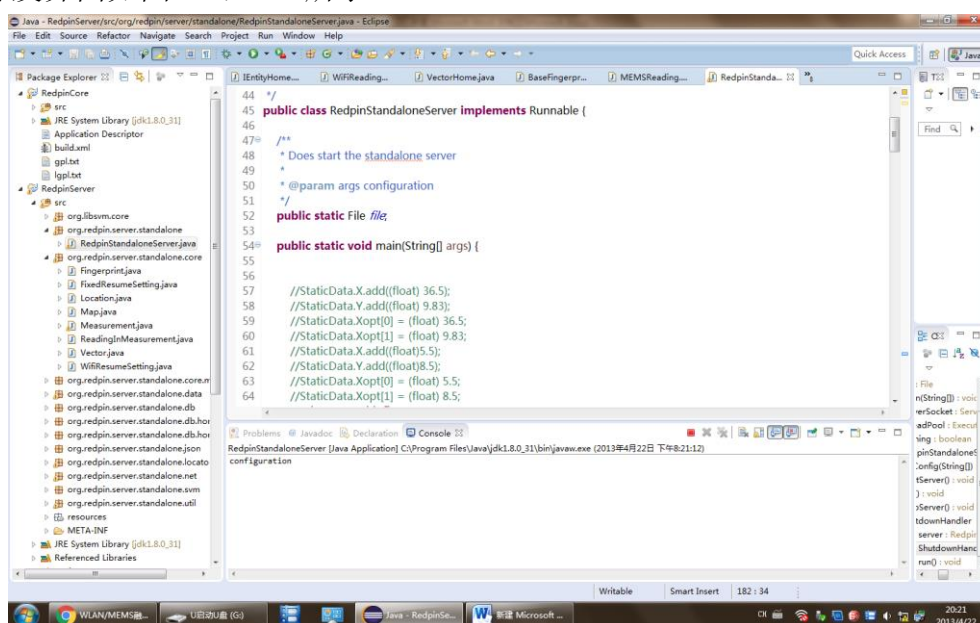


图 4-6 服务器源码

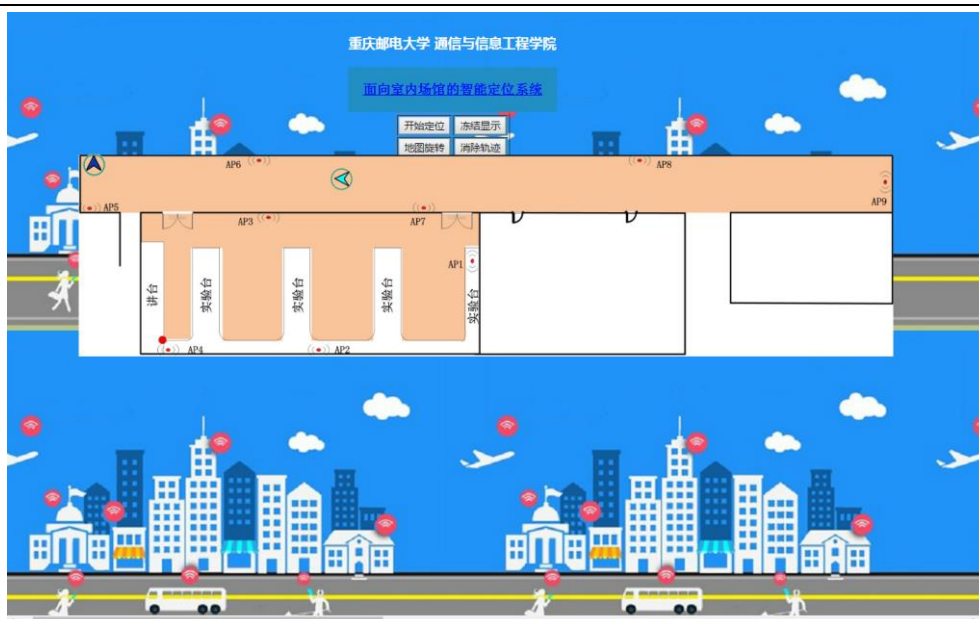


图 4-7：网页服务器开发界面

第5章 系统测试与误差分析

5.1 测试环境介绍

以重庆邮电大学行政楼一楼为测试环境，该环境为典型的办公环境，总面积约为 $65 \times 17 \text{ m}^2$ ，其中大厅面积约为 140 m^2 ，走廊面积约为 116 m^2 ，由大厅、走廊、多间办公房间组成，如图 5-1 所示，彩色区域为测试区域，红色小塔为 Anchor，Anchor 的布放位置由下面的自适应 AP 布局算法得出。

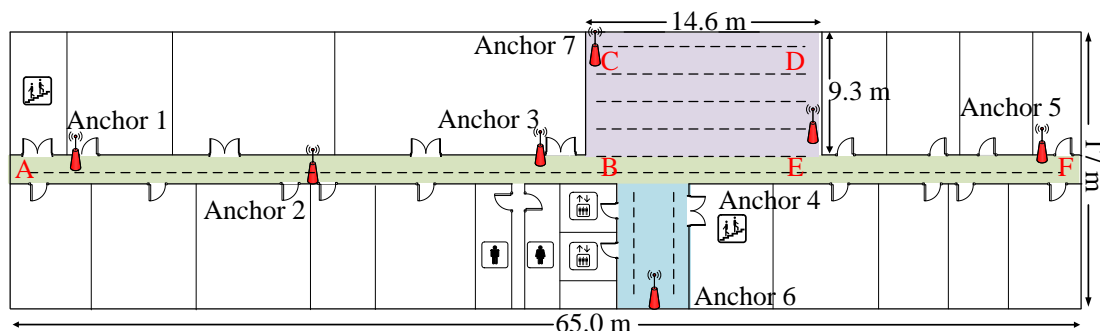


图 5-1：实验环境平面图

5.2 自适应 AP 部署性能测试

本节对比利用传统经验布局方式和本文提出的自适应 AP 布局方式得到的结果，图 5-2 为利用本文提出的自适应布局算法得到的 AP 位置示意图，图 5-3 为利用经验布局算法得到的 AP 位置示意图。基于两种方案，我们分别进行数据采集、构建指纹库以及定位精度分析。图 5-4 为在两种 AP 布局方式下所建立指纹库对应的误差累积分布函数(Cumulative Distribution Function, CDF)，可以看出，本文提出的自适应布局方法所建立的指纹库定位精度优于经验布局方式建立指纹库的定位精度。

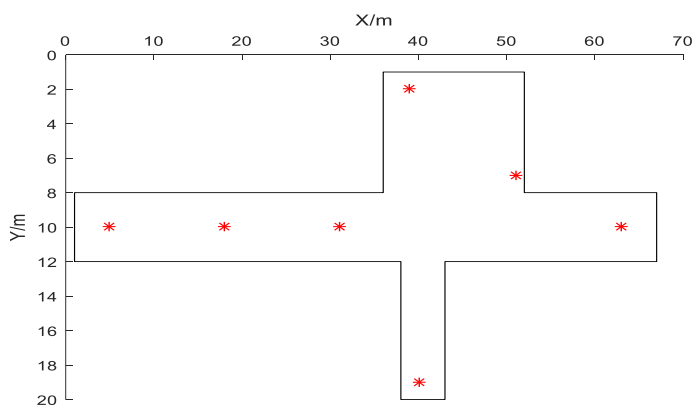


图 5-2：利用自适应算法得到的 AP 位置示意图

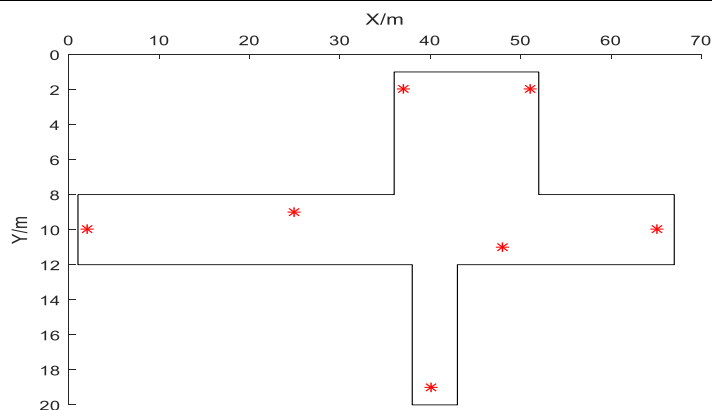


图 5-3: 利用经验布局算法得到的 AP 位置示意图

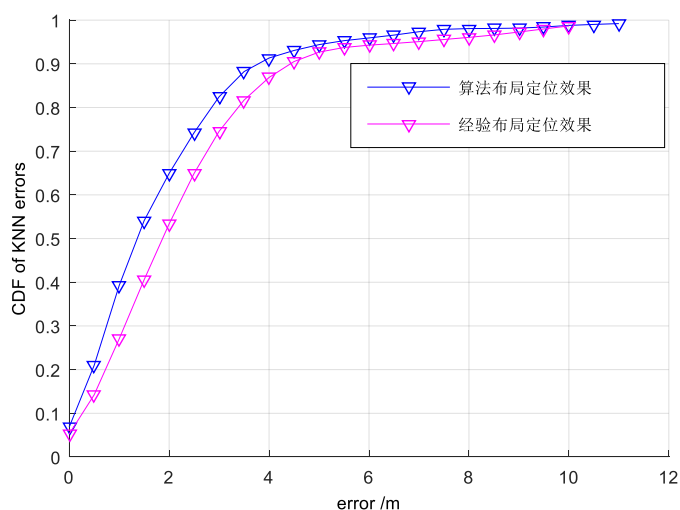


图 5-4: 两种 AP 布局方式所建立指纹库对应的 CDF 图

5.3 快速建库定位性能测试

5.3.1 优化参数

(1) 指纹样本容量

图 5-5 显示了在同一路径上动态收集的 4 次指纹样本的 RSS 变化趋势，图 (a) 和图 (b) 分别显示了行人接收的来自 Anchor2 和 Anchor6 的 RSS。可以发现，由 BLE 平台动态收集的蓝牙指纹数据具有稳定的特征。因此，我们在标记路径上只行走一个来回，从而收集每条路径上两个方向的指纹样本序列。

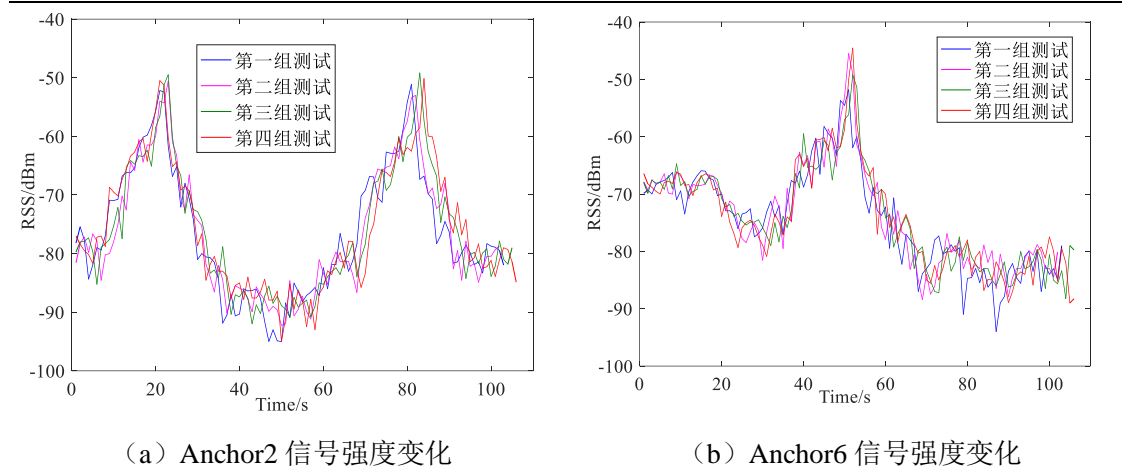


图 5-5: 信号强度变化趋势

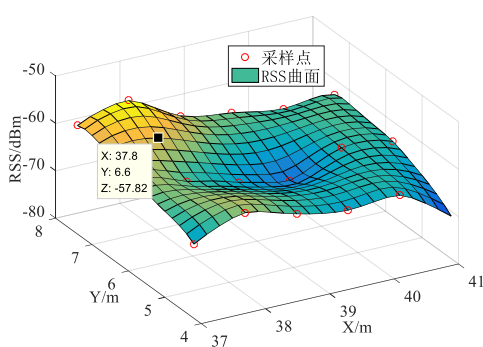
为了确定合适的标记路径间隔, 分别根据 1.6 米、2.4 米和 3.2 米的间隔设计路径的位置, 收集指纹并构造指纹数据库。表 5-1 显示了指纹数据库不同百分比值的定位误差。可以看出, 当路径间隔为 1.6 米时, 指纹数据库的定位误差最优, 因此我们设置标记路径间隔为 1.6 米。

表 5-1: 指纹库百分比定位误差

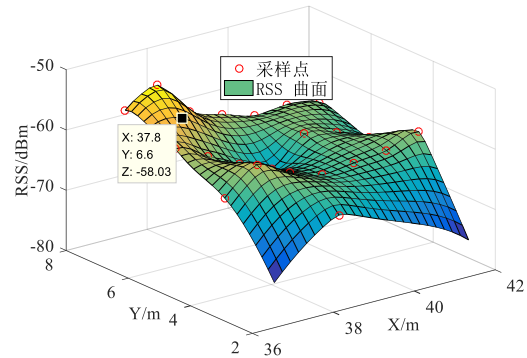
百分比 (%)	间隔=1.6 米	间隔=2.4 米	间隔=3.2 米
50	<2.0 米	<2.2 米	<2.5 米
70	<2.8 米	<3.1 米	<3.4 米
90	<4.7 米	<4.8 米	<5.0 米

(2) 截止区半径

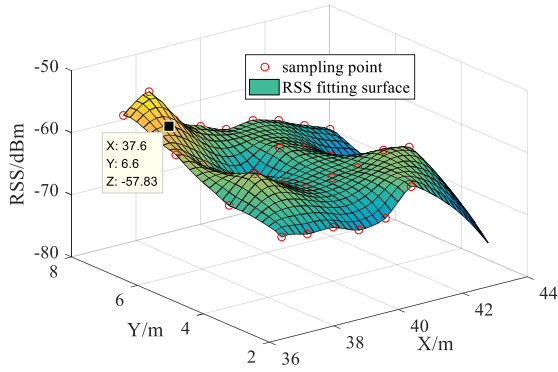
为了验证利用 RBF 函数对稀疏指纹库插值方法的有效性, 我们随机选择一个采样点, 坐标是 (37.8, 6.6), 实际测量的 RSS 是 -62.50dBm。基于采样点的坐标和来自 Anchor7 的 RSS, 我们获得 RSS 表面, 如图 5-6 所示, 可以发现, 在一定的范围内, 利用 Multi-quadric RBF 插值方法可以很好地估计非采样点的 RSS 值。



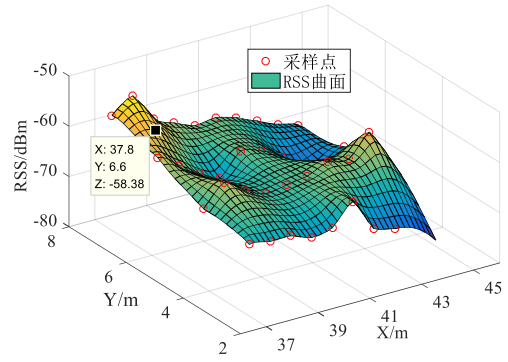
(a) 截止区半径为3米



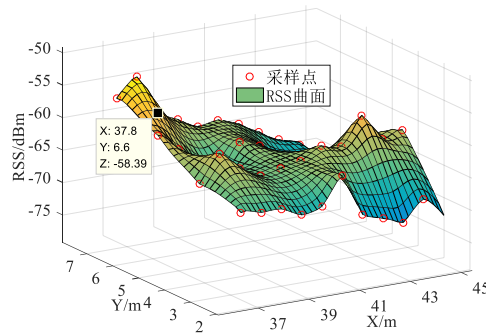
(b) 截止区半径为4米



(c) 截止区半径为5米



(d) 截止区半径为6米



(e) 截止区半径为6米

图5-6: 对应不同截止区半径的RSS曲面图

为了进一步确定截止区半径，在定位环境中随机选择 16 个参考点测量 RSS 序列，随后根据截止区半径取不同值分别估计这些参考点的 RSS 序列。表 5-2 为这些参考点估计 RSS 与实测 RSS 的差。可以看出，当半径取 3 米，4 米时，存在较大的估计误差，当半径取 5 米，6 米和 7 米时，RSS 估计误差较小。另外，图 5-6 显示了不同截止区半径对应的 CDF 图。可以看出，截止区半径 5m 所对应的指纹库定位精度优于其他情况，3 米内的置信率为 77%。综合以上讨论，本文选择截止区半径为 5 米。

表 5-2: 参考点 RSS 估计值与实测值之差 (dBm)

坐标	半径=3 米	半径=4 米	半径=5 米	半径=6 米	半径=7 米
(41.8,6.6)	5.34	5.24	1.98	2.00	2.01
(42.6,6.6)	3.12	1.64	1.95	1.96	1.97
(45.00, 6.60)	0.15	4.18	3.16	3.99	3.92
(37.00, 5.00)	2.22	7.74	0.48	0.47	0.42
(40.20, 5.00)	1.35	5.16	1.45	1.45	1.46
(42.60, 5.00)	3.19	1.35	1.35	1.44	1.44
(44.20, 6.60)	3.72	1.85	1.86	1.86	1.86

(46.60, 6.60)	0.49	5.00	1.00	1.00	1.00
(38.60, 3.40)	3.00	3.12	2.78	2.88	2.81
(39.40, 3.40)	3.79	2.40	2.48	2.54	2.56
(40.20, 3.40)	2.11	2.19	2.30	2.34	2.40
(41.80, 3.40)	1.72	5.97	1.16	1.21	1.22
(42.60, 3.40)	1.18	6.32	2.13	1.21	2.22
(43.40, 3.40)	1.94	1.62	1.76	1.84	1.90
(45.80, 3.40)	1.27	8.24	0.83	0.76	0.68

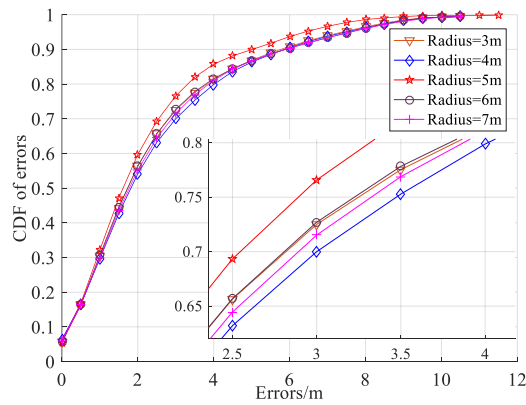


图 5-7: 不同截止区半径对应的 CDF 图

(3) 形状参数

图 5-8 显示了基函数矩阵的矩阵条件数与形状参数随着形状参数增加的变化, 可以看出, RMSE 在 0 到 14 的范围内逐渐增加, 在形状参数大于 14 后出现震荡; 基函数的矩阵条件数在形状参数取 14 之后开始大于 10^{15} , 综上, 应该在 0 到 5 的范围内求解形状参数的具体值。本文利用遗传算法, 通过设置种群大小为 50, 交叉概率为 0.8, 变异概率为 0.04 和迭代次数为 200, 寻优的形状参数取 0.07。

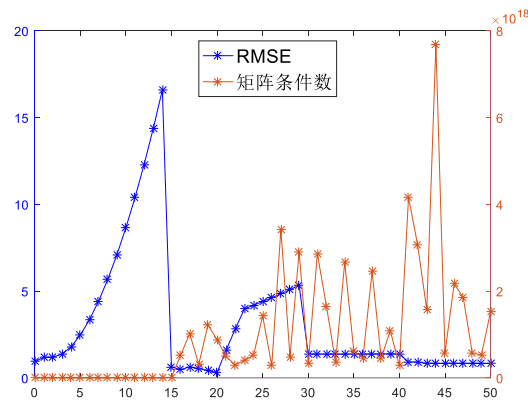


图 5-8: 矩阵条件数与均方根误差变化曲线

5.3.2 定位结果

图 5-9 比较了由逐点采集方法 (Point-by-Point, PbP)、定点采集指纹联合 RBF 插值方法 (Static Collection of Fingerprints and RBF interpolation, SCF-RBF)、动态采集指纹方法 (Dynamic Collection of Fingerprints, DCF) 和本系统提出的动态采集部分指纹联合 Multi-quadric RBF 插值方法 (Dynamic Collection of Fingerprints and Multi-Quadric RBF interpolation, DCF-MQRBF) 所建立指纹库对应的定位误差情况图, 测试路径为 A->B->C->D->E->F->E->D->C->B->A。随后, 选择 20 组测试路径数据并汇总 CDF 图, 如图 5-10 所示。可以看出, 与其他建库方法相比, 本文提出的算法可以达到相当的定位精度。

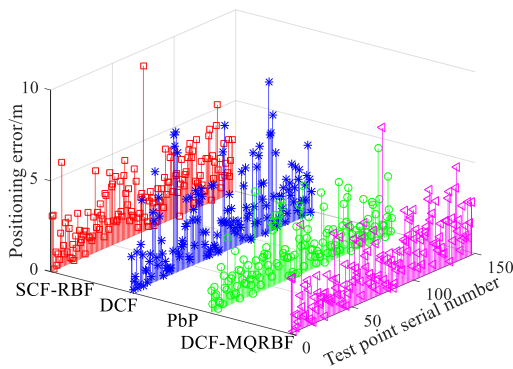


图 5-9: 不同建库方法对应的定位误差

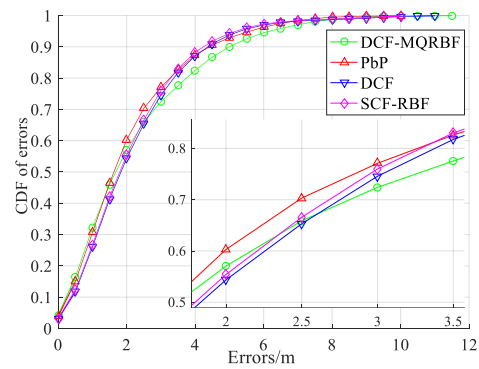


图 5-10: 不同建库方法对应的 CDF 图

5.3.3 时间开销

我们选择 3 个不同的室内环境测试对比不同建库方法的时间开销, 环境 1 为 256 m², 环境 2 为 374 m², 环境 3 为 288 m², 时间开销如图 5-11 所示, 可以发现本文提出的建库方法的建库时间开销远低于其他几种方法的时间开销。

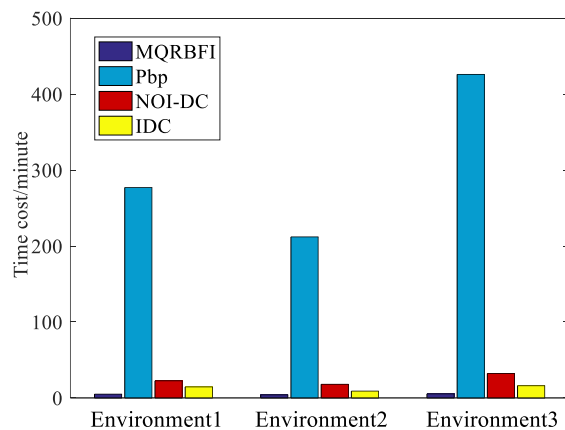


图 5-11: 不同建库方法的建库时间开销

5.4 自适应更新指纹库定位性能测试

在得到初始位置指纹数据库后, 我们对指纹库进行了三次更新, 图 5-12 比

较了指纹库更新前后指纹定位轨迹，可以看出，经指纹库更新后指纹定位效果更贴合于真实轨迹，更新效果较为显著。

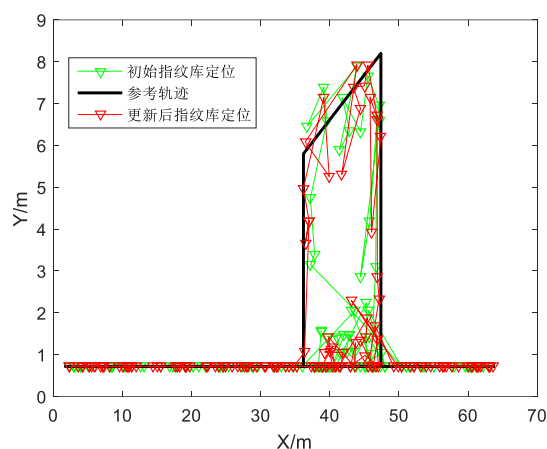


图 5-12: 自适应更新效果图

表 5-3 列出了指纹库不同分位值对应的定位误差，图 5-13 比较了指纹库三次更新前后的 CDF 图。可以看出，经过三次指纹库更新，定位精度有了较大的提升且大的拖尾误差有明显降低。

表 5-3: 不同分位值对应的定位误差（米）

分位值	原始库	一次更新	两次更新	三次更新
50%	≤ 1.4	≤ 1.2	≤ 1.2	≤ 1.1
70%	≤ 1.8	≤ 1.6	≤ 1.5	≤ 1.4
90%	≤ 2.7	≤ 2.6	≤ 2.3	≤ 2.2

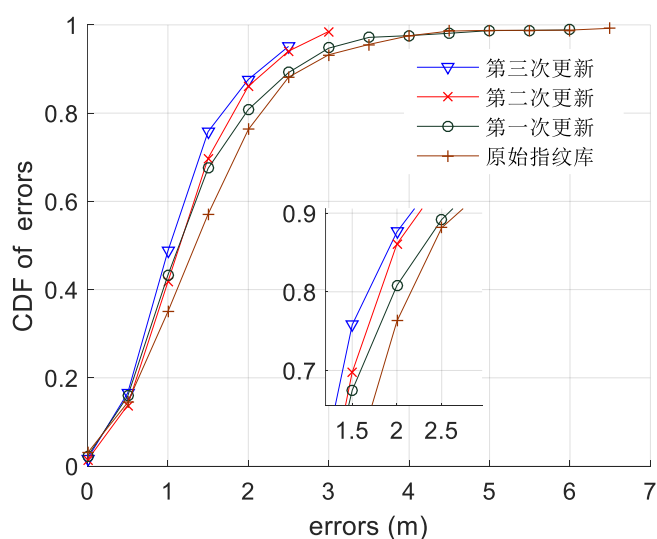


图 5-13: 自适应更新效果图

第6章 总结

本文设计了一种基于 BLE 指纹/MEMS 融合定位的智能室内定位系统, 基于该系统, 可对任意室内场景实现自适应 AP 布局、快速有效地构建高精度的位置指纹数据库和自适应更新指纹库等一系列功能, 满足用户对室内定位服务的需求。大量的实验验证, 通过本文提出的自适应 AP 布局算法能够有效、快速地给出 AP 布局方案, 避免了传统人工布局方式的弊端; 通过动态采集指纹联合 RBF 插值算法的快速指纹库构建方法, 可建立高精度的位置指纹库, 时间开销相比传统建库方法减小 80% 以上; 经过基于 EKF 的 BLE 指纹和 MEMS 融合定位算法, 结合两种定位技术的优点, 实现室内 1.5 米的定位精度; 最后, 利用本文提出的自适应更新指纹库方法, 可使指纹库的定位精度长期保持不下降, 解决了保持定位精度困难的问题。

第 7 章 参考文献

- [1]任晓东, 张柯柯, 李星星, 张小红. BeiDou、Galileo、GLONASS、GPS 多系统融合精密单点[J]. 测绘学报, 2015, 44(12): 1307-1313.
- [2]田增山, 舒月月, 刘仪瑶,等. 改进的蜂窝网室内定位匹配算法[J]. 重庆邮电大学学报(自然科学版), 2017(6):744-750.
- [3]席瑞, 李玉军, 侯孟书. 室内定位方法综述[J]. 计算机科学, 2016, 43(04): 1-6.
- [4]周牧, 王斌, 田增山, 张千坤. 室内 BLE/MEMS 跨楼层融合定位算法[J]. 通信学报, 2017, 38(05): 1-10.
- [5]马鑫迪, 马建峰, 高胜. 室内定位系统中指纹库的优化方法[J]. 西安电子科技大学学报, 2015, 42(06): 81-87.
- [6]Shu Tang, Lin Ma, Yubin Xu. A Novel AP Placement Algorithm Based on User Distribution for Indoor WLAN System[J]. 中国通信, 2016, 13(10): 108-118.
- [7]李燕君, 徐凯锋, 邵剑集. 利用众包更新 Wi-Fi 室内定位指纹库的方法研究[J]. 传感技术学报, 2014, 27(12): 1692-1698.
- [8]A. Saranli and B. Baykal, "Complexity reduction in radial basis function (RBF) networks by using radial B-spline functions," Neurocomputing, vol. 12, no.1, pp. 183-194, 1998.
- [9]Pouderoux J, Gonzato J C, Tobor I, et al. Adaptive hierarchical RBF interpolation for creating smooth digital elevation models[C]// ACM, 2004:232-240.
- [10]Song D, Yang J, Cai Z, et al. Wind estimation with a non-standard extended Kalman filter and its application on maximum power extraction for variable speed wind turbines[J]. Applied Energy, 2017, 190:670-685.
- [11]张梦影, 曾成, 狄素素,等. 多传感器滤波融合的惯性定位算法[J]. 电子技术应用, 2017, 43(10):86-88.
- [12]S. He, B. Ji, and S. C. Gary, "Chameleon: Survey-free updating of a fingerprint database for indoor localization," IEEE Pervasive Comput., vol. 15, no. 4, pp. 66_75, Apr. 2016.
- [13]Liu W, Fu X, Deng Z, et al. Smallest enclosing circle-based fingerprint clustering and modified-WKNN matching algorithm for indoor positioning[C]// International Conference on Indoor Positioning and Indoor Navigation. IEEE, 2016:1-6.
- [14]郑学理, 付敬奇. 基于 PDR 和 RSSI 的室内定位算法研究[J]. 仪器仪表学报, 2015, 36(05): 1177-1185.
- [15]毛勤. 基于 WiFi 位置指纹的室内定位算法的研究与优化[D]. 广东工业大学, 2016.
- [16]J. H. Jung and V. R. Durante. An iterative adaptive multiquadric radial basis

function method for the detection of local jump discontinuities. *Applied Numerical Mathematics*, vol. 59, no. 7, pp. 1449-1466, 2009.

[17]刘志建, 关维国, 华海亮, 孙泽鸿. 基于克里金空间插值的位置指纹数据库建立算法[J]. 计算机应用研究, 2016, 33(10): 3139-3142.

[18]KOO B, LEE S, LEE M, et al. PDR/fingerprinting fusion indoor location tracking using RSS recovery and clustering[C]. *International Conference on Indoor Positioning and Indoor Navigation*. 2014: 699-704.

[19] Zhou M, Wang B, Tian Z, et al. A Case Study of Cross-floor Localization System Using Hybrid Wireless Sensing[C]. *IEEE GLOBECOM*. 2017: 1-6.