

LDE 算法在 WiFi 室内定位中的应用研究

周才发

院（系）：电子与信息工程学院

专 业：通信工程专业

学 号：1090510316

指导教师：马琳

2013 年 7 月

哈爾濱工業大學

畢業設計（論文）

題 目 LDE 算法在 WiFi 室內定位中的應用研究

專 業 通信工程

學 號 1090510316

學 生 周才發

指 導 教 師 馬 琳

答 辯 日 期 2013 年 7 月 5 日

哈尔滨工业大学毕业设计（论文）评语

姓名： 周才发 学号： 1090510316 专业： 通信工程
毕业设计（论文）题目： LDE 算法在 WiFi 室内定位中的应用研究

工作起止日期： 2012 年 11 月 01 日起 2013 年 07 月 04 日止

指导教师对毕业设计（论文）进行情况，完成质量及评分意见：

周才发同学对本科毕业设计的理解深入透彻。在阅读了一定量参考文献的基础上，系统地分析了 PCA 算法、MDS 算法、LLE 算法、ISOMAP 算法、LDA 算法、NPE 算法、SPCA 算法以及 LDE 算法。通过对这些算法的研究，给出了数学推导公式，并对算法进行了全面的分析。论文采用 MATLAB 作为仿真平台实现了对 LDE 算法及其在 WiFi 室内定位中的应用的验证，并且将 LDE 算法与多种其它降维算法进行定位对比。结论表明，该算法能够有效地对数据进行降维处理并且在参数优化的条件下，该算法在 WiFi 室内定位中的定位性能与原始算法的定位性能可比拟。论文结构合理，叙述清晰，分析正确，数据可靠。论文反映出该同学具有较强的表达能力，以及较坚实的专业基础和动手能力。该同学优异的完成了毕业设计论文。

本人已对该论文进行全程指导和严格审查，方案和数据真实、可靠，是该生的原创论文。

指导教师签字： _____ 指导教师职称： 讲师/硕导

评阅人评阅意见：

评阅教师签字： _____ 评阅教师职称： _____

哈尔滨工业大学毕业设计（论文）任务书

姓名：周才发	院（系）：电子与信息工程学院
专业：通信工程	班号：0905103
任务起至日期：2012年11月01日至2013年07月04日	
毕业设计（论文）题目：LDE 算法在 WiFi 室内定位中的应用研究	
立题的目的和意义： WiFi 室内定位中位置指纹数据库（即 Radio Map）随着 AP 数目增加将引起 Radio Map 呈“膨胀”式增长。同时在 WiFi 网络中，由于引入绿色通信管理后，AP 的开关由通信量决定，从而导致终端在在线阶段接收到的 AP 的数目会发生动态变化，造成位置指纹定位算法的出现非对称匹配问题。针对上述问题，本文提出采用 LDE 算法对 Radio Map 进行降维。在减少数据运算量的同时，将在线阶段与离线阶段的数据均降维至低维嵌入维数，解决了非对称匹配问题。	
技术要求与主要内容： 技术要求： 1. 采用 LDE 算法对 Radio Map 降维处理并采用 KNN 算法实现定位； 2. 采用 LDE 算法解决大数据量问题，即对 Radio Map 进行降维，降低在线定位阶段的计算复杂度； 3. 采用 LDE 算法解决非对称匹配问题，即通过将在线与离线阶段的 RSS 均降维至本征维数，达到位置指纹定位算法的匹配定位要求。 主要内容： 1. 学习流形及流形学习的主要概念、方法及原理，掌握其基本理论、算法及应用； 2. 对 LDE 算法的数学理论进行分析并给出推导证明； 3. 采用 MATLAB 软件实现 LDE 算法并将其应用于对 Radio Map 的降维并实现 WiFi 室内定位； 4. 选取不同的降维算法，如 PCA、LDA、SPCA、NPE 等算法应用于 WiFi 室内定位，并与 LDE 算法进行性能比较。	

进度安排：

1. 2012.11.01-2012.11.30：了解本课题的研究现状、确定毕业设计任务，准备开题；
2. 2012.12.01-2013.01.30：查阅文献，学习有关流形及流形学习算法的基本理论并完成 LDE 算法的理论推导，开始撰写中期检查报告；
3. 2013.02.01-2013.05.08：采用 MATLAB 软件及 GUI 界面编写 LDE 算法代码并应用于 Radio Map 的降维处理，实现定位，并完成中期检查报告；
4. 2013.05.09-2013.06.15：选取多种不同的降维算法应用于 WiFi 定位，并与 LDE 算法的性能进行比较，并采用 MATLAB 软件实现其算法，同时开始撰写毕业论文；
5. 2013.06.16-2013.06.30：撰写毕业论文并准备毕业答辩。

同组设计者及分工：无。

指导教师签字_____

年 月 日

教研室主任意见：

教研室主任签字_____

年 月 日

摘 要

随着可移动计算终端和无线技术的发展，定位导航系统越来越受到研究人员的关注，而且定位导航的应用也越来越广泛。在室外，可以利用卫星信号进行定位导航；在室内，由于可见卫星数目不足，卫星定位导航方法并不适用于室内。近年来，多种基于不同方式的室内定位系统被提出，包括基于红外线、超声波、以及射频等信号的室内定位系统。基于 WLAN 的室内定位系统属于一种典型的基于射频的室内定位系统。基于 WLAN 的室内定位系统利用已有的无线局域网的基础设施来完成室内定位系统的构建，不需要增加硬件设备，因此受到许多研究人员的重视。

目前基于 WLAN 的室内定位系统面临两方面的挑战。一方面，由于 WLAN 的首要目标是为用户提供有效地无线网络接入，因此接入点数目一般都会出现冗余。这一种冗余会增加基于 WLAN 室内定位系统的 Radio Map 及在线定位算法的复杂度。另一方面，引入绿色通信的 WLAN 室内定位系统中，在线定位阶段的可见接入点数目动态变化，导致在线匹配定位算法无法有效地进行定位。这两方面的问题都是由于接入点数目的变化造成，通常分别称为高维位置指纹和非对称匹配定位。

针对目前基于 WLAN 的室内定位系统中存在的高维位置指纹及非对称匹配定位的问题，采用局部鉴别嵌入（LDE）算法对 Radio Map 降维，并且提取特征矩阵。通过对 Radio Map 降维，达到减少数据冗余的目的，同时利用 Radio Map 的特征矩阵解决在线阶段的非对称匹配定位问题。局部鉴别嵌入是一种监督型非线性流形学习方法，需要对高维数据进行类别标记。考虑到目前基于 WLAN 的室内定位区域内的参考点的数目大，不适合人为地对所有的参考点进行分类，因此采用基于核函数的模糊 c 均值聚类（KFCM）方法对 Radio Map 进行类标记，从而简化 LDE 算法的实现过程，提高 LDE 算法的运算效率。

本文分析了流形及流形学习的基本数学概念及理论，并且对 PCA、SPCA、MDS、ISOMAP、LLE、LDA、NPE 等七种常见的降维算法进行基本理论分析；详细地推导了 LDE 算法的理论过程、给出了 LDE 算法流程及其 MATLAB 实现方案；并对聚类方法的引入及聚类、模糊聚类、基于核函数的模糊聚类方法等基本理论进行了分析，以便采用 KFCM 算法对位置指纹数据进行类标记。通过对基于 LDE 算法的 WiFi 室内定位（LDE-KNN）方法的所有 6 个参数进行 MATLAB 仿真分析，实现 LDE-KNN 算法的优化。对 A1 子区域的仿真结果表明，采用 LDE-KNN

方法与 KNN 方法在 2 米内定位精度均可达到 70%左右，且优化后的 LDE-KNN 方法的复杂度仅为 KNN 方法的三分之一。通过与 PCA、SPCA、LDA、NPE 等四种降维方法进行性能对比可以得出：在同等参数条件下，LDE-KNN 算法的性能优于上述四种方法。

关键词：无线局域网；室内定位；维数约简；特征提取；局部鉴别嵌入；

Abstract

With the development of portable computation ultimate and wireless technology, position and navigation systems are attracted more and more attention of researchers and its applications are more and more comprehensive. In outdoor environment, using satellite to position and navigation, but satellite positioning navigation systems cannot adapt in the lacking visible satellites conditions in indoor environment. Recent years, several indoor positioning systems based on different standards were proposed, including infrared-based, ultrasonic-based and radio frequency (RF) –based. And indoor positioning system based on WLAN is a typical RF-based indoor positioning approach. Indoor positioning system based on WLAN is a hotspot field for its realization is depended on the available wireless local area networks (WLAN) without additive infrastructure.

Indoor positioning system based on WLAN faces two challenges. On the one hand, since the prime purpose of WLAN is offering wireless access of its users, therefore, the number of access points (APs) always is redundancy. And this kind of redundancy increase Radio Map of indoor positioning systems and the complexity of its online position algorithm. On the other hand, as the prevalent of green WLAN, the visible AP at online positioning stage is dynamic and this causes the failure of online positioning algorithm. Both of these two problems are caused by the number of APs. Generally, the former is called hyper-dimensional Radio Map, and latter is called asymmetric matching.

To solve the two tickle problems caused by the dynamic of visible APs in indoor positioning systems, proposing local discriminant embedding (LDE) algorithm to reduce the dimensionality of Radio Map and extract the feature matrix. Solving the problem of hyper-dimensional Radio Map by dimensionality reduction and using its feature matrix to realize asymmetric matching at the online positioning stage, the both problems are tackled. And LDE is supervised nonlinear manifold learning algorithm and the classification of hyper-dimensional Radio Map is used in the process of dimensionality reduction. Considering about the large number of reference points in the interested region of indoor positioning system and the classification could not be achieved by artificial, Kernel based c-means cluster (KFCM) algorithm is proposed to fulfill this purpose and by applying this method, the efficiency of LDE algorithm could be improved.

In this paper, basic mathematical analysis of manifold and manifold learning are analyzed and several dimensionality reduction algorithms are analyzed, including Principal Component Analysis (PCA), Simple PCA (SPCA), Multi-Dimensional Scaling

(MDS), Isometric Mapping (ISOMAP), Local Linear Embedding (LLE), Linear Discriminant Analysis (LDA) and Neighbor Preserving Embedding (NPE). And the details of extrapolation process of LDE algorithm, procedure of LDE and its realization based on MATLAB are proposed. To label the class of hyper-dimensional Radio Map, fuzzy cluster and KFCM are analyzed in this paper. Through simulating the six key parameters of online matching algorithm based on LDE (LDE-KNN) by MATLAB, the optimal parameter of LDE-KNN could be obtained. Simulation of A1 sub-region shows that the performance of proposed LDE-KNN approach is comparable to old online matching algorithm under the circumstance that the complexity of LDE-KNN is only one-third to the old one. And its positioning probability within 2 meters of LDE-KNN is 70 percent. And the simulation shows that the performance of LDE-KNN outperforms the other four dimensionality reduction algorithms, PCA, SPCA, LDA and NPE, using the same parameters.

Keywords: WiFi, indoor position, dimensionality reduction, feature extraction, local discriminant embedding

目 录

摘 要	I
ABSTRACT	III
第 1 章 绪 论	1
1.1 课题来源及其背景	1
1.2 研究目的与意义	3
1.3 国内外研究现状	4
1.4 本文的主要研究内容	5
第 2 章 流形学习及常见的降维算法	8
2.1 流形基本概念	8
2.2 常见的降维方法	9
2.2.1 主成分分析法	9
2.2.2 多维尺度变换法	10
2.2.3 等距映射法	10
2.2.4 局部线性嵌入	11
2.2.5 简化主成分分析法	12
2.2.6 线性鉴别分析	13
2.2.7 邻域保持嵌入	14
2.3 降维算法的比较分析	14
2.4 本章小结	15
第 3 章 LDE 算法的理论分析	16
3.1 LDE 算法概述	16
3.2 LDE 算法的理论推导	17
3.3 LDE 算法的 MATLAB 实现流程	19
3.4 本章小结	20
第 4 章 基于核函数的模糊 C 均值算法	21

4.1 聚类算法的引入	21
4.2 聚类的基本思想	21
4.3 模糊 c 均值聚类	22
4.4 基于核函数的模糊 c 均值聚类	23
4.5 本章小结	25
第 5 章 基于 LDE 算法的 RADIO MAP 降维处理	26
5.1 LDE 算法在 WiFi 室内定位应用方案及其性能对比算法	26
5.1.1 实验环境	26
5.1.2 应用方案	26
5.2.2 对比算法选择	27
5.2 基于 LDE 算法的 WiFi 室内定位方法的性能分析	28
5.2.1 基于 LDE 算法的 WiFi 室内定位方法的参数分析	28
5.2.2 基于 LDE 算法的 WiFi 室内定位方法性能分析	32
5.2.3 基于 LDE 算法的 WiFi 室内定位方法的性能对比分析	35
5.3 本章小结	36
结 论	37
参考文献	39
攻读学士学位期间发表的论文及其它成果	43
哈尔滨工业大学本科毕业设计（论文）原创性声明	44
致 谢	45
附录 一	46
附录 二	56

第 1 章 绪 论

1.1 课题来源及其背景

本课题来源于国家自然科学基金（项目编号：61101122）：《基于绿色 AP 的 WLAN 室内定位算法研究》。

随着可移动计算终端和无线技术的发展，定位导航系统越来越受到研究人员的关注而且定位导航的应用也越来越广泛。在移动终端上集成用户位置信息可以为用户提供许多基于位置的服务（Location Based Service, LBS），如个人安全，旅游及物流^[1-2]。在室外，可以利用卫星信号进行定位导航，如全球卫星定位导航系统（Global Positioning Systems, GPS）、北斗卫星导航系统等。基于卫星的定位导航系统在室外定位导航中占据了主导地位，而且其发展状况日趋完善。但当用户携带移动终端进入室内时，由于可见的卫星数量少，无法提供基于卫星的定位导航服务。因此，需要室内定位的定位导航系统为室内移动终端提供位置信息的服务。

从目前发展情况来看，已有多种基于不同方式的室内定位系统。基于红外信号的Active Badge系统^[3-4]、基于超声波的Active Bat系统^[3-4]、基于视频图像^[5]、基于射频（Radio Frequency, RF）信号^[6-15]。在基于RF的室内定位系统的中典型代表有：基于RFID的LANDMARC系统以及众多基于WLAN（Wireless Local Area Network）室内定位系统。基于WLAN的室内定位系统的优越性在于可以利用已有的基础设施来完成室内定位系统的构建，并不需要增加硬件设备；对于移动终端来说，目前很多智能手机上都有WiFi(Wireless Fidelity)接入模块，可以方便地接收不同接入点（Access Point, AP）发送的信号强度值。正是这一优点，使得基于WLAN的室内定位系统越来越流行。

基于WLAN的室内定位系统在系统构建及终端接入上的优点，却给定位算法的实现带来了一定的挑战。这个挑战是源于目前的WLAN网络的架构与设置是随机的，而且在多数场所AP的数目是处于冗余状态，也就是在某一热点区域内，AP布置的数目大大地超过的用户本身需求，或者说远大于定位系统本身的需要。美国常青藤院校达特茅斯学院在其校区面积为4km×5km内布置了近500个AP^[16]，而微软总部更在其公司所有在科技园内布置了近5000个AP^[17-18]。在采用位置指纹匹配定位的WLAN室内定位系统中，随着AP数目增加，其位置指纹，即Radio Map的数据量及维数也会随之增加。因此，当AP数目增加到5000时，Radio Map也就变成了一个高维数据库。这给定位过程带来了高维位置指纹问题。

海量数据多数出现在图像处理、机器视觉及模式识别等领域。对于海量数据处理的方法主要有两大类^[19]，一类是粗分集，另一类是维数约简，即降维。现今应用较为广泛的方法是维数约简。目前已提出的降维方法多达三十多种^[20]，总体上可以分为线性方法和非线性方法。典型的线性方法有主成分分析法^[21]（Principal Component Analysis, PCA）、线性鉴别分析^[22~24]（Linear Discriminant Analysis, LDA）。而非线性降维方法以流形学习方法为主。从目前流形学习算法的研究情况来看，包括等距映射法^[25]（Isometric Mapping, ISOMAP）、局部线性嵌入法^[26]（Local Linear Embedding, LLE）局部鉴别嵌入（Local Discriminant Embedding, LDE）算法等。LDE算法与LLE、ISOMAP等算法相比，其优势在于LDE除了可以对海量数据进行维数约简之外，还可以根据海量的数据的类信息提取数据特征。通过降维算法，对海量数据进行约简，同时提取出Radio Map中的特征用于WiFi室内定位。这一方法可以解决由于高密度、AP大量冗余的环境下的位置指纹匹配定位算法存在的问题。

另一方面，近年来，绿色通信的概念越来越受到重视。2009年，美国加利福尼亚大学与Intel联合实验实验室在WLAN的基础上提出了绿色WLAN的概念^[27]。绿色WLAN运用能效管理技术，使传统的AP根据用户通信需求开启或关闭成为绿色AP，从而保证在WLAN通信服务能力基本不变的前提下，大幅降低AP的能耗和射频辐射。在引入绿色通信概念的WLAN网络中，处于开启状态的AP的数目是动态变化的。这种动态变化目标是在满足用户通信要求条件下尽可能开启较少的AP，以达到节能的目的。但是这种动态变化使位置指纹匹配定位算法产生另一个问题——非对称匹配。由于位置指纹定位算法的原理是通过各个AP间的信号空间的特征达到预测热点室内定位区域的坐标的目标。因此，当AP开启的数目动态变化时，位置指纹定位算法就会失效，无法实现定位。

解决非对称匹配问题同样可以采用维数约简方法。这是因为在Radio Map中，每一维实际代表的一个不同的AP。通过维数约简，将不同的AP数目均可以等效成相同数目维数，而维数的概念，可以认为是将原有的AP等效成为虚拟AP的概念，从而达到位置指纹匹配定位算法的定位要求。从基于WLAN的室内定位系统的指纹定位算法的要求出发，分析AP数目的变化及绿色WLAN的前提下，提出采用维数约简算法对高维位置指纹进行数据压缩，同时实现非对称条件下的指纹数据库的匹配定位，达到解决WiFi室内定位中的高维位置及非对称匹配问题。

1.2 研究目的与意义

针对 Wireless Fidelity (WiFi) 室内定位中 Radio Map, 随着 AP 的数目的增加会引起 Radio Map 呈“膨胀”式增长, 对 WiFi 室内定位实现造成困难。在引入绿色 AP 的 WiFi 室内定位网络中, 采用动态地按需分配方案来对 AP 的开关进行控制。其按需分配的含义是在满足用户通信量及通信质量 (Quality of Service, QoS) 前提下, 开启尽可能少的 AP。而基于 WiFi 室内定位网络是 WiFi 网络的一个附属功能, 而不是无线网络的首要目标。由于 WiFi 室内定位方案 (图 1-1) 的实现中, 采用 K-Nearest Neighbors (KNN) 指纹匹配定位算法对离线及在线数据进行匹配, 其关键是离线阶段与在线阶段的不同位置上的接收到的 AP 的信号强度 (Received Signal Strength, RSS) 的信号空间匹配。而引入绿色 AP 的 WiFi 网络的 AP 的开关数目是随着网络通信状态与 QoS 要求动态变化的, 导致 KNN 算法的指纹定位无法实现非对称匹配。针对目前室内定位中对于多 AP 条件下实现定位存在的高维位置指纹及非对称匹配问题, 提出采用流形学习算法对 Radio Map 进行降维, 从而减少数据量, 实现对 WiFi 室内定位算法的改进。

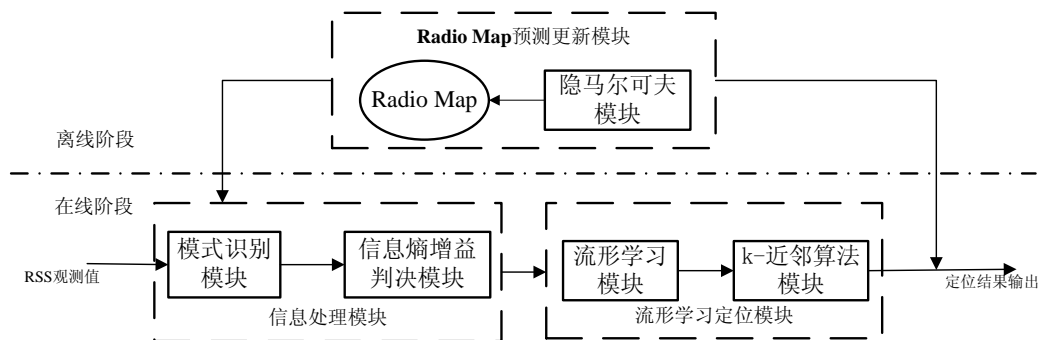


图 1-1 基于绿色 AP 的 WiFi 室内定位算法的实现流程

从 2000 年 Science 发表的有关 ISOMAP 算法及 LLE 算法的论文起, 流形学习算法的研究及其实现受到了许多的研究人员的重视。LDE 算法提出的时间较晚, 从目前研究情况来看, 并没成为主流的流形学习算法。但从 LDE 算法及其功能实现来看, 有一定的优势。针对目前 Radio Map 的建立及其应用的室内环境 (图 1-2 哈尔滨工业大学科学园 2A 栋 12 层平面图示意) 来看, 采用一定的算法^[28]对走廊进行了分区。在 LDE 算法实现流程^[29~30]中对分区进行了进一步分析, 采用聚类算法对数据集进行聚类, 再计算分类结果的类间散度及类内散度。LDE 算法的优化目标是类间散度的最大化及类内散度的最小化。从而进一步对 Radio Map 的特征进行区分, 即 LDE 算法中提出的“鉴别”的思想。LDE 算法比其它已有流形学习算法相比, 其稳定性及对数据集的特征的区分度更强。其主要的缺陷在于其实现的复杂度较大。但从课题的应用背景来看, 在 WiFi 室内定位的离线阶段, 可以采用高性能的

计算设备来看实现对Radio Map的LDE算法的处理，所以LDE算法在目前的条件可以实现，并且有一定的优势。

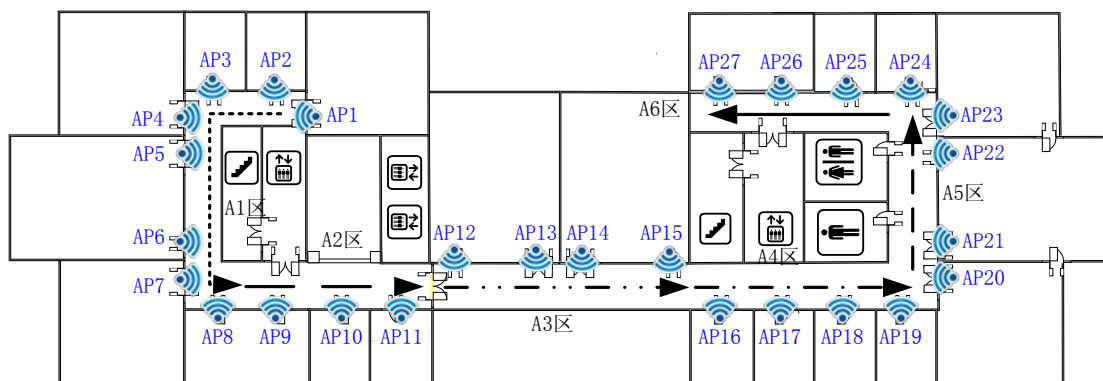


图 1-2 哈尔滨工业大学科学园 2A 栋 12 层平面示意图

实现了将 LDE 算法应用于 Radio Map 的降维处理后，将采用目前已有的 WiFi 室内定位算法实现对未知位置的坐标预测。从 LDE 算法及其与目前 WiFi 室内定位系统的融合来看，LDE 算法相比于其它经典的流形算法有一定的优势，并且可以实现。因此，研究 LDE 算法在 WiFi 室内定位的应用有关重要意义。

1.3 国内外研究现状

随着科学技术的发展，通信工程、机器学习、数据挖掘、计算机视觉和其他方面的科研需要从一个大的数据集上获得一个分类器或特征提取函数。现代数据集通常包含大量的数据，且包含很多的特征。极其复杂的特征会增加算法的负担，使寻找全局最优解的过程变得更加复杂。为了减轻处理高维数据的负担，人们开发了大量的算法提取数据集的特征，从而实现降低数据的维度。

数据降维，除简化数据使其能够高效处理外，还有很多其他的用途。最明显的应用就是数据可视化。假设数据处于 100 维的空间里，对于数据的直观表示没有确定的认知。但是，如果能找到一个有意义的二维或三维表现形式，那就能清楚地感知数据的直观印象。虽然这些看起来可能是一个琐碎的问题，但是由于很多统计学的和机器学习算法对于最优解的保证很差，数据降维的可视化应用，也就是给人实际看到数据和算法输出的能力，具有很高的应用价值。

不同降维算法的假设不同，其应用领域也各有不同。本文研究的流形学习算法是近年来提出的方法。特别地，一个指定的高维数据可能包含很多特征，这些特征都在描述同一个暗含的原因，所以这些特征是紧密相连的。这种现象很常见。例如当从各个角度对同一个物体同时拍照时，得到的数据就含有重叠的信息。如果能得到这些数据的一些简化的不重叠的表达，这些表达的特征可以被通过主导

这些数据的隐含参量识别，这种方法将会是非常有益处的。假设数据处于一个被嵌入在高维空间的低维流形上，低维空间反映了潜在的影响参数，而高维空间是特征空间。尝试解释数据集的流行结构的过程，称为流形学习。

在国外，流形学习的可以追溯到 1984 年Hastie在技术报告中提出的有关于主曲线、主曲面等概念^[31]。之后的许多研究中也发现高维数据存在一定的低维流形特征。如Nayar等人发现数字图像具有内在的低维特性^[32]，Bregler等人也发现了语音信号中的高维特性^[33]。在 1984 年，Bregler提出了“流形学习”(Manifold Learning)并应用于多个不同的领域^[34]。

2000 年，Seung和Daniel发表论文，认为人的感知以流形方式存在^[35]，给出了流形学习的认知基础。同一年发表在《Science》上的两篇文章中提出了两种经典的流形学习算法ISOMAP和LLE。之后大量的流形学习算法被提出包括：Belkin等提出的拉普拉斯特征映射(Laplacian Eigenmap, LE)^[36]、Weinberger等给出的最大差异展开(Maximum Variance Unfolding, MVU)^[37]、Hwann-Tzong Chen等在 2005 年提出LDE算法。

国内学者大流形学习算法研究方面也取得了丰硕的成果。在经典流形学习方法研究方面，张振跃等给出子局部切空间对齐(Local Tangent Space Alignment, LTSA)^[38]算法，并将其应用于图像识别领域、Yang于提出了局部多维尺度分析(Local Multi-Dimensional Scaling, LMDS)^[39]以及林通等人于 2008 年提出的黎曼流形学习(Riemannian Manifold Learning, RML)^[40]等算法。在流形学习的应用方面，国内代表性的研究工作主要包括：流形学习的可视化方法^[41]；流形学习算法应用于人脸识别^[42]及人体动作识别^[43]。同时，詹德川及杨剑等人提出了将流形学习与回归算法相结合^[44~45]。

综述国内外研究现状，可以得出：国内外学者对于流形学习算法做出了很多深刻的研究，提出了大量的流形学习算法，但是每种流形学习算法应用的领域各有不同。现今流形学习算法主要应用于图像处理、人脸识别及网络数据信息处理。总的来说，流形学习算法主要应用于对海量数据降维处理，其本质是对数据进行压缩，同时能保持数据本身的某些特征。国内外研究成果对于本文的研究及流形学习算法在 WiFi 室内定位中的应用提供一定参考。

1.4 本文的主要研究内容

就目前的 WLAN 网络而言，存在以下两方面的问题。一是 WLAN 网络中 AP 的布置密度过高，AP 的数量太大。在基于 WLAN 的位置指纹匹配定位算法中，随着 AP 数目的增加，Radio Map 的维数及数据量都会增加，导致在实时定位速度

不能达到相应的要求。由于 AP 数目增加引起的 WLAN 室内定位算法的大数据量问题通常称为高维位置指纹问题。另一个问题是，在引入绿色通信控制的 WLAN 中，实时可观测到的 AP 的数目是动态变化的，导致采用位置指纹匹配定位的 WLAN 室内定位系统无法工作，这一问题通常称为非对称匹配问题。

针对目前 WLAN 室内定位网络中存在的以上两个问题，提出采用 LDE 算法对位置指纹数据库进行维数约简，从而对 Radio Map 的高维位置指纹进行有效地压缩，从而使得实时定位时间减少（以在线定位复杂度衡量）；同时通过将位置指纹数据与实时接收到的高维数据均降维至相同维数，使得实时虚拟 AP 数目与指纹数据库的虚拟 AP 的数目相等，从而达到位置指纹匹配定位的要求，解决非对称匹配问题，实现 WiFi 室内定位。

本文主要研究 LDE 算法在 WiFi 室内定位系统中的应用方案并验证其在目前室内定位系统中应用的有效性。在对 LDE 算法进行深入讨论之前，对流形及流形学习涉及的重要的数学概念、理论进行分析，并对常见的降维算法的基本理论进行讨论分析，如 PCA、简化主成分分析法^[46]（Simple PCA, SPCA）、LDA、多尺度变换^[47~48]（Multi-Dimensional Scaling, MDS）、LLE、ISOMAP、邻域保持嵌入（Neighbor Preserving Embedding, NPE）等。有了对 LDE 的理论分析后，采用 MATLAB 软件编程实现 LDE 算法并采用 GUI 界面对算法的定位过程进行仿真。同时，分析影响 LDE 算法的各个参数以便对算法的性能进行初步优化。为了比较 LDE 算法与其它降维算法的性能，在本文中选择了 PCA、SPCA、LDA、NPE 等四种算法应用 WiFi 室内定位中，并与 LDE 算法的仿真结果进行比较。

绪论中主要介绍了本课题的来源及背景。在自然科学基金项目《基于绿色 AP 的 WLAN 室内定位算法研究》的背景下，提出采用 LDE 算法对 Radio Map 进行降维处理，其优点在于在对高维位置指纹进行压缩的同时，也可以提取出高维位置指纹中的特征及其特征变换矩阵，可以有效地用于基于 WLAN 的室内定位系统。然后对本课题研究的目的与意义及国内外研究现状进行了概述。

第 2 章主要介绍了有关流形及学习的一些基本概念及相关理论，给出了有关流形及流形学习的定义。对 PCA、SPCA、LDA、MDS、LLE、ISOMAP、NPE 等七种常见的降维算法的基本理论及算法流程进行简要分析，为后续的研究打下基础。

第 3 章对 LDE 算法的理论推导进行了详细的介绍。主要包括 LDE 算法的流程、LDE 算法目标函数及其广义特征分解表达式的推导。同时为了提高算法的稳定性，引入了正则化因子，从而改善了算法的性能，并给出了 LDE 算法在 MATLAB 中的实现流程，为后续进一步的仿真分析奠定基础。

第 4 章引入了基于核函数的模糊 c 均值聚类（Kernel based Fuzzy c -means），

KFCM) 算法对分区的位置指纹数据 (Radio Map) 进行类标记, 以便给 LDE 算法提供有效的类信息。主要给出了 c 均值 (c -Means, CM) 聚类算法的基本思想及其通用的算法流程, 并通过引入隶属度的描述, 对模糊 c 均值聚类 (Fuzzy c -Means, FCM) 算法基本理论及其算法流程进行分析。最后, 通过平行推广的方式, 对 KFCM 算法的理论及其在希尔伯特空间中的实现方式, 为 LDE 算法的输入数据的类标记提供有效的算法。

第 5 章给出了 LDE 算法应用于 WiFi 室内定位系统的方案及与 LDE 进行性能对比的算法的选择指标及其的仿真结果。主要内容包括实验环境概述、总体应用方案及 LDE 对比算法选择指标、给出 LDE 算法在定位区域内不同分区的仿真结果, 同时对 LDE 算法中涉及的多个参数进行分析, 以便基于 LDE 算法的 WLAN 室内定位算法进行初步优化。在本章的最后给出了 PCA、SPCA、LDA、NPE 等四种不同的降维算法应用于 WiFi 室内定位系统中的仿真, 并与 LDE 算法的定位结果进行对比。

第 2 章 流形学习及常见的降维算法

流形是拓扑学的一个子范畴。从严格的数学定义来讲，流形及光滑的条件十分严苛。本章对流形及流形学习的基本理论及概念进行简要研究。在此基础上，对 PCA、MDS、ISOMAP、LLE、SPCA、LDA 及 NPE 等七种常见的降维算法的基本理论及算法流程作概要分析，从而为后续的研究奠定基础。

2.1 流形基本概念

拓扑学最基本的研究对象是拓扑空间。拓扑空间是一个集合且包含一定的拓扑结构。 X 、 Y 是 2 个拓扑空间，满足： $f: X \rightarrow Y$ 是一个连续映射， f 有逆映射，而且逆映射是连续的，则 X 与 Y 同胚^[49]。

拓扑空间 M 在满足以下条件时，称 M 为 m 维流形，即：

(1) M 为豪斯朵夫 (Hausdorff) 空间。豪斯朵夫空间是数学拓扑学中的一个分离空间，满足分离定理：对于拓扑空间 M 中任意 2 个不同的点 x 和 y ，存在 x 的邻域 U 和 y 的邻域 V ，满足 $U \cap V = \Phi$ 。

(2) 任意一点 $p \in M$ ，存在包含的 m 维坐标邻域 (U, Φ) ，坐标邻域是拓扑空间中的开集与其在欧几里得空间中的映射 Φ 有序对。根据流形的上述定义及特征，对不同特殊的流形给出以下定义：

微分流形： d 维 C^k 流形存在同胚映射对 (M, Λ) ，其中 M 为 d 维流形， $\Lambda = \{(U_\alpha, \varphi_\alpha)\}_{\alpha \in A}$ 为一 C^k 的微分流形，其条件为：

(1) (局部欧式性) $\{U_\alpha : \alpha \in A\}$ 构成 M 的开覆盖， $\varphi_\alpha : U_\alpha \rightarrow \varphi_\alpha(U_\alpha) \subset R^d$ 为同胚映射；

(2) (C^k 相容性) 若 $U_\alpha \cap U_\beta \neq \Phi$ ，则双射

$$\varphi_\alpha \varphi_\beta^{-1} : \varphi_\beta(U_\alpha \cap U_\beta) \rightarrow \varphi_\alpha(U_\alpha \cap U_\beta) \quad (2-1)$$

及其逆映射均 k 次可微的，则称 $(U_\alpha, \varphi_\alpha)$ 和 (U_β, φ_β) 是相容；

(3) (最大性) 若 U 为 M 中的开集， $\varphi : U \rightarrow \varphi(U) \subset R^d$ 与 Λ 中的每个 $(U_\alpha, \varphi_\alpha)$ 都相容，则 $(U, \varphi) \in \Lambda$ 。

从上述条件可以得出，微分流形满足连续、光滑及其几何空间不存在跳跃等基本特征。

切向量和切空间：光滑流形 M 在点 x 的切向量就是一个映射 $v_x : C^\infty(M) (\text{光滑映射}) \rightarrow R$ ，且对 $\forall g, h \in C^\infty(M), a, b \in R$ 满足：

$$v_x(ag + bh) = av_x(g) + bv_x(h) \quad (2-2)$$

$$v_x(gh) = v_x(g)h(x) + g(x)v_x(h) \quad (2-3)$$

假设 (U, φ) 为 x 点的一个坐标系，则映射

$$\left(\frac{\partial}{\partial x_i}\right)_x : g \rightarrow \left(\frac{\partial g}{\partial x_i}\right)_x \equiv \left(\frac{\partial(g\varphi^{-1})}{\partial x_i}\right)(\varphi(x)), \quad g \in C^\infty(M) \quad (2-4)$$

式(2-4)为 x 点的一个切向量。点 x 的切向量构成的集合称之为 M 在点 x 的切空间。

黎曼流形：若光滑流形 M 的每个切空间 $T_x(M)$ 中都给定了内积，则称 M 为黎曼流形。

在流形学习算法中涉及对于空间距离的描述，以下对弧长、欧氏距离及测地距离做出如下数学定义：

弧长：设 $S(t), a \leq t \leq b$ 是流形 M 中的一条曲线，在 S 上每点的切向量记为 v_t ，则可以定义曲线 S 的弧长 $l(C)$ 为 $l(S) = \int_a^b \|v_t\| dt$ 。

欧式距离：也称欧几里得距离，它是一个通常采用的距离定义，表示 m 维欧氏空间中两点的真实距离。在一个 m 维欧式空间中，表示两个点的欧式距离的数学

$$\text{表达式为 } d = \sqrt{\sum_{i=1}^m (x_i - y_i)^2}。$$

测地距离：设 p, q 是流形 M 中任何两点，则这两点间的测地距离 $d_M(p, q)$ 为 M 中连接 p, q 的所有高维点的距离的和的最小值。

通过对流形及其基本理论的分析，可以给出流形学习的定义：假设数据是低维流形在高维空间的采样，通过高维数据的特征学习来发现其低维嵌入流形。其本质过程是通过事物的外在表现得到相应的内在结构。

用数学语言可以这样描述：高维数据集 $X = \{x_1, x_2, x_3, \dots, x_n\}$ 是高维的光滑流形，假设其本征低维空间为 $Y \subset R^d$ 且 $f : Y \rightarrow R^D$ 是一个光滑的嵌套，一般情况 $D \gg d$ 。流形学习的目标是基于 R^D 上的一个给定的数据集 $\{x_i\}$ 去恢复 Y 及 f 。

2.2 常见的降维方法

2.2.1 主成分分析法

主成分分析法^[21]（PCA）是一种线性降维算法，其衡量信息的方式是方差的大小。其算法的目标是最大化低维数据的方差，从而最大化其信息。

主成分分析法的算法流程如下所述：

(1) 对于高维数据集 $X = \{x_1, x_2, x_3, \dots, x_n\}$ ，计算其数据中心，即其平均值，记为 \bar{X} ，对数据中心化为 $X - \bar{X}$ ；

(2) 计算中心化数据的协方差，即：

$$C_{ij} = (x_i - \bar{x})(x_j - \bar{x}) \quad (2-5)$$

上式中 C_{ij} 表示协方差。

(3) 对式(2-6)做广义特征分解，得到前 d 个最大的非零特征值对应的特征向量为变换矩阵。其降维结果即为 $y_i = \alpha^T x_i$ 。

$$C\alpha = \lambda\alpha \quad (2-6)$$

上式中 α 、 λ 分别为特征向量及对应的特征值。

2.2.2 多维尺度变换法

多维尺度变换法^[47~48]（MDS）的主要思想是：利用两点间的欧式距离（两个高维数据点的 2 范数），构造关系矩阵，再对关系矩阵进行特征分解，进而得到高维数据点在低维的坐标。

现假设给定的高维数据点 $X = \{x_1, x_2, \dots, x_n\}$ ， $x_i \in R^D$ ，则两点间的欧式距离可以表示为 $\Delta_{ij} = \|x_i - x_j\|$ 。

MDS 算法的基本步骤可以表示为：

(1) 通过计算观测数据点中任意两点的欧式距离，得到欧式距离平方的矩阵 $A = (\Delta_{ij}^2)_{n \times n}$ 。

(2) 对矩阵 A 进行双中心化计算，得到中心化矩阵 B 。

(3) 通过对矩阵 B 的奇异值分解，设 B 的最大的 d 个特征值 $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ ，其所对应的特征向量 $U = \{u_1, u_2, \dots, u_n\}^T$ ，最后得到 X 的目标低维数据 $Y = \sqrt{\Lambda}U$ 。

2.2.3 等距映射法

等距映射法^[25]（ISOMAP）算法是在 MDS 算法基础上，力求保持数据点的内部几何性质，即保持两点间的测地距离。ISOMAP 算法是一种非线性的学习方法，它适用于学习内部平坦的低维流形，在算法中需要确定两个参数，一个是邻域的大小，一个是降维的维数。

ISOMAP 算法与 MDS 算法的主要区别就是，MDS 用欧式距离来构成关系矩阵，而 ISOMAP 算法是利用两点间的测地距离来代替 MDS 算法中的欧式距离，

从而构造关系矩阵，得到高维流形中的低维数据关系。

ISOMAP 算法的基本步骤：

(1) 首先建立邻接图，有两种方法确定邻域： k 邻域法和 ε 邻域法，其中 k 邻域法是取与 x_i 点欧式距离最短的 k 个点作为 x_i 点的邻近点；而 ε 邻域法是以 x_i 为球心 ε 为半径的球面内的点为 x_i 的邻近点。

(2) 然后计算测地线距离。如果点 x_j 在 x_i 的邻域内，则用它们之间的欧式距离 Δ_{ij} 表示它们的测地距离 $d_M(x_i, x_j)$ 。如果点 x_j 不在 x_i 的邻域内，则利用 Dijkstra 算法（或 Floyd 算法）计算它们之间的最短路径来表示它们的测地距离 $d_M(x_i, x_j)$ ，进而构造出关系矩阵 D 。

(3) 下面的步骤就和 MDS 算法一致了，对 D 进行中心化，再奇异化分解，最后得到 X 的目标低维数据 $Y = \sqrt{\Lambda}U$ 。

ISOMAP 是一种全局型流形学习方法，其嵌入结果可以最大限度地保持高维数据与低维流形的等距特性。但 ISOMAP 计算量大，容易出现空洞。

2.2.4 局部线性嵌入

局部线性嵌入^[26]（Local Linear Embedding, LLE）降维算法的目标函数为：

$$\min \varepsilon(y) = \sum_{i=1}^m \left\| y_i - \sum_{j=1}^k W_{ij} y_{i,j} \right\|^2 \quad (2-7)$$

式(2-7)中， y_i 表示降维结果， W_{ij} 表示权重。LLE 算法流程如下所述：

- (1) 邻域的确定和局部邻域点的搜索和确定；
- (2) 计算重构权值矩阵 W ；
- (3) 通过特征映射计算低维嵌入；

一般情况下，采用 k 邻域算法来计算相应的邻域大小。在确定好点 x_i 的邻域后，也就是确定 x_i 的 k 个邻域点，对这 k 个邻域点进行加权，定义权重系数 $W_{ij}, j=1, 2, \dots, k$ 。首先通过每个样本点的近邻点计算出该样本点的局部重构权值矩阵。为了达到线性重构目的，定义一个代价函数使得总体重构误差最小，采用式(2-8)计算权值：

$$\min \varepsilon(W) = \sum_{i=1}^m \left\| x_i - \sum_{j=1}^k W_{ij} x_{i,j} \right\|^2 \quad (2-8)$$

为了保证计算过程不出现发散的情况，要对权值矩阵归一化，即式所示：

$$\sum_{j=1}^m W_{ij} = 1 \quad (2-9)$$

采用拉格朗日乘数法得出构造函数：

$$F(W) = \sum_{j=1}^k \sum_{l=1}^k w_{ij} w_{il} Q_{jl}^{(i)} + \lambda (\sum_{j=1}^k w_{ij} - 1) \quad (2-10)$$

其中 $Q_{jl}^{(i)}$ 的计算式为：

$$Q_{jl}^{(i)} = (x_i - x_{i,j})^T (x_i - x_{i,l}) \quad (2-11)$$

由此，得出权值矩阵的计算式为：

$$W_{ij} = \frac{\sum_{l=1}^k (Q_{jl}^{(i)})^{-1}}{\sum_{p=1}^k \sum_{q=1}^k (Q_{pq}^{(i)})^{-1}} \quad (2-12)$$

计算得出权值矩阵后，代入式(2-7)中，对下式特征分解：

$$My^T = \lambda y^T \quad (2-13)$$

其中 M 为：

$$M = (I - W)^T (I - W) \quad (2-14)$$

为了满足上式的条件， y 是 M 的特征向量，显然使代价函数最小， y^T 是 M 的最小 d 个非零特征值对应的特征向量。需要注意的是，如果从 m 维降到 d 维，需要取 $d+1$ 个特征向量，并舍去其中的 0 向量，这样就得到最后的低维嵌入 y ，实现从 n 维降到 d 维的过程。

2.2.5 简化主成分分析法

简化主成分分析法^[46] (Simple PCA) 是 PCA 算法的迭代方法的实现。与 PCA 相比，其目标同样是最大化降维后的方差。但在分析最大化的方差时，采用方法不是分析数据集的最大化特征分解，而是采用迭代方法来寻找相应的最大的特征向量，即变换向量。

数据降维过程可以由式(2-15)表示：

$$y_i = x_i \cdot \alpha_k \quad (2-15)$$

式(2-15)中， x_i 表示高维数据， y_i 表示低维嵌入结果， α_k 表示变换特征向量。在 PCA 算法中，通过特征分解的方法得到上述 α_k ，而在 SPCA 中采用迭代方法得到相应的 α_k 。其实现过程如下所述：

- (1) 对于给定高维数据集 $X = \{x_1, x_2, x_3, \dots, x_n\}$ ，计算平均向量 $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ ，

从而计算中心化数据矩阵 $X - \bar{X} = X = \{x_1 - \bar{x}, x_2 - \bar{x}, x_3 - \bar{x}, \dots, x_n - \bar{x}\}$;

(2) 对于中心化数据矩阵, 给出以下门限函数来计算其迭代指数:

$$\Phi(y_j, x_i) = \begin{cases} x_i & y_i \geq 0 \\ -x_i & \end{cases} \quad (2-16)$$

对于门限函数, 同样可以采用另外一种形式:

$$\Phi(y_j, x_i) = y_j x_i \quad (2-17)$$

(3) 对于第 $k+1$ 次迭代, 其特征向量计算式为:

$$\alpha^{k+1} = \frac{\sum_i \Phi(y_j, x_i)}{\left\| \sum_i \Phi(y_j, x_i) \right\|} \quad (2-18)$$

通过多次迭代后直至 α 不变, 由此可以计算得到第一个最大的特征向量, 并将该特征向量从所有的特征向量中移除, 然后再次计算相应的特征向量。通过迭代方法得到相应的 d 个特征向量。

SPCA 算法的本质与 PCA 算法一致, 只是采用迭代方法实现相应的算法功能, 实现速度快, 避免了大量的矩阵运算。这种方法的输出与 PCA 方法的输出有一定的不同, 因为迭代方法存在一定的误差。

2.2.6 线性鉴别分析

线性鉴别分析^[22~24] (LDA) 算法是一种监督型的降维算法。其中监督的含义在于需要人为地对所有的数据进行分类处理。假设, 高维数据集为 $X = \{x_1, x_2, x_3, \dots, x_n\}$, 对于高维数据的类标记为 $c = \{c_1, c_2, \dots, c_p\}$, 即将所有数据标记为 p 类。LDA 是一种基于费舍尔准则 (Fisher Criterion) 的算法。费舍尔准则表示为下式:

$$\max \left\{ \frac{V^T S_b V}{V^T S_w V} \right\} \quad (2-19)$$

式(2-19)中, V 表示特征变换矩阵, S_b 、 S_w 分别代表类间散度矩阵、类内散度矩阵, 其计算式分别为:

$$\begin{cases} S_w = \sum_c p_c (\text{cov}(X_c - \bar{X}_c)) \\ S_b = \text{cov}(X - \bar{X}) - S_w \end{cases} \quad (2-20)$$

式(2-20)中, p_c 表示第 c 类的概率, 是一种统计结果。 $\text{cov}(X_c - \bar{X}_c)$ 为第 c 类

的中心化数据的协方差。 $\text{cov}(X - \bar{X})$ 表示所有数据集的协方差。式(2-19)所示的目标函数可以通过计算式的特征分解的前 d 最大特征值对应特征向量为特征变换矩阵 V 。

$$S_b V = \lambda S_w V \quad (2-21)$$

由式(2-21)的特征值分解可以计算得出特征变换矩阵，因此降维结果表示为式(2-22):

$$Y = V(X - \bar{X}) \quad (2-22)$$

2.2.7 邻域保持嵌入

邻域保持嵌入^[50] (Neighbor Preserving Embedding, NPE) 算法的降维目标函数与LLE算法的目标函数一致，如式(2-7)所示。实际上，NPE算法是对LLE算法的一种近似表示。NPE算法的分析推导过程与LLE算法一致，但在求解相应的特征向量时，采用下式进行分析。

$$(X - \bar{X})^T (I - W)^T (I - W)(X - \bar{X})V = \lambda (X - \bar{X})^T (X - \bar{X})V \quad (2-23)$$

其中， V 为变换矩阵，其降维结果由式(2-24)计算：

$$Y = V(X - \bar{X}) \quad (2-24)$$

2.3 降维算法的比较分析

在计算机等现代计算设备出现后，海量数据成为数据处理的主要问题之一。最早的线性降维（如 PCA 算法，LDA 算法）都只能反应出高维数据中内部的线性结构。而数据是非线性性质时，线性降维方法通常不能适用于非线性数据的维数约简。流形学习是一类可以用于挖掘高维数据中内部的低维流形的非线性降维方法。

ISOMAP 是最早提出的一种全局型降维的方法，它能很好的利用一些算法表示高维流形中多维数据点的测地距离，在没有数据空洞的情况下，有很好的学习效果。但保持等距嵌入的带来巨大的计算量，这是 ISOMAP 算法与局部算法相比之下的最大的不足。因此 LLE 作为一个典型的局部降维方法被提出。LLE 作为局部线性嵌入算法，能够较好的保持局部高维结构，将局部近似为线性结构。ISOMAP 算法和 LLE 算法均可以较好的得出高维数据的可视化结果，但并没有对数据的类信息及数据的高维特征进行分析。为了更好的认识高维数据的特征，研究人员提出了局部鉴别嵌入型降维算法。LDE 算法就是一种典型局部鉴别嵌入算法。这种

算法将局部算法与鉴别算法的优点相结合，通过引入对高维数据的类进行分析，从而得到高维数据的特征。LDA 算法同样是一种监督型算法。

本文中拟采用 LDE 降维算法应用于 WiFi 室内定位，分析研究 Radio Map 中的特征，从而利用 LDE 算法来提取相应的特征并对数据进行降维处理。

2.4 本章小结

本章从拓扑学的角度出发，分析了有关于流形的基本的假设，并且介绍了两种典型流形：微分流形及黎曼流形。同时对流形中常用的概念给出了数学定义。基于流形及其相关基本定义，给出流形学习的描述性及数学定义。然后对本文中涉及的多种常见的降维算法的理论及算法的流程进行分析，包括：PCA、SPCA、LDA、MDS、LLE、ISOMAP、NPE 等七种降维算法。通过分析上述七种算法，了解了降维算法的基本理论，为后续 LDE 算法性能的对比研究打下基础。

第 3 章 LDE 算法的理论分析

LDE 算法是一种监督型的非线性流形学习算法。本章拟对 LDE 算法的理论推导进行详细的介绍：主要包括 LDE 算法的流程、LDE 算法目标函数及其广义特征分解表达式的推导。同时为了提高算法的稳定性能，引入了正则化因子，从而改善了算法的性能，并给出了 LDE 算法在 MATLAB 中的实现流程，为后续进一步的仿真分析奠定基础。

3.1 LDE 算法概述

Hwann-Tzong Chen 等在 2005 年提出 LDE 算法。LDE 算法是一种局部鉴别型流形学习算法。LDE 算法的目标是最大化类间散度及且最小化类内散度。从降维算法的分类来讲，LDE 是基于高维数据的局部结构进行降维的算法，因此是一种非线性算法。LDE 算法与 LDA 算法都是鉴别算法的一种。鉴别算法的本质是对高维数据进行标记，如高维数据的类别信息。因此，LDE 算法又是一种监督型算法。作为一种监督型算法，需要人为对高维数据信息进行标记。本文中，采用 KFCM 算法标记相应的高维数据的信息，从而简化 LDE 算法的实现过程。在第四章中对 KFCM 算法进行详细介绍。

LDE 算法通常采用以下步骤实现：

(1) 创建邻接图。创建邻域图时有两种典型的方法来衡量 RSS 信号空间的距离，分别为汉明 (Hamming) 距离和欧几里得 (Euclidean) 距离。本文中，采用欧氏距离近似表示信号空间之间的距离。对于任意一个点 i ，计算其余参考点与它的距离，并按照距离大小排序，距离最近的 k 个参考点定义为参考点 i 的邻域。对于邻域中的点，测地距离等于欧氏距离；对于邻域外的点，用无穷大代表测地距离。

(2) 计算权值分布。在创建邻域图后，根据邻域图的构造情况，根据 LDE 算法要求，计算相应的重构权值距离。目前常用的两种权值的计算是，对于在邻域内的点权值为 1，邻域外的点权值为 0；或者对于邻域内的点，根据高斯近似来估计权值，如公式(3-1)所示，对于邻域外的点权值为 0。本文中，采用后者给出的相应公式与方法来确定重构嵌入矩阵的权重。

$$w = \exp(-dis^2/t) \quad (3-1)$$

其中， w 表示某一点的权值， dis 表示邻域内两点的距离，采用欧几里得距离表示， t 表示归一化系数。

(3) 广义特征分解。在 LDE 算法中，需要对给定的分类数据的类间散度及

类内散度,并且根据特征方程通过广义特征分解的前 d 个最大的特征值对应的特征向量构成特征变换矩阵。

3.2 LDE 算法的理论推导

LDE 算法的输入数据的说明如下:输入高维数据点 $\{x_i\}_{i=1}^m \in R^n$, 数据点 x_i 的类标记为 $l_i \in \{1, 2, \dots, P\}$, 其中 P 表示将高维数据划分为 P 个子流形, 即将输入的高维数据分成 P 类。将输入的高维数据表示成矩阵的形式: $X = [x_1, x_2, \dots, x_m] \in R^{n \times m}$ 。从矩阵表示的形式来看, 矩阵中的列代表一个高维数据点。LDE 算法理论推导如下所述:

(1) 构造邻接图。其中近邻关系是采用 KNN (K Nearest Neighbors, KNN) 算法给出的准则, 即选择数据点最近的 K 个点作为其邻居, 根据高维数据点的类标记信息及其近邻关系构造无方向图 G 及 G' 。 G 表示当 x_i 与 x_j 的类标记信息 $y_i = y_j$ 时且 x_i 、 x_j 互为 K 近邻关系; G' 示当 x_i 与 x_j 的类标记信息 $y_i \neq y_j$ 时且 x_i 、 x_j 互为 K 近邻关系。

(2) 计算权值矩阵。根据(1)构造的邻接图采用类高斯函数进行权值矩阵的计算。其表达式为(3-2)所示。公式(3-2)中 w_{ij} 表示近邻点 x_i 与 x_j 之间的权值, $\|x_i - x_j\|^2$ 为近邻点 x_i 与 x_j 之间的范数 (Frobenius) 距离^[51], 采用矩阵方式计算范数距离, t 为权值归一化参数。

$$\begin{cases} w_{ij} = \begin{cases} \exp\left(-\|x_i - x_j\|^2 / t\right); & \text{若 } x_i, x_j \in G \\ 0; & \text{其它} \end{cases} \\ w'_{ij} = \begin{cases} \exp\left(-\|x_i - x_j\|^2 / t\right); & \text{若 } x_i, x_j \in G' \\ 0; & \text{其它} \end{cases} \end{cases} \quad (3-2)$$

(3) 计算嵌入结果。对式(3-4)进行广义特征值分解, 得出其特征值分解的特征值及特征向量, 表示为: $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_n]^T$, 其对应的特征向量为: $v = [v_1, v_2, \dots, v_n]^T$ 。取前 d 个最大的特征值对应的特征向量构成变换矩阵 $V = [v_1, v_2, \dots, v_d]$ 。由 LDE 算法的输出数据变换方法可以得出, 降维后数据为: 有关式(3-4)的推导及证明在后续的分析中给出。

$$z_i = V^T \cdot x_i \quad (3-3)$$

式(3-3)中, z_i 表示输入高维数据点 x_i 变换后的低维输出数据,

$$X(D' - W')X^T v = \lambda X(D - W)X^T v \quad (3-4)$$

式(3-4)中： X 为输入数据， λ, v 为特征值与特征向量， W, W' 分别为 G 及 G' 对应的权值矩阵， D 及 D' 均为对角阵，其对角元素可以由式(3-5)表示。

$$\begin{cases} d_{ii} = \sum_j w_{ij} \\ d'_{ii} = \sum_j w'_{ij} \end{cases} \quad (3-5)$$

上述的三步即为根据 LDE 算法流程给出的理论分析及说明。

对于根据 LDE 算法的目标——最大化类间散度地同时最小化类内散度。散度采用表示同类数据点及不同类的范数距离表示。由 LDE 算法的目标可以得出其相应的优化目标函数，如式所示。

$$\begin{cases} \text{Maximize } J(V) = \sum_{i,j} \|V^T x_i - V^T x_j\|^2 w'_{ij} \\ \text{subject to } \sum_{i,j} \|V^T x_i - V^T x_j\|^2 w_{ij} = 1 \end{cases} \quad (3-6)$$

根据式(3-6)给出的优化目标函数推导式(3-4)。

根据矩阵范数的计算式： $\|A\|^2 = \sum_{i,j} a_{ij}^2$ ，计算式表示为矩阵 A 的矩阵范数的计算方法，计算式给出的方法与矩阵的迹的计算式一致^[51]，即： $\|A\|^2 = \text{tr}(AA^T)$ 。由此式(3-6)可以表示为矩阵的迹的计算方式：

$$J(V) = \sum_{i,j} \left\{ \text{tr} \left[(V^T x_i - V^T x_j)(V^T x_i - V^T x_j)^T w'_{ij} \right] \right\} \quad (3-7)$$

式(3-7)可以简化为：

$$J(V) = \sum_{i,j} \left\{ \text{tr} \left[V^T (x_i - x_j)(x_i^T - x_j^T) V \right] w'_{ij} \right\} \quad (3-8)$$

由矩阵迹的计算的标量性质及权值元素均为实数，可以将式(3-8)简化为：

$$J(V) = \text{tr} \left\{ V^T \sum_{i,j} \left[(x_i - x_j) w'_{ij} (x_i^T - x_j^T) \right] V \right\} \quad (3-9)$$

将式(3-9)展开：

$$J(V) = \text{tr} \left\{ V^T \sum_{i,j} \left[(x_i w'_{ij} x_i^T) + (x_j w'_{ij} x_j^T) - (x_i w'_{ij} x_j^T) - (x_j w'_{ij} x_i^T) \right] V \right\} \quad (3-10)$$

式(3-10)可以简化为：

$$J(V) = \text{tr} \left\{ V^T \left[\left(\sum_i x_i x_i^T \sum_j w'_{ij} \right) + \left(\sum_j x_j x_j^T \sum_i w'_{ij} \right) - 2 \left(\sum_j x_j^T \sum_i x_i w'_{ij} \right) \right] V \right\} \quad (3-11)$$

对于式(3-11)中：

$$\begin{cases} \sum_i x_i x_i^T = XX^T \\ \sum_j w'_{ij} = d'_{ii} \\ \sum_{i,j} x_i^T w'_{ij} x_j = XW'X^T \end{cases} \quad (3-12)$$

根据式(3-12)中变换关系，可以将式(3-11)简化为：

$$J(V) = 2tr\{V^T [X(D' - W')X^T]V\} \quad (3-13)$$

根据式(3-13)的推导方式，同理可以将(3-6)中的约束条件写成如式(3-13)相似的形式，由此，可以将(3-6)表示为如下形式：

$$\begin{cases} \text{Maximize } J(V) = 2tr\{V^T [X(D' - W')X^T]V\} \\ \text{subject to } 2tr[X(D - W)X^T] = 1 \end{cases} \quad (3-14)$$

对式(3-14)应用拉格朗日（Lagrange）乘数法^[51]，可以得出式(3-4)所示：

$$X(D' - W')X^T v = \lambda X(D - W)X^T v \quad (3-15)$$

由式(3-7)~(3-15)给出 LDE 算法的理论推导。

式(3-15)所示为 LDE 算法的广义特征分解式。广义特征分解的基本条件是分解矩阵满足半正定条件。而在实际应用 LDE 算法时，分解矩阵并不能满足半正定的基本条件，因此，引入正则化方法，对式(3-15)进行正则化处理。因此，实际应用的广义特征分解表达式为(3-16)所示。

$$X(D' - W')X^T v + \mu I = \lambda X(D - W)X^T v \quad (3-16)$$

式(3-16)中， μ 为正则化因子，一般情况下为大于零。 I 为单位阵。通过正则化处理后，可以增加广义特征分解的稳定性^[52]。

3.3 LDE 算法的 MATLAB 实现流程

图 3-1 所示为 LDE 算法的 MATLAB 实现流程图。输入数据为高维待降维数据。由于 LDE 算法是一种监督型，通常要对数据进行一定的数据标记。数据标记由基于核函数的模糊 c 均值聚类算法（第 4 章中详细介绍）进行数据标记。对数据的处理流程包括了基于 k 最近邻方法的近邻估计、权值矩阵的估计及广义特征分解。输出结果包括降维结果，即低维流形的嵌入结果和特征变换矩阵，即表征高维数据的变换特征的特征向量矩阵。

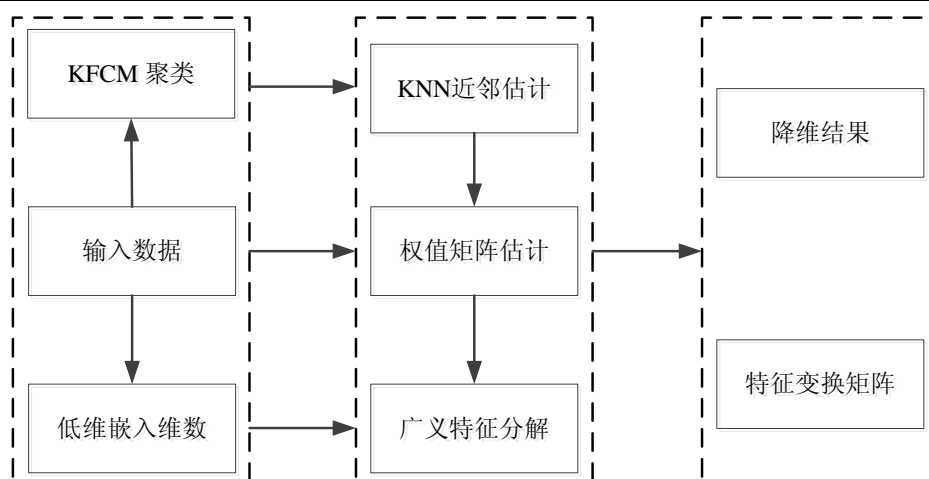


图 3-1 LDE算法流程图

为了使得 LDE 算法更好的应用于目前的 WiFi 室内定位系统中，独立采用一定的算法^[20]对 Radio Map 的维数进行估计，从而得到各个分区的维数的预测结果。因此，在 LDE 算法应用于目前的 WiFi 室内定位系统时，直接输入相应的低维嵌入维数，从而提高 LDE 算法的运行效率。

3.4 本章小结

本章对 LDE 算法的理论推导进行了详细的介绍。主要包括 LDE 算法的流程、LDE 算法目标函数及其广义特征分解表达式的推导。同时为了提高算法的稳定性，引入了正则化因子，从而改善了算法的性能，并给出了 LDE 算法在 MATLAB 中的实现流程，为后续进一步的仿真分析奠定基础。

第 4 章 基于核函数的模糊 c 均值算法

作为一种监督型流形学习算法，LDE 算法需要对所有参考点的 RSS 空间进行类标记。本章拟引入基于核函数的模糊 c 均值聚类算法对分区的位置指纹（Radio Map）进行类标记，以便给 LDE 算法提供有效的类信息。主要给出了 CM 算法的基本思想及其通用的算法流程，并通过引入隶属度的描述，对 FCM 基本理论及其算法流程进行分析。最后，通过平行推广的方式，对 KFCM 算法的理论及其在希尔伯特空间中的实现方式，为 LDE 算法的输入数据的类标记提供有效的算法。

4.1 聚类算法的引入

LDE 是一种监督型流形学习算法，其实现如图 3-1 流程所示。在 LDE 算法实现的过程中需要对所有的参考点进行类标记。考虑到目前的 WiFi 室内定位环境中，参考点数目将近 1000 点，并没有人为地对所有的参考点进行类标记，而是采用一定的分类算法对参考点的类别进行标记。

聚类是一种重要的数据挖掘方法，主要用于发现数据的不同类型以及从数据中识别特定的分布及模式。本文中采用基于核函数的模糊 c 均值（Kernel-based Fuzzy c -means, KFCM）算法对不同分区的参考点进行聚类，从而对所有的参考点进行类信息标记。

4.2 聚类的基本思想

聚类的目标是将数据集 $X = \{x_1, x_2, \dots, x_n\}$ 划分为 c 类且各类数据之间互不相关。基本的聚类算法按如下步骤实现：

- (1) 生成 c 个聚类中心，记为 v_i ， $i=1, 2, \dots, c$ 。
- (2) 将数据集 $X = \{x_1, x_2, \dots, x_n\}$ 的每个元素归类至相应的类，采用最近邻（Nearest Neighbor）算法判定元素的归属关系，其等价表达式为：

$$x_i \in G_i \Leftrightarrow \arg \min_{1 \leq j \leq c} D(x_i, v_j) \quad (4-1)$$

式(4-1)中， x_i 为第 i 个数据点， G_i 为第 i 类构成的邻接关系图。 $D(x_i, v_j)$ 表示计算 x_i 与 v_j 之间的欧式距离。

- (3) 聚类中心的更新，对于第 i 类的聚类中心更新为：

$$v_i = \frac{1}{|G_i|} \sum_{x_k \in G_i} x_k \quad (4-2)$$

式(4-2)中， $|\bullet|$ 表示计算某类中元素的个数。

(4) 收敛性校验及迭代

若满足以下四种情况下的收敛性条件之一，则迭代停止，否则重复执行(2)~(3)，直到迭代收敛或者达到最大执行次数。四种收敛性判定条件为：

条件一：聚类中心不变；

条件二：每个聚类的元素不变；

条件三：聚类中心变化收敛于半径 ε 内；

条件四：聚类元素变化收敛于半径 ε 内。

总的来说，上述收敛性判定条件可以表述为下式：

$$\max \|v_i - v'_i\| \leq \varepsilon, \varepsilon \geq 0 \quad (4-3)$$

其中， v'_i 表示更新后的第 i 类的聚类中心。

从上述聚类算法的基本实现方式分析，用于判定元素的归属关系的算法构成了聚类的核心。不同类型的聚类算法提出不同的归类指标，一般将该函数称为损耗函数 (Loss Function)。基本的聚类方法中采用欧式距离作为损耗函数。以下就本文中提及的模糊 c 均值聚类 (Fuzzy c -means, FCM) 及基于核函数的模糊 c 均值聚类的理论及其实现进行讨论。

4.3 模糊 c 均值聚类

在模糊 c 均值聚类算法中，引入隶属度函数描述数据集 $X = \{x_1, x_2, \dots, x_n\}$ 类的归属。当隶属度函数表述为式(4-4)所示时，此时，模糊 c 均值聚类即为普通 c 均值聚类方法。

$$\begin{cases} J_u = \sum_{i=1}^c \sum_{k=1}^n u_{ki} D(x_k, v_i) \\ u_{ki} = \begin{cases} 1 & x_k \in G_i \\ 0 & x_k \notin G_i \end{cases} \end{cases} \quad (4-4)$$

式(4-4)中， u_{ki} 表示数据元素 x_k 对于类 G_i 的隶属度。当 $D(x_k, v_i)$ 表述为损耗函数时，式(4-4)中的目标是最小化损耗函数的值；若 $D(x_k, v_i)$ 表述为相似度函数，式(4-4)中的目标是最大化相似度函数的值。

通过对隶属度函数进行推广，表述为归一化条件下的概率分布，如式(4-5)所示。

$$U = \left\{ u = \{u_{ki}\}; \sum_{j=1}^c u_{kj} = 1; 1 \leq k \leq n, u_{ki} \in [0,1], 1 \leq k \leq n; 1 \leq i \leq c \right\} \quad (4-5)$$

同时采用非线性方法，对模糊 c 均值聚类算法的目标函数作如下定义：

$$\begin{cases} J_{fcm}(U, V) = \sum_{k=1}^n \sum_{i=1}^c (u_{ki})^m D(x_k, v_i) \\ \text{s.t. } \sum_{i=1}^c u_{ki} = 1, m > 1 \end{cases} \quad (4-6)$$

式(4-6)中， m 表示非线性指数。对式(4-6)采用拉格朗日乘数法（Lagrange Multiplier）解算，得到如下解：

$$\begin{cases} u_{ki} = \left[\sum_{j=1}^c \left(\frac{D(x_k, \bar{v}_i)}{D(x_k, \bar{v}_j)} \right)^{1/m-1} \right]^{-1} \\ v_i = \frac{\sum_{k=1}^n (u_{ki})^m x_k}{\sum_{k=1}^n (u_{ki})^m} \end{cases} \quad (4-7)$$

模糊 c 均值聚类算法的执行流程与普通聚类算法的执行方法一致，只是引入隶属度函数，并采用隶属度来更新相应的聚类中心。

4.4 基于核函数的模糊 c 均值聚类

引入核函数的模糊 c 均值聚类的目标是将原始的数据集所在空间变换至无穷维的希尔伯特空间（Hilbert Space），再对变换后的空间作相应的聚类分析。通过核函数的变换，将原始数据之间的类别特征进一步变换后更易于表述和区分。基于核函数的模糊 c 均值聚类算法的目标函数为：

$$J_{KFCM} = \sum_{i=1}^c \sum_{k=1}^n (u_{ki})^m \|\Phi(x_k) - W_i\|_H^2 \quad (4-8)$$

式(4-8)中， $\Phi(x_k)$ 、 W_i 分别表示在希尔伯特空间下的数据集及相应的聚类中心。由 FCM 算法平行推广，可以得出 KFCM 算法的解表述为：

$$\begin{cases} u_{ki} = \left[\sum_{j=1}^c \left(\frac{\|\Phi(x_k) - W_i\|_H^2}{\|\Phi(x_k) - W_j\|_H^2} \right)^{\frac{1}{m-1}} \right]^{-1} \\ v_i = \frac{\sum_{k=1}^n (u_{ki})^m \Phi(x_k)}{\sum_{k=1}^n (u_{ki})^m} \end{cases} \quad (4-9)$$

KFCM 的解的关键在于计算希尔伯特空间的损耗函数或者相似度函数。在本文中考虑引入高斯核函数（Gaussian Kernel Function）的 FCM 算法的理论分析及其实现。高斯核函数如式(4-10)所示。

$$K(x, y) = e^{-\lambda\|x-y\|^2} \quad (4-10)$$

希尔伯特空间中，由式 $\|\Phi(x_k) - W_i\|_H^2$ 表述其相应的损耗函数，该式进一步表述为式(4-11)。

$$\|\Phi(x_k) - W_i\|_H^2 = \langle \Phi(x_k) - W_i, \Phi(x_k) - W_i \rangle \quad (4-11)$$

式(4-11)中， $\langle \cdot, \cdot \rangle$ 表示计算相应式的核函数值。而实际上，无穷空间的变换不存在，因此，对式(4-11)进一步简化为式(4-12)所示。

$$D_{ki} = \|\Phi(x_k) - W_i\|_H^2 = \langle \Phi(x_k), \Phi(x_k) \rangle - 2\langle \Phi(x_k), W_i \rangle + \langle W_i, W_i \rangle \quad (4-12)$$

式(4-12)的全展开式为式(4-13)所示。

$$\begin{aligned} D_{ki} &= \langle \Phi(x_k), \Phi(x_k) \rangle - \frac{2}{\sum_{k=1}^n (u_{ki})^m} \sum_{j=1}^n \langle \Phi(x_k), \Phi(x_j) \rangle \\ &\quad + \frac{1}{\left(\sum_{k=1}^n (u_{ki})^m \right)^2} \sum_{j=1}^n \sum_{l=1}^n \langle \Phi(x_l), \Phi(x_j) \rangle \end{aligned} \quad (4-13)$$

式(4-13)中， $\langle \Phi(x_k), \Phi(x_j) \rangle$ 由高斯核函数计算，即：

$$\langle \Phi(x_k), \Phi(x_j) \rangle = K(x_j, x_k) = e^{-\lambda\|x_j - x_k\|^2} \quad (4-14)$$

在算法实现中，不是随机生成聚类中心，而是从数据集 $X = \{x_1, x_2, \dots, x_n\}$ 中随机选择 c 个元素作为聚类中心，构成集合 $Y = \{y_1, \dots, y_c\}$ 。因此，初始化的损耗函数值计算如式所示。

$$D_{ki}^o = \|\Phi(x_k) - \Phi(y_i)\| = K(x_k, x_k) - 2K(x_k, y_i) + K(y_i, y_i) \quad (4-15)$$

4.5 本章小结

本章引入了基于核函数的模糊 c 均值聚类算法对分区的位置指纹数据（Radio Map）进行类标记，以便给 LDE 算法提供有效的类信息。主要给出了 CM 算法的基本思想及其通用的算法流程，并通过引入隶属度的描述，对 FCM 基本理论及其算法流程进行分析。最后，通过平行推广的方式，对 KFCM 算法的理论及其在希尔伯特空间中的实现方式，为 LDE 算法的输入数据的类标记提供有效的算法。

第 5 章 基于 LDE 算法的 Radio Map 降维处理

通过上述几章对于流形及常见的降维算法的分析，LDE 算法的理论推导及 KFCM 算法分析，为 LDE 算法在 WiFi 室内定位中的应用奠定了基础。本章拟给出 LDE 算法应用于 WiFi 室内定位系统的方案及与 LDE 进行性能对比的算法的选择指标及其的仿真结果。主要内容包括应用方案及 LDE 对比算法选择指标、实验环境概述、给出 LDE 算法在定位区域内不同分区的仿真结果，同时对 LDE 算法中涉及的多个参数进行分析，以便基于 LDE 算法的 WLAN 室内定位算法进行初步优化。在本章的最后分析了 PCA、SPCA、LDA、NPE 等四种不同降维算法应用于 WiFi 室内定位系统中的仿真结果，并与 LDE 算法的定位结果进行对比。

5.1 LDE 算法在 WiFi 室内定位应用方案及其性能对比算法

5.1.1 实验环境

WiFi 室内定位的实现过程由离线阶段和在线阶段两部分组成（如图 1-1 所示）。离线阶段完成 Radio Map 的建立；在线阶段对实时接收到的 RSS 值采用 KNN 算法进行在线匹配定位。目前 WiFi 室内定位的实验场所为哈尔滨工业大学科学园 2A 栋 12 楼走廊环境，其平面结构图如图 1-2 所示。在实验场所内布置了 27 个 AP，离地面高度为 2 m。所有 AP 几乎均匀分布在实验场所内（图 1-2 中蓝色无线图标所示）。图 5-1 所示为采样网格示意图。以 $0.5m \times 0.5m$ 的采样间隔对实验场所的地理坐标进行标记并采用联想 V450 电脑安装 NetStumbler 软件采集相应位置的 RSS 值及相应的无线接入点的标号，形成 Radio Map。

5.1.2 应用方案

基于 LDE 算法的 WiFi 室内定位方法（LDE_KNN）由离线阶段及在线阶段构成，其实现流程如图 5-2 所示。在离线阶段，采用 LDE 算法对 Radio Map 进行降维，得到相应的降维后的 Radio Map（记为 DR_MAP ）及相应的特征变换矩阵（记为 V ），并将 LDE 算法的输出数据存储为离线匹配数据库。在线阶段，对实时接收的信号强度值进行降维。在线阶段时，不是重复采用 LDE 算法对相应的 RSS 降维处理，而是直接采用特征变换矩阵对 RSS 进行降维。这样处理有方面的优点：一是避免了在线阶段的大量的矩阵运算，提高了算法的实时效率；另一方面，LDE 算法并不可以在单个数据条件下应用。因为单一数据不构成流形，无法进行降维处理。由于特征变换矩阵包含了离线阶段 Radio Map 的特征提取参数，因此，可

以从特征提取的角度对实时接收到的信号强度进行相同的特征提取，从而使得降维后的结果与离线匹配数据库的结果可以相匹配。降维后的 RSS 与降维后的 Radio Map 采用 KNN 算法进行在线匹配定位，从而估计相应位置的坐标。

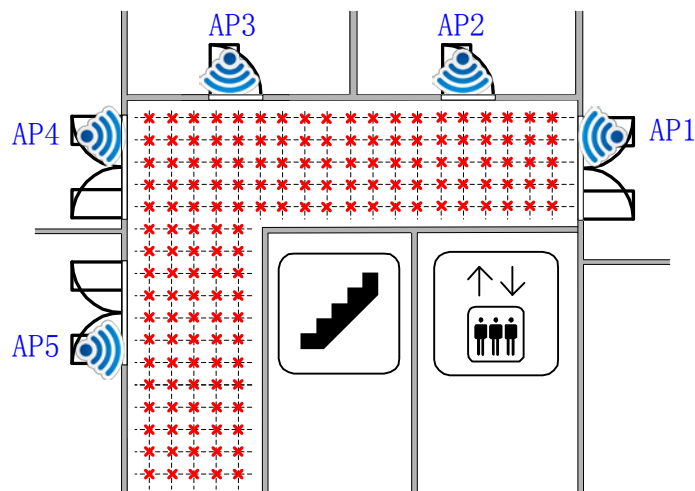


图 5-1 采样网格示意图

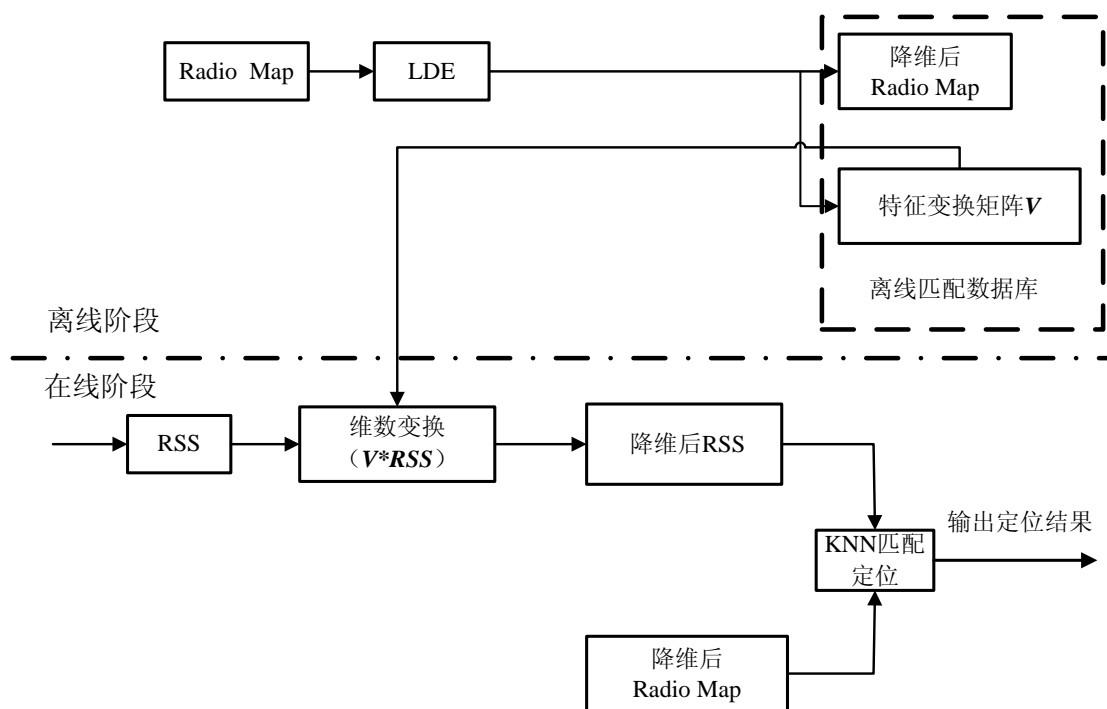


图 5-2 基于 LDE 算法的 WiFi 室内定位的实现流程

5.2.2 对比算法选择

从基于 LDE 算法的 WiFi 室内定位方法出发，与 LDE 算法进行性能比较的算

法应当具备以下特征：

(1) 该算法能对 Radio Map 进行有效地降维；

(2) 该算法的输出至少包含相应的降维结果及特征变换矩阵。也就是该算法实现降维过程与 LDE 算法的过程相似，采用式(3-3)所示形式。

满足上述两个条件的算法很多。本文拟采用 PCA、SPCA、LDA 及 NPE 等四种算法与 LDE 算法的进行性能对比。通过第二章的分析可以得出，以上四种算法的都可以对高维数据进行有效地降维，并且输出数据中包含特征变换矩阵及其降维结果。

5.2 基于 LDE 算法的 WiFi 室内定位方法的性能分析

5.2.1 基于 LDE 算法的 WiFi 室内定位方法的参数分析

实验环境被划分为六个不同的子区域，其分区结果如图 1-2 所示。由于本文并没有分析相应的分区及选区算法，因此，只对 A1 区域的定位情况及相应的定位结果进行分析。A1 区域包含 168 个参考点，每个参考点分东南西北四个不同方位各采集 100 次数据。A1 区域的 Radio Map 选择每个参考点处的接收到的所有 RSS 的某一次的随机时，而没有将所有 400 次不同时刻采集得到的 RSS 进行平均。在线定位阶段也是只在某一个未知位置进行 1 次采样用于在线定位性能测试。

LDE-KNN 算法中包含 LDE 算法及 KNN 算法的所有参数。LDE 算法输入参数有：高维数据集 (RSS_{ref})、聚类数目 (k_{clu})、邻域范围 (k_{LDE})、低维嵌入维数 (d_{LDE}) 及正则化参数 (μ)，输出结果有：低维嵌入结果 (RSS_{dr_ref}) 及特征变换矩阵 (V)。KNN 算法的输入参数包括：在线接收到 RSS (RSS_{test})、参考近邻数 (k_{ref}) 及信号空间距离描述 (d_{rss})，输出结果为测试点定位结果 ((\hat{x}, \hat{y}))。上述参数中，需要优化的有 k_{clu} 、 k_{LDE} 、 d_{LDE} 、 k_{ref} 、 d_{rss} 等五个参数。本文中并没有从理论上分析参数的不同取值的影响，主要是从对 A1 区的仿真分析得出局部优化的参数。对参数进行局部优化时，以误差半径为 0.5 米、1 米、1.5 米及 2 米时的置信概率的变化来分析不同的变量的影响及作用。

下面来分析聚类数目变化对定位性能的影响。本文采用 KFCM 算法对输入的数据进行聚类，从而对输入的 Radio Map 进行类别标记。图 5-1 所示为聚类数目变化对定位性能影响的分析示意图。在 A1 区域，聚类数目从 1 至 9 变化时，误差半径分别取 0.5 米、1 米、1.5 米及 2 米时的置信概率的变化有着近似的趋势。从图中可以得出两个局部峰值，分别是 $k_{clu}=2, 5$ 。从算法运行效率来看，聚类数目应取 2；从 2 米内定位精度来看，聚类数目取 5 较为合理。由于 LDE 算法是在离

线阶段对 Radio Map, 可以承受较高的算法复杂度且对算法的实时运行效率没有高要求, 因此, 取 $k_{clu}=5$ 。同时考虑到局部最优的不稳定性, 在 LDE 算法的全参数分析时, 对 $k_{clu}=4、5、6$ 均进行相应的仿真分析。

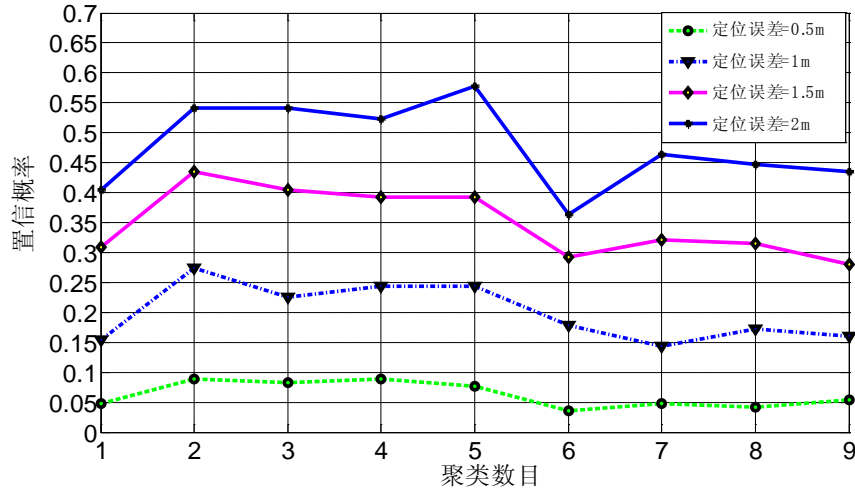


图 5-3 聚类数目变化对定位性能的影响

在确定 KFCM 的参数之后, 在 $k_{clu}=5$ 条件下分析低维嵌入维数对定位结果的影响。图 5-4 所示为低维嵌入维数变化对于定位性能的影响。对于 A1 区域, 当低维嵌入维数发生变化时, 不同的定位误差条件的曲线变化趋势近似相同。以 2 米内定位精度为例, 当 $d_{LDE}=9$ 时, 定位保持稳定且同时可使得其他条件时的定位精度达到局部的最优。因此, 对 A1 区域的仿真采用 $d_{LDE}=9$ 。同时考虑到局部最优的不稳定性, 在 LDE 算法的全参数分析时, 对 $d_{LDE}=8, 9, 10$ 均进行相应的仿真分析。

在确定 KFCM 的参数及低维嵌入维数之后, $k_{clu}=5$ 且 $d_{LDE}=9$ 的条件下仿真 LDE 算法中邻域范围发生变化时对定位性能的影响。在仿真过程中, 分析了邻域范围的 24 个不同的取值。图 5-5 所示为仿真结果。从图中可以看出, 对于 A1 区域而言, 邻域范围的变化对定位结果影响的变化很大。从局部最优来看, 可以找到多个不同的取值。从 4 条不同曲线的变化趋势来看, 局部最优解基本上是重合的。从局部最优解中, 可以得出局部最优的最优解为 $k_{LDE}=5$ 时。同时考虑到局部最优的不稳定性, 在 LDE 算法的全参数分析时, 对 $k_{LDE}=4, 5, 6$ 均进行相应的仿真分析, 同时由于邻域范围对定位的影响的波动性很大, 因此, 在后续所有五个参数进行仿真中会对邻域范围的变化进行更大范围的取值分析。

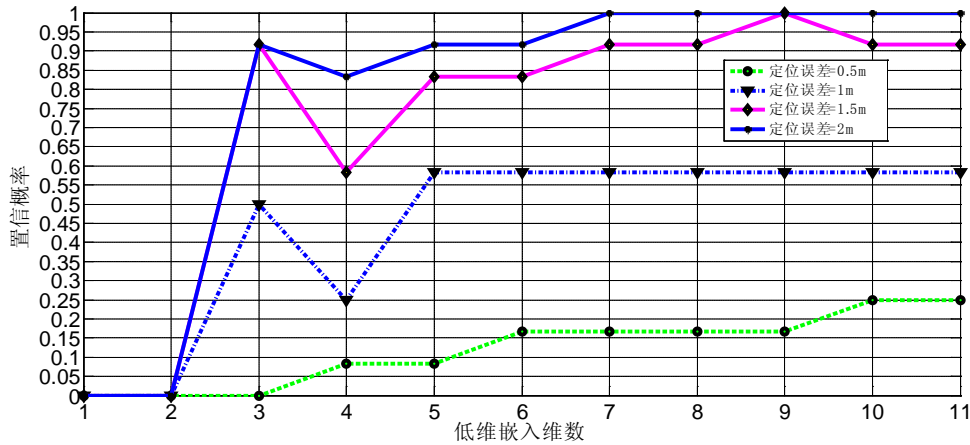


图 5-4 低维嵌入维数变化对定位性能的影响

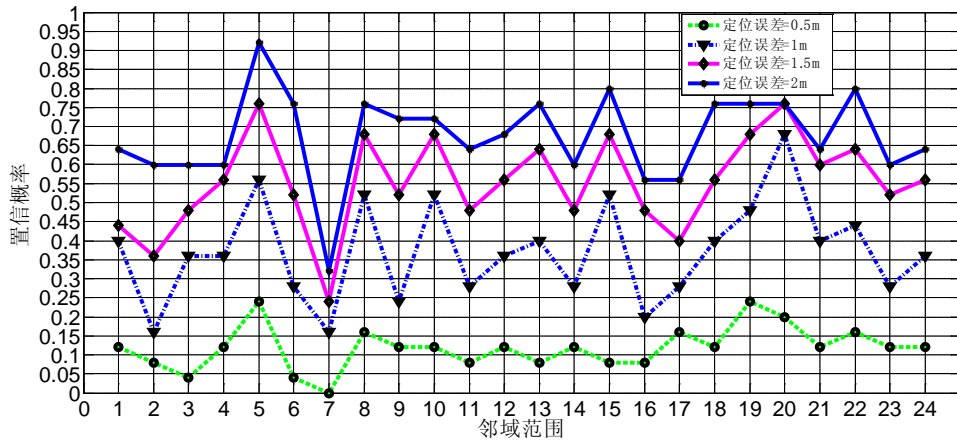


图 5-5 邻域范围变化对定位性能的影响

在确定上述三个参数后，分析正则化因子对于定位结果的影响。从理论上分析，引入正则化因子的目的是保证特征分解矩阵的半正定性，其取值比较灵活，不会对 LDE 算法的输出结果产生影响。图 5-6 所示的定位结果也证实了这一点。对于 LDE 算法来说，正则化因子的取值对定位结果保持很稳定。 μ 取值为 1 可以简化算法运行过程。

通过对 LDE 算法的四个重要的输入参数进行分析可以得出：LDE 算法存在局部优化参数。通过对上述参数的取值进一步组合后，可以得出九组不同的 LDE 算法仿真参数。对于 KNN 算法的 k_{ref} 及 d_{rss} 的取值并没有进一步进行仿真。同时分析 LDE 算法的邻域范围的所有局部最优解。因为对于信号的空间的距离描述沿用的是 LDE 算法中对于距离的描述。在 LDE 算法中寻找邻域范围时，实质上采用信号空间的欧式距离描述相应的距离。因此， $d_{rss}=2$ 。对于 k_{ref} 采用平面三点定位的思考，取值为 3。上述分析中得出了 LDE-KNN 算法的全部五个仿真参数。通过多次

仿真得出：当 $k_{clu}=4$ ， $d_{LDE}=9$ 且 $K_{LDE}=8$ 时有局部最优的结果。在接下来的 A1 区性能分析中，在未指定相应的仿真参数时，通常采用上述参数进行仿真。

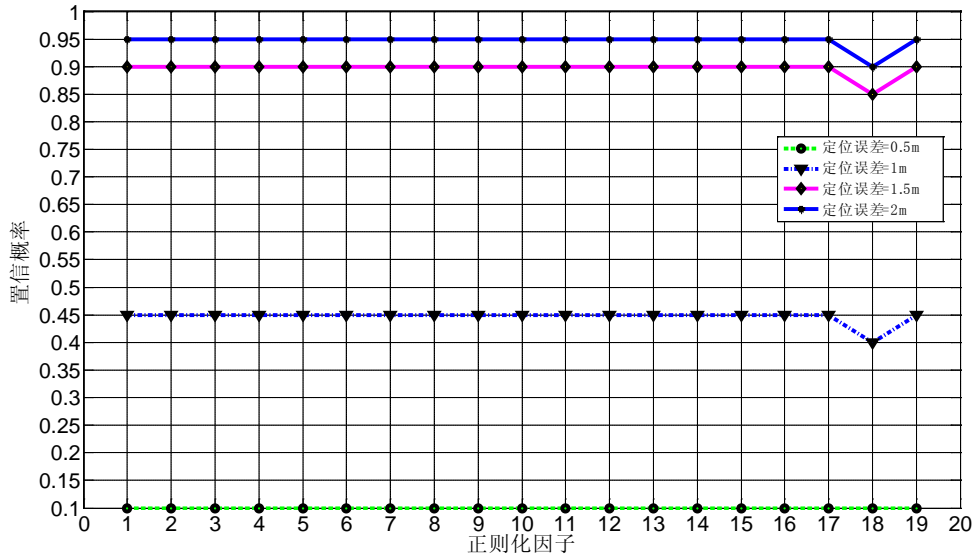


图 5-6 正则化因子变化对定位性能的影响

图 5-7 所示为优化参数条件下的 LDE-KNN 算法的定位结果。从图中可以得出： $k_{clu}=4$ ， $d_{LDE}=9$ 且 $K_{LDE}=8$ 时，在 2 米内定位精度 LDE-KNN 与 KNN 相同。从定位结果的整体趋势来看，LDE-KNN 与 KNN 算法的定位结果可以比拟。而 LDE-KNN 算法采用的维数仅为 KNN 的三分之一。从 KNN 算法的复杂度来看，LDE-KNN 算法的复杂度仅为 KNN 的三分之一，而在线阶段的定位性能与 KNN 算法几乎相同。

在线定位时，采用 KNN 算法对位置指纹数据库进行匹配定位。KNN 算法的复杂度与总的参考点数目 (m)、AP 的数目，即 Radio Map 的维数 (D) 有关。KNN 算法是通过计算在线阶段 RSS 与所有参考的点的距离来选择最近的 k 个参考点计算相应的测试点的位置坐标。以平方次数计算复杂度，KNN 算法复杂度为 $O(mD)$ 。而对于 LDE-KNN 算法，LDE 算法计算过程主要在离线阶段完成，对于在线阶段的定位效率不会有明显的影响。LDE-KNN 算法在线定位阶段的复杂度包括：在线阶段 RSS 的降维及降维后的 KNN 算法的复杂度。假设降维后维数为 d ，则降维后的 KNN 算法复杂度为 $O(md)$ 。以乘法次数计算复杂度，在线阶段 RSS 的降维复杂度为 $O(Dd)$ 。在一般情况下， $m \gg D$ 。因此，LDE-KNN 算法的复杂度为 $O(md)$ 。由上述分析可以得出 LDE-KNN 算法的在线定位复杂度仅为 KNN 算法的 d/D 。当 $D \gg d$ 时，采用 LDE-KNN 算法可以极大程度上降维在线定位的复

杂度。从本文针对的 WiFi 室内定位方法来看，在优化参数时，算法复杂度下降为原来的三分之一，而 LDE-KNN 算法的性能与 KNN 算法性能可比拟。

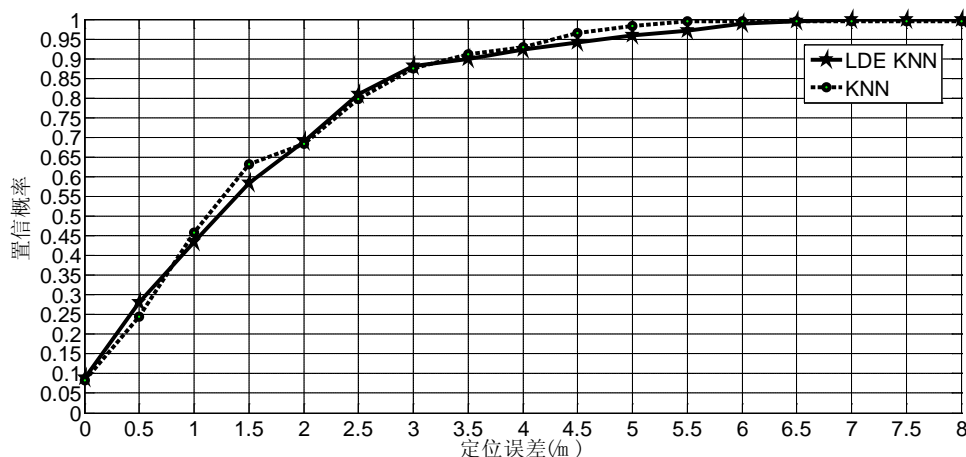


图 5-7 优化参数的 LDE-KNN 算法定位结果

5.2.2 基于 LDE 算法的 WiFi 室内定位方法性能分析

实验环境被划分为六个不同的子区域，其分区结果如图 1-2 所示。由于本文并没有分析相应的分区及选区算法，因此，只对 A1 区域的定位情况及相应的定位结果进行分析。A1 区域包含 168 个参考点，每个参考点分东南西北四个不同方位各采集 100 次数据。A1 区域的 Radio Map 选择每个参考点处的接收到的所有 RSS 的某一次的随机时，而没有将所有 400 次不同时刻采集得到的 RSS 进行平均。在线定位阶段也是只在某一个未知位置进行 1 次采样用于在线定位性能测试。

一般情况下，假设所有 AP 均开启。采用 NetStumbler 软件进行在线测试数据时，假设环境噪声为 -100 dBm 。因此，采集到的 RSS 出现信号值低于 -100 dBm 时，则认为没有接收到相应的 AP 的信号，即该 AP 不可见。

图 5-8 所示为所有 AP 开启时，A1 区域的仿真定位结果。图中实线表示的是 KNN 算法的定位精度，不同标记的虚线表示基于 LDE 算法的 WiFi 室内定位方法的定位结果。图中 d 表示降维后的维数。目前 WiFi 室内定位的 Radio Map 为 27 维。当 $d=2$ 时，只保持了很少的特征，因此定位结果很差。随着 d 的取值的增加，特征矩阵中包含的 Radio Map 的特征信息也会增加。因此，在线阶段的定位性能也随着 d 的增加而提高。当 $d \geq 9$ 时，在线定位结果，定位误差小于 2 米时，可以很好的逼近的 KNN 算法。当 $d=27$ 时，即提取出所有的特征时，在线定位结果会有一些的提高，而且可以很好的逼近 KNN 算法。

在线阶段随机关闭部分 AP，其本质就是增加了不可见 AP 的数目。这一种情形造成离线阶段与在线阶段的可见 AP 不匹配，给在线匹配定位过程中带来非对称

匹配问题。当在某一区域内，无法接收到某一 AP 的 RSS，称该 AP 不可见，反之，称为可见。在 A1 区，Radio Map 中的多数情况下不可见的 AP 有 6 个，分别是 AP22~AP27。从实验环境的布置来看，AP22~AP27 距离 A1 区域较远，而且受到了墙体等的阻隔。因此，AP22~AP27 容易出现在 A1 区域不可见。图 5-9 所示为关闭 A1 区域内的多数情况不可见 AP：AP22~AP27 时的定位结果。图 5-10 中，三条仿真曲线分别为 LDE 表示的是关闭 AP22~AP27 时应用基于 LDE 算法的 WiFi 室内定位方法时的定位结果、原始 KNN 是指没有关闭 AP 条件时的 KNN 定位结果、AP22~AP27 关闭条件下，KNN 算法的定位结果。从图中可以得出，关闭 AP22~AP27 并没有对定位结果产生太大的影响。原始 KNN 与关闭 AP 条件下的 KNN 的仿真结果有一定的差距，但是在定位误差小于 2 米时，两种条件下的定位结果几乎重合。而在关闭 AP22~AP27 时，LDE_KNN 可以保持较好的定位精度，在不可见 AP 关闭条件下仍可以有效地提取出相应的特征数据，进行定位。同时，从图中可以得出，在保持一定定位精度条件下，在线阶段进行匹配定位时，根据 AP 的可见性，自适应地选择可见 AP 进行定位，从而减少在定位阶段的运算量，提高实时定位算法的实时性。

在实际在线定位时，由于实验环境的变化，可能出现在建立 Radio Map 时可见的 AP，在线定位时不可见。这种情形下，是否可以采用 LDE 算法来保持一定的定位精度呢？当假设关闭可见 AP 时，认为只是在线阶段的不可见 AP 的增加，而并不从 RSS 中分析 AP 的不可见性的原因。图 5-10 所示为随机关闭 1 个可见 AP 时的定位结果。图中，仿真的情形是随机关闭了可见的第 18 个 AP。2 米内定位精度相比，当关闭 AP18 时，直接采用 KNN 算法进行定位时比没有关闭 AP 时的定位精度下降 10%，采用 LDE-KNN 算法进行定位匹配时，定位精度下降 5%。在关闭 1 个 AP 时，在一定的参数条件下，LDE-KNN 算法定位精度的下降幅度小于直接采用 KNN 算法进行匹配定位。但是当关闭 2 个或者 2 个以上的可见 AP 时，直接应用 KNN 算法和 LDE-KNN 算法都无法有效地保持定位精度不变。

从仿真结果来看，当随机关闭一个可见 AP 时，LDE-KNN 算法能够有较好地定位精度。而当有 2 个或者 2 个以上的可见 AP 关闭时，LDE-KNN 算法同样不可以保持较好地定位精度，无法在非对称条件实现匹配定位。

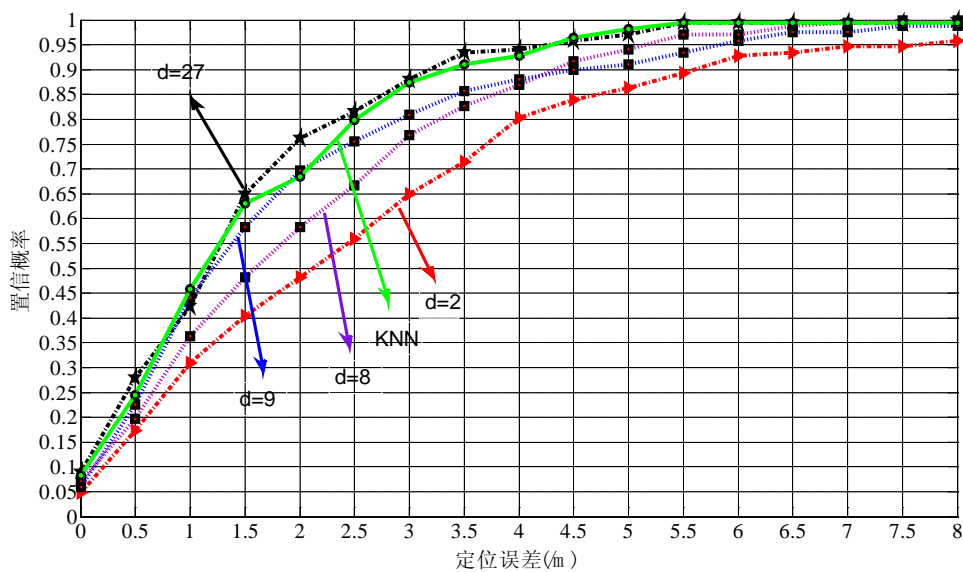


图 5-8 AP 全开启时 A1 区域的定位结果

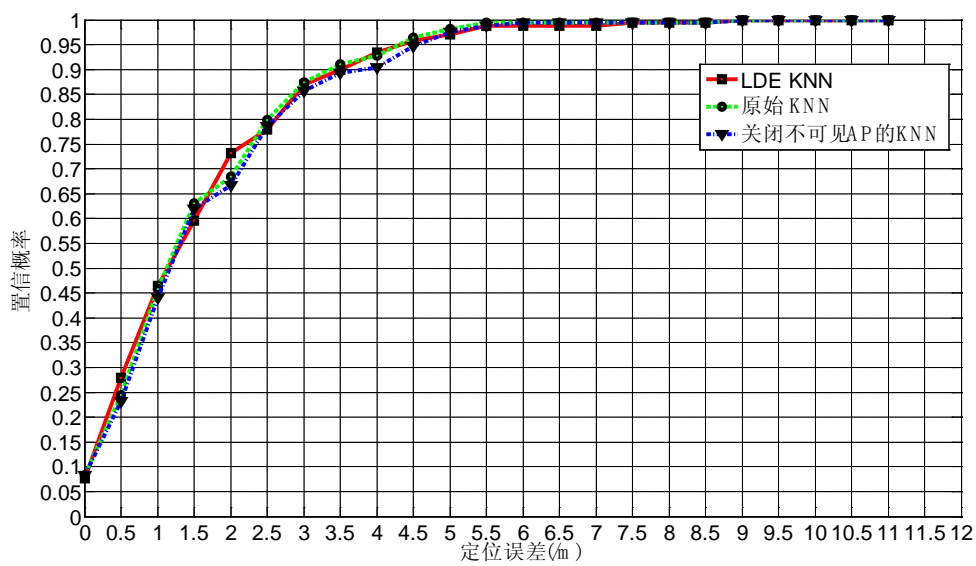


图 5-9 关闭不可见 AP 时的定位结果

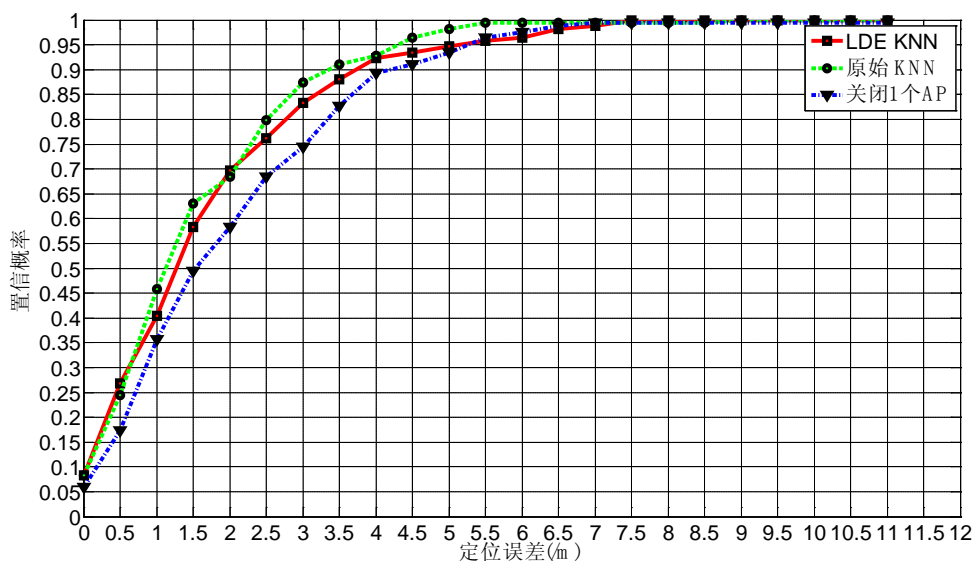


图 5-10 随机关闭 1 个可见 AP 时的定位结果

5.2.3 基于 LDE 算法的 WiFi 室内定位方法的性能对比分析

本文提出采用 LDE 算法对 Radio Map 进行降维处理，从而实现对高维 Radio Map 的降维处理。而现有的降维处理算法很多。本文选择 PCA、SPCA、LDA 及 NPE 联合 KNN 算法 (*-KNN) 进行定位处理，从而与 LDE-KNN 进行性能对比。在同等参数条件下，分析比较 A1 区的定位结果。

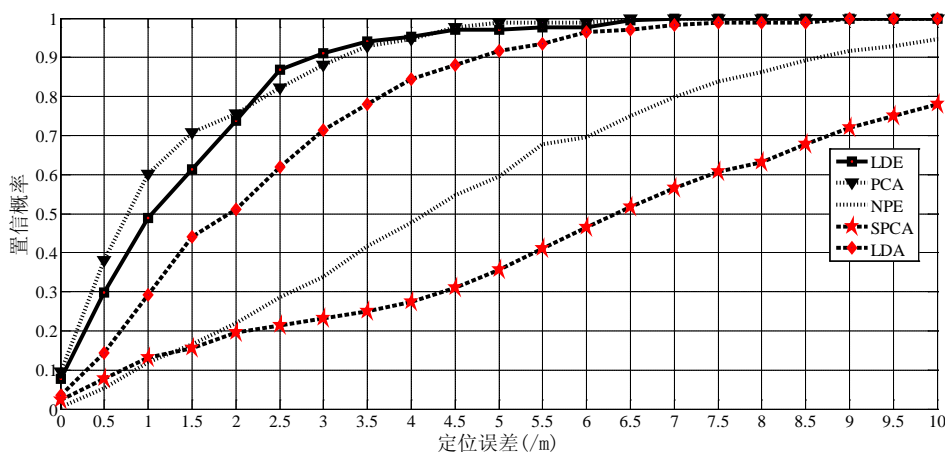


图 5-11 基于不同降维方法的 WiFi 室内定位方法的仿真分析

图 5-11 所示为基于不同降维方法的 WiFi 室内定位方法的仿真分析。从图中可以看出，在相同参数的条件下 LDE-KNN 与 PCA-KNN 性能可以比拟。但是 PCA-KNN 需要传递 Radio Map 的均值到在线定位阶段，而在实际定位过程中，要将大量的数据传输到在线定位阶段会影响实时定位效率。从仿真结果来看

LDA-KNN、NPE-KNN 及 SPCA-KNN 都没有较好的性能。LDA 算法与 LDE 算法同是一种监督型算法，均采用 KFCM 算法提供相应的类标记信息。而在同等参数条件下，LDA-KNN 算法性能远低于 LDE-KNN 算法。而 NPE-KNN 及 SPCA-KNN 的性能都很差。SPCA 算法是 PCA 算法的迭代方式实现。由于是采用迭代方法实现，对于同一条件时，由于迭代次数的不确定性，无法保证相应的降维结果与 PCA 算法相同，因此，PCA-KNN 算法与 SPCA-KNN 算法相差很大

5.3 本章小结

本章给出了 LDE 算法应用于 WiFi 室内定位系统的方案及与 LDE 进行性能对比的算法的选择指标及其的仿真结果。主要内容包括应用方案及 LDE 对比算法选择指标、实验环境概述、给出 LDE 算法在定位区域内不同分区的仿真结果，同时对 LDE 算法中涉及的多个参数进行分析，以便基于 LDE 算法的 WLAN 室内定位算法进行初步优化。在本章的最后给出了 PCA、SPCA、LDA、NPE 等四种不同的降维算法应用于 WiFi 室内定位系统中的仿真，并与 LDE 算法的定位结果进行对比。

结 论

针对目前基于 WLAN 的室内定位系统中存在的由于接入点数目变化引起的高维位置指纹及非对称匹配定位的问题，采用 LDE 算法对 Radio Map 降维并提取特征矩阵。通过对 Radio Map 降维，达到减少数据冗余的目的，同时利用 Radio Map 的特征矩阵解决在线阶段的非对称匹配定位问题。局部鉴别嵌入是一种监督型非线性流形学习方法，需要对 Radio Map 进行类标记。考虑到目前基于 WLAN 的室内定位区域内的参考点的数目大，不适合人为地对所有的参考点进行分类。因此采用 KFCM 方法对 Radio Map 进行类标记，从而简化 LDE 算法的实现过程，提高 LDE 算法的运算效率。仿真结果表明，本文提出的 LDE-KNN 方法可以有效地降低在线定位阶段的算法复杂度，并且与 KNN 位置指纹匹配定位算法的定位性能可比拟。

(1) 针对目前基于 WLAN 的室内定位系统中存在的由于接入点数目变化引起的高维位置指纹及非对称匹配定位的问题，采用 LDE 算法对 Radio Map 进行降维。本文分析了流形及流形学习的基本数学概念及理论，并且对 PCA、SPCA、MDS、ISOMAP、LLE、LDA、NPE 等七种常见的降维算法的基本理论进行了分析；详细推导了 LDE 算法的理论过程、给出了 LDE 算法流程及其 MATLAB 实现，为后续研究奠定基础。

(2) 针对目前基于 WLAN 的室内定位系统中无法人为地为 LDE 算法的输入位置指纹提供有效地类标记的问题，采用 KFCM 算法对位置指纹进行聚类分析。本文对聚类方法的引入及 CM、FCM、KFCM 等方法等基本理论进行了详细分析，以便采用 KFCM 算法标记位置指纹的类信息。

(3) 通过对 LDE-KNN 方法的所有 6 个参数进行 MATLAB 仿真分析，实现 LDE-KNN 算法的优化。A1 子区域的仿真结果表明，采用 LDE-KNN 方法与 KNN 方法在 2 米内的定位精度均可达到 70% 左右，且优化后的 LDE-KNN 方法的复杂度仅为 KNN 方法的三分之一。通过与 PCA、SPCA、LDA、NPE 等四种降维方法进行性能对比可以得出：在同等参数条件下，LDE-KNN 算法的性能优于上述四种方法。

本文有待于进一步研究的问题有以下几点：首先，本文仅研究了目前 WiFi 定位区域中的 A1 子区域的参数及其性能，后续工作将对所有子区域的参数及性能进行分析，从而将算法应用于整个 WiFi 定位区域内；其次，本文采用的 KFCM 聚类算法中聚类中心的选取是一个随机过程，通过仿真分析可以得出：聚类算法的波动性对于算法性能有一定的影响，后续应研究如何提高聚类方法的分类准确度及

其稳定性；最后，后续研究应对本文提出的四种进行性能比较的降维算法的参数作进一步分析、对算法进行必要的改进，并且分析四种算法应用于不同定位区域时的性能。

参考文献

- [1] K. Axel. Location-based Services: Fundamentals and Operation [M]. New York: Wiley: 2005.
- [2] K. W. Kolodziej, J. Hjelm. Local Positioning Systems: LBS Applications and Services [M]. New York: Taylor & Francis: 2006.
- [3] R. Want, A. Hopper, V. Falco et al. The Active Badge Location System [J]. ACM Transactions on Information Systems. 1992: 10(1): 91-102.
- [4] J. Hightower, G. Borriello. Location Systems for Ubiquitous Computing [J]. IEEE Computer, 2001: 34: 57-66.
- [5] T. Darrell, G. Gordon, M. Harville, J. Woodfill. Integrated Person Tracking Using Stereo Color, and Pattern Detection [J]. International Journal of Computer Vision, 2000: 175-185.
- [6] P. Bahl, V. N. Padmanabhan. RADAR: An In-building RF-based User Location and Tracking System [C]. In: INFOCOM 2000. NINETEENTH Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings of IEEE. 2002: 775~784.
- [7] P. Bahl, V. N. Padmanabhan, A. Balachandran. Enhancements to the RADAR User Location and Tracking System [R]. Microsoft Research Technical Report: MSR-TR-00-12, Fed., 2000.
- [8] T. Roos, P. Myllymaki, and H. Tirri et al. A Probabilistic Approach to WLAN User Location Estimation [J]. International Journal of Wireless Information Networks. 2002: (9) 155~164.
- [9] E. A. Martinez, R. Cruz, J. Favela. Estimating Use Location in a WLAN Using Back-propagation Neural Networks [J]. Advances in Artificial Intelligence-IBERAMIN 2004. 2004: 737~746.
- [10] J. J. Pan, J. T. Kwok, and Yang Q. et al. Multidimensional Vector Regression for accurate and Low-cost Location Estimation in Pervasive Computing [J]. IEEE Transactions on Knowledge and Data Engineering. 2006: 1181~1193.
- [11] A. Kushki, K. N. Plataniotis, A. N. Venetsanopoulos. Kernel-based Positioning in Wireless Local Area Networks [J]. IEEE Transaction on Mobile Computing. 2007: 689~705.
- [12] I. Guvenc, C. Abdallah, and R. Jordan et al. Enhancements to RSS Based Indoor Tracking System Using Kalman Filter [C]. In: Proceedings of International Signal Processing Conference (ISPC), Dallas, TX, US, 2003.
- [13] D. Niculescu, B. Nath. VOR Base Stations for Indoor 802.11 Positioning [C]. In:

- Proceedings for the 10th Annual International Conference on Mobile Computing and Networking, 2004: 69.
- [14] C. Gentile, L. Klein-Berndt. Robust Location Using System Dynamics and Motion Constraints [C]. In: IEEE International Conference on Communications. 2004: 1360~1364.
- [15] D. Niculescu, B. Nath. VOR Base Stations for Indoor 802.11 Positioning [C]. In: Proceedings for the 10th Annual International Conference on Mobile Computing and Networking, 2004: 69.
- [16] Dartmouth College WLAN. <http://crowdad.cs.dartmouth.edu>.
- [17] Aruba Selected By Microsoft For Next Generation Wireless LAN. <http://www.arubanetworks.com/company/press/2005/06/13/>.
- [18] A. P. Jardosh, G. Iannaccone, K. Papagiannaki, Vinnakota B. Towards an Energy-Star WLAN Infrastructure [J]. Eighth IEEE Workshop on Mobile Computing Systems and Applications. 2007: 85-90.
- [19] 刘兴堂, 周自全, 李为民, 孙德建. 现代导航与测控[M]. 北京: 科学出版社. 2010: 283-284.
- [20] L. J. P. Maaten, E.O. Postma, and H. J. Herik. Dimensionality reduction: A comparative review [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007.
- [21] S. Y. Kung, K. I. Diamantaras, and J. S. Taur. Adaptive Principal component EXtraction (APEX) and Applications [J]. IEEE Transactions on Signal Processing, 42(5):1202–1217, 1994.
- [22] R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification [M]. Wiley Interscience Inc., 2001.
- [23] R. A. Fisher. The use of multiple measurements in taxonomic problems [J]. Annals of Eugenics, 7:179–188, 1936.
- [24] H. Hotelling. Analysis of a complex of statistical variables into principal components [J]. Journal of Educational Psychology, 24:417–441, 1933.
- [25] J. B. Tenenbaum. Mapping a manifold of perceptual observations [J]. In Advances in Neural Information Processing Systems, MA, USA. The MIT Press. 1998: (10) 682–688.
- [26] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by Locally Linear Embedding [J]. Science, 290(5500):2323–2326, 2000.
- [27] A. Jardosh, K. Papagiannaki, E. Belding. Green WLANs: On-Demand WLAN Infrastructures [J]. Mobile Networks and Applications, 2009, 14(6): 798-814.
- [28] Lin Ma, Yubin Xu, Di Wu. A Novel Two-Step WLAN Indoor Positioning Method [J]. Journal of Computational Information Systems. 2010: 6(14)

- 4627-4636.
- [29] 万鸣华. 基于图嵌入的特征提取与人脸识别研究[D]. 南京理工大学, 2011.06: 27~29.
- [30] 李波. 基于流形学习的特征提取方法及其应用研究[D]. 中国科学技术大学, 2008.08: 43~62.
- [31] I. Hattie. Principal Curves and Surfaces [R]. 1984. 055-056.
- [32] S. K. Nayar, S. A. Nene, H. Murase. Subspace Methods for Robot Vision[R]. 1995. 06-95.
- [33] C. Bregler, S. Omohundro. Nonlinear Manifold Learning for Visual Speech Recognition [C]. In Proceedings of 5th International Conference on Computer Vision. 1995: 494-499.
- [34] C. Bregler, S. M. Omohundro. Nonlinear Image Interpolation Using Manifold Learning [C]. Advances in Neural Information Processing Systems. Cambridge, MA, 1995.
- [35] H. S. Seung, D. D. Lee. The Manifold Ways of Perception [J]. Science.2000, 290(5500): 2268-2269.
- [36] M. Belkin, P. Niyogi. Laplacian Eigen-maps for Dimensionality Reduction and Data Representation [J]. Neural Computation. 2003: 15(6): 1373-1396.
- [37] K. Q. Weinberger, F. Sha, L. K. Saul. Learning A Kernel Matrix for Nonlinear Dimensionality Reduction [C]. Proceedings of the 21st International Conference on Machine Learning. 2004: 106-113.
- [38] Z. Zhang, H. Zha. Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment [J]. SIAM Journal of Scientific Computing.2004, 26(1): 313-338.
- [39] L. Yang. Alignment of Overlapping Locally Scaled Patches for Multidimensional Scaling and Dimensionality Reduction [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2008: 30(3) 438-450.
- [40] T. Lin, H. Zha. Riemannian Manifold Learning [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2008, 30(5): 796-809.
- [41] 詹德川, 周志华. 基于集成的流形学习可视化[J]. 计算机研究与发展. 2005, 42(9): 1535-1537
- [42] 续爽, 贾云得. 基于表情相似性的人脸表情流形[J]. 软件学报. 2009: 20(8) 2191- 2198.
- [43] 刘法旺, 贾云得. 基于流形学习与隐条件随机场的人体动作识别[J]. 软件学报. 2008: 19(12) 69-77.
- [44] 詹德川, 周志华. 基于流形学习的多示例回归算法[J]. 计算机学报.

- 2006:29(11) 1948–1955.
- [45] 杨剑, 王珏, 钟宁. 流形上的 Laplacian 半监督回归[J]. 计算机研究与发展. 2007: 44(7) 1121–1127.
- [46] Qiong Yang and Xiao qing Ding. Symmetrical PCA and its application to face recognition. Chinese journal of computers, Vol.26, No. 9, pp.1146-1151,2003
- [47] T. Cox and M. Cox. Multidimensional scaling [M]. Chapman & Hall, London, UK, 1994.
- [48] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a non-metric hypothesis [J]. Psychometrika,29:1–27, 1964.
- [49] 李养成, 郭瑞之, 崔登兰. 微分流形基础[M]. 北京: 科学出版社, 2011.
- [50] G. E. Hinton and S. T. Roweis. Stochastic Neighbor Embedding [J]. In Advances in Neural Information Processing Systems, volume 15, pages 833–840, Cambridge, MA, USA, 2002. The MIT Press.
- [51] 吴勃英, 王佳明, 丁效华, 李道华. 数值分析原理[M]. 北京: 科学出版社, 2003: 39~45.

攻读学士学位期间发表的论文及其它成果

（一）发表的学术论文

- [1] Lin Ma, **Caifa Zhou**, Xi Liu, Yubin Xu. Adaptive Neighboring Selection Algorithm Based on Curvature Prediction in Manifold Learning. Journal of Harbin Institute of Technology (New Series). (EI 源, 已录用)
- [2] Lin Ma, **Caifa Zhou**, Danyang Qin, Yubin Xu. Green WLAN RSS Dimensionality Reduction and Indoor Localization Based on Fingerprint Algorithm. International Journal of Communication Systems. (under reviewing)
- [3] 夏颖, 张中兆, 马琳, **周才发**. 基于半监督流形学习的 WLAN 室内定位算法. 系统工程与电子技术. (在审)

（二）申请及已获得的专利

- [1] 马琳, **周才发**, 徐玉滨, 强蔚, 孟维晓. 基于 LDE 算法的 WiFi 室内定位方法. (申请号: 201310029536.1)
- [2] 马琳, **周才发**, 刘曦, 马欣如, 徐玉滨, 强蔚. 基于曲率预测的流形学习自适应邻域选择算法. (申请号: 201210325652.3)
- [3] 马琳, 马欣如, 刘曦, **周才发**, 徐玉滨, 强蔚. 基于子区域信息熵的 WLAN 室内定位方法. (申请号: 201210329662.4)
- [4] 马琳, 邓仲哲, 徐玉滨, 孟维晓, **周才发**. 一种卫星辅助信号覆盖图建立/更新的 WiFi 室外定位方法. (申请号: 201310073032.X)

（三）参与的科研项目及获奖情况

- [1] 基于绿色 AP 的 WLAN 室内定位算法研究, 国家自然科学基金项目. 课题编号: 61101122.

哈尔滨工业大学本科毕业设计（论文）原创性声明

本人郑重声明：在哈尔滨工业大学攻读学士学位期间，所提交的毕业设计（论文）《LDE 算法在 WiFi 室内定位中的应用研究》，是本人在导师指导下独立进行研究工作所取得的成果。对本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，其它未注明部分不包含他人已发表或撰写过的研究成果，不存在购买、由他人代写、剽窃和伪造数据等作假行为。

本人愿为此声明承担法律责任。

作者签名：

日期： 年 月 日

致 谢

在论文的最后，首先要衷心地感谢我的导师马琳老师，在本论文完成期间，马老师给我提供了完成论文的一切条件。

我要衷心地感谢马琳老师给我的耐心指导和无微不至的关怀，在论文完成期间，马琳老师给了我很多建议和帮助，尤其是在课题遇到重大困难的时候，他总能给我带来最及时的指导和建议；并且在本论文的完成过程中，马老师耐心地帮助我分析论文结构，并对论文做了修改，如果没有马老师的帮助、关怀，本论文是不可能完成的。在这里，再一次对马老师的帮助表示最真诚的感谢。

感谢家人在生活及学习上给予的大力支持，正是有了父母无私的关怀及家人不时地鼓励，让我能够在学习道路走得坚定顺利。此外，还有实验室的师兄们，一起做课题的同学们，以及同窗、好友们，在论文完成期间，他们给了我很多很好的建议，开阔我的思路，在这里，一并表示感谢。

最后要感谢项目组的老师和同学们，在项目组的例会讨论中给我的宝贵建议，令我开阔思路，更好的完成了论文。

附录 一

室内位置估计的自适应的时变无线电地图

尹杰，杨强，倪明选

计算机科学，香港科学技术大学

清水湾，九龙，香港，中国

{ yinjie, qyang, ni } @cs.ust.hk

摘要

本文提出一种新的方法：通过用数据挖掘和参考点的数据分析来抵消由于环境因素造成的无线电地图的时变特征来形成室内定位的自适应时变的无线电地图。由于环境变化导致在同一位置的无线信号的不同时刻的测试值都不同，给基于 802.11b 无线局域网的结构下实现室内位置估计方法带来了挑战。在这种环境变化的条件，某一时期内测试的无线电地图不能应用于其它时刻的室内位置估计。为了解决这个问题，本文采用回归分析的方法来测试的稀疏参考点与移动终端接收到的无线信号之间的关系。然后再应用回归分析方法实现采用接收到的新信号强度值估计在线阶段的用户的室内位置。通过实验证明这种方法能有效地适应信号强度随着时间的变化而不需要反复建立无线电地图。同时，本文也证实了这种方法能在极稀疏参考点条件下估计移动终端的位置。

1 概述

在现今从文字传输到监控都广泛应用电脑的生活中，位置估计是一项重要的任务。在室内环境中，越来越多的研究者关注采用便宜且应用广泛的 802.11 无线局域网作为基础设施实现室内位置估计。现今，很多室内定位系统采用接入点接收到信号强度值来估计移动终端的位置，包括：确定性方法和概率型方法。

通常，采用射频信号值进行定位的系统工作在两个阶段：离线训练阶段和在线定位阶段。在离线阶段，测试定位区域内的参考点的接收到的接入点的信号强度值构造无线电地图；在线阶段中，采用确定型定位模型或统计模型分析无线电地图来估计室内位置。在在线定位阶段，根据已有的无线电地图的特征，结合实时接收到的接入点的信号强度值来估计移动终端的位置。

在已有的研究中，无线电地图都被假设成是静态的，即一次测试的无线电地图被应用于之后的室内位置估计而没有更新不同时刻的无线电地图。这种简化方法导致了严重的问题。在多变的环境下，由于不可预测的人为移动、物体变化、无

线干扰及信号传输，在线阶段接收的接入点的无线信号可能大大地偏离了无线电地图存储的无线信号。因而，采用静态无线电地图估计的室内位置粗糙而不准确。即使如文献 11 中有关于 RFID 网络所述，采用增加测试点方法来消除环境带来的影响，只有当参考点的密度足够大时才能保证定位的准确度，但是因此导致无线电地图的构造的工作大幅度增加和室内位置估计所需的时间。相反，稀疏地参考点并不能解决上述问题。因此，设计一个准确而且考虑到动态环境的变化的位置估计系统是一项极具挑战性的任务。

本文提出一种新的方法：通过用数据挖掘和参考点的数据分析来抵消由于环境因素造成的无线电地图的时变特征来形成室内定位的自适应时变的无线电地图。图 1 中表述了本文提出的方法的结构图。与之前研究者的方法一样，首先，在 t_0 时刻构成系统地无线电地图；在下一个时刻 t_i ， $i > 1$ ，并不重复地构造无线电地图，而是在参考点

上放置一个射频信号接收器。基于参考点处接收到的信号强度值采用回归分析方法来修正静态的无线电地图的信号值。本文提出的定位方法是：根据回归分析无线电地图的特点，结合在线接收到的实时无线信号强度来估计移动终端的位置。在线定位阶段，采用回归模型来估计移动终端的室内位置。根据已有文献，最大似然估计是最好的位置算法之一。实验证明，采用回归模型的最大似然估计方法比原始的最大似然估计方法的定位精度提高了 15%（误差半径为 1.5m）。此外，本文提出的方法不需要已知参考点的物理位置。因此，与之前的 LANDMARC 系统相比，本文的方法在稀疏参考点条件下仍然可以估计移动终端的位置。

本文的创新之处：

- (1) 与已有的静态无线电地图的定位方法相比，本文提出的方法能很好地适应由于环境变化导致的信号强度值的变化；
- (2) 通过采用回归模型分析参考点的信号及移动终端接收到的实时无线信号的关系，因而参考点可以稀疏地分布在定位区域内。

本文剩余的部分按如下结构撰写。第二部分讨论采用 RF 信号确定型定位方法；第三部分概述本文问题的范畴；第四部分提出本文采用的算法；第五部分描述实验结果；第六部分给出结论及未来工作。

2 相关工作概述

许多重要的采用 RF 信号进行室内位置估计的系统已经建成。几乎所有前人研究室内位置估计的方法可以分为两类：确定型方法及概率型方法。确定型方法采用确定性关系估计客户端的位置。例如，微软研究院的 RADAR 系统采用最邻近

点和三角定位法来估计移动终端的位置。每一个信号强度均与无线电地图及相应坐标进行匹配，采用最为匹配的信号值的坐标的平均值来估计用户的位置。而 RADAR 系统的在 3 米内的定位精度为 50%。在文献[3]提出采用在线阶段的用户反馈来修正位置的估计。

概率型方法在不同位置构造无线电地图并采用概率型方法来估计位置。在机器位置感知系统中，采用贝叶斯概率算法及条件概率来计算某一位置接收到的信号强度的概率。在后续的处理中，限制用户可能运动的位置来优化位置估计及去除位置产生急剧变化的位置估计。采用后续处理与没有采用后续处理的算法在 1.5 米内的定位精度分别为 83%和 77%。同样，Youseff 等采用聚类算法来将位置分组从而减少计算量。该方法先采用方法来估计移动终端最有可能所在的聚类，然后再应用是最大似然估计方法来确定移动终端在聚类中的哪个位置。文献[17]中分析连续位置采集的信号的相关性。通过采集信号序列来提高相应的定位精度。几乎所有上述工作都基于一个假设：离线阶段的信号强度与在线阶段的信号特征没有大的变化。这一假设的主要缺陷就是丢失了信号传播过程的特征和环境的特征。而环境特征会使得在线阶段测试得到的信号偏离无线电地图中的信号强度，因此这一假设限制了定位精度的提高。

另一个相关工作是基于 RFID 技术的 LANDMARC 定位系统。LANDMARC 系统采用参考标记的方法来减轻 RF 信号的波动。该方法先依次计算跟踪标记的接收信号空间与参考标记的信号空间之间的距离，然后采用 k 个最近信号空间的参考标记的坐标来估计跟踪标记的坐标。然而，LANDMARC 系统定位精度的保障来自于高密度的参考标记。采用上述方法的定位时，在每平方米 1 个参考标记时定位误差在 1 米到 2 米之间。然而，在许多定位系统中，设备无法承受高密度参考标记的计算量。此外，RFID 接收设备很昂贵，不适合不范围的布置。而本文提出的自适应系统采用 802.11b 无线局域网，这种网络广泛存在且价格不贵。更重要的是，本文提出的方法能在低密度的参考点条件下确定移动终端的位置。此外，本文采用 RF 无线定位系统与 LEASE 系统相近，采用少量的发射器与接收器来辅助室内位置估计。在本文中，每个探测器都有一个生成模型：根据每个发射器的固定位置及接收到的相应信号强度值来估计网格点上的信号强度值。LEASE 系统由作者在两种不同的仿真条件下测试。然而，本文的主要工作证明本文提出方法在不同时刻的动态环境下的自适应性。此外，与文献[19]相比，本文提出的方法不需要已知参考点的物理位置。

3 无线环境

在这一部分，描述了实验环境的设置及分析导致无线定位具有挑战性无线信道

的噪声特征。

3.1 实验环境的设置

实验在 802.11b 无线以太网的实际环境中实施。关于实验环境细节在 5.1 中讨论。实验条件设置如下：（1）定位区域内的参考点的数目已知，但不必要知道参考点的物理位置；（2）定位区域内可检测到的接入点的数目已知，但其接入层不必要知道；（3）实验在二维环境下进行，但易于推广到三维条件下。

在 XP 操作系统下，安装一个 API 用于检测所有接入点的信号强度值以及其相应的物理地址与信道。采用这个 API，可以同时检测到不同接入点的信号强度值。由于参考点和客户端都可以与 802.11b 网络进行通信，所有接入点的信息是由特别地程序首次发送。当信息被服务器接收后，每个参考点的信号强度会被打包发送到客户端。在线定位是由移动终端完成。

3.2 噪声特征

电子电气工程师协会 802.11b 标准采用 2.4GHz，由于其自由使用，因此，802.11b 协议被广泛应用。但是，它也本质的缺点。在 2.4GHz，微波炉、蓝牙、2.4GHz 无绳电话会是该无线信号的干扰。此外，2.4GHz 是一种与能发生共振的频点，人体也会吸收其无线信号。

由于物体及人体的反射、折射、散射以及吸收，室内的信号传播产生了严重的多径衰落现象。因此，信号可以由不同的路径到达接收器而且具体不同的幅度与相位。这种不同的组合会导致原始信号产生新变化。除此之外，室内环境，如温度和湿度的变化会极大地影响信号的传播。因此，同一位置接收到的同一接入点的无线信号的强度值随着时间与建筑物的变化而不同。

图 2 给出了一个典型的示例：三个归一化后的同一位置同一接入点不同时刻的信号强度值变化的柱状图。为了绘制柱状图，每个测试点在 45 秒内采集了 450 个样本。由图中，可以明显地看出，同一位置接收到同一接入点的信号强度值发生强烈地变化。已有的研究给出证实，对柱状图的数据直接使用，而不是将数据转化为平均值。这样处理的前提是在训练过程中，信号强度值并没有发生太大的变化。然而，如图 2 所示，在实际中，在一段时间内信号发生强烈地变化。从上图中，可以得出，如果直接采用相应的数据进行位置估计，则在线阶段的位置估计可能会产生较大地偏差，因为离线阶段和在线阶段测试到的数据有很大的偏差。这一点是本文提出采用一定的方法来抵消由于环境变化带来的信号强度的变化。

4 理论分析

首先，定义位置状态空间 L 为地面上的 n 个网络点构成的一个集合。 L 表示为：

$$L = \{l_1 = (x_1, y_1, \theta_1), \dots, l_n = (x_n, y_n, \theta_n)\},$$

其中： $l_i = (x_i, y_i, \theta_i), 1 \leq i \leq n$ 表示第 i 个终端的位置和方位。需要注意的，本文采用二维方式来分析算法，但该算法可以推广到三维情况。

假设在定位环境中可以检测到 p 个接入点。由移动终端接收到的信号强度表示为： $s = \{s_1, s_2, s_3, \dots, s_p\}, 1 \leq j \leq p$ ，其中 s_j 表示第 j 个接入点的接收到的信号强度值。如果信号强度值过小，如小于 -95dBm ，则认为该接入点的信号为 0。假定定位区域内有 m 个参考点，将第 k 个参考点接收到的信号强度向量表示为： $r_k = \{r_{k1}, r_{k2}, r_{k3}, \dots, r_{kp}\}$ ，其中 $r_{kj}, 1 \leq k \leq m, 1 \leq j \leq p$ 表示第 k 个参考点接收到的第 j 个接入点的信号强度值。如 3.1 中所述，对于位置 l_i ，移动终端可以得到一个 m 个向量 r 和 m 个参考点的信号值。

本文的目标是采用自适应的方式根据参考点来确定移动用户的室内位置，关键问题是怎样找到参考点接收到的信号强度值与移动终端在不同时刻接收到的信号强度值之间的相关性。为解决这一问题，本文提出以下两个解决方法：

- (1) 在离线阶段，即对应于 t_0 时刻，采用回归分析方法来预测每一个跟踪位置上的参考点与移动终端间的接收到的信号强度之间的关系。首先，对于位置 $l_i, 1 \leq i \leq n$ ，其中 i 表示第 i 个移动终端的位置，通过回归分析得到 f_{ij} ，其中 j 表示第 j 个接入点。因此 f_{ij} 表示第 k 个参考点的接收信号强度值与第 i 个移动终端到的第 j 个接入点的信号强度值之间的关系。由此，可以得出回归分析函数。在下面的两小节中将讨论该函数。

$$s_j(t_0) = f_{ij}(r_{1j}(t_0), r_{2j}(t_0), \dots, r_{mj}(t_0)), 1 \leq i \leq n, 1 \leq j \leq p.$$

虽然这个函数是分析 t_0 时刻的无线电地图得到的，函数 f 揭示了参考点接收到的信号强度值与移动终端接收到的信号强度值的本质关系。

- (2) 在线定位阶段时刻 t ，根据参考点上接收到的信号向量计算信号估计向量 $s_{est}(t) = (s_1(t), s_2(t), \dots, s_p(t))$ ，即根据函数得到的每个参考点的接收信号强度值。在本文中，用 $s_{est}(t)$ 表示根据 f_{ij} 得到估计的接收到的信号强度值；用 ss_{act} 表示移动终端实际接收到的信号强度值的向量。然后采用下式计算 $s_{est}(t)$ 与 ss_{act} 之间的欧几里得距离：

$$D_i(t) = \sqrt{\sum_{j=1}^p (s_j(t) - ss_j(t))^2}$$

最后，最小化的 D_i 的信号估计的位置即为 l_i 的估计位置。

由于参考点的作用与移动终端的作用是可以互换的，因此，参考点上的新观测到的信号强度值可以用于位置区域的无线电地图的实时更新。因此，这种方法更能自适应环境的动态变化。然后，为了达到较高的定位精度，关键是找到信号强度之间的回归模型。在下面的两小节中，本文讨论两种计算 f_{ij} 的算法。

4.1 多变量回归分析

本文中，首先采用多变量回归方式来计算 f_{ij} 。多变量回归分析是采用多变量方式来计算数据之间的线性回归函数。

根据多变量回归模型，对于每一个接入点，在每一个位置上可以将接收到的信号强度表示为 m 个参考点接收到的相应的接入点的信号强度值的线性组合，即：

$$s_j(t) = \alpha_{0j} + \alpha_{1j}r_{1j} + \dots + \alpha_{mj}r_{mj} + \varepsilon_j.$$

在公式中， s_j 表示移动终端接收到的第 j 个接入点信号强度值， $r_{kj}, 1 \leq k \leq m$ 表示第 k 参考点接收到的相应的接入点的信号强度值。回归系数 α_{kj} 表示每一个参考点接收到的相应的接入点的信号强度值对移动终端接收到的相应的相应接入点的信号强度值的影响。如果所有参考点上接入到的相应的接入点的信号强度值均为 0，则回归系数相关。此外， ε_j 表示随机误差，通常假设其分布为零均值及方差为 σ^2 的正态分布。

1. 多变量回归模型的离线学习：在离线阶段，采用最小方差估计方法计算依次计算 p 个接入点回归系数，表示为 $\alpha_j = (\alpha_{0j}, \alpha_{1j}, \dots, \alpha_{mj})^T$ 。特别地，对定位区域内，在 m 个参考点连续采集 q 个信号强度值。本文假设 $q > m+1$ ，因此可以估计 $m+1$ 个回归系数。然后采用最小方差估计算法计算回归系数。因此，在每个位置，可以得到一个回归系数向量， α_j 表示第 j 个接入点的回归系数。
2. 多变量回归模型的在线应用：在线阶段，根据参考点上接收到的信号强度值，以及回归系数来估计每个地点可能接收到的信号强度向量，最后，预测的终端的位置为误差距离最小的参考点的位置。
3. 在线定位的时间复杂度分析：在线阶段应用多变量回归分析算法，其时间复杂度为 $O(pmn)$ ，与参考点的数量、移动终端的数目以及接入点的数目成线性关系。然而，一般情况下， m, n 是较小的整数，因此，在线定位的时间度与接入点数目成正比，且复杂度较低。

基于多变量回归的算法简单直接，但这种算法的有效性是基于移动终端与参考点的信号强度值之间为线性关系的假设上。在室内环境中，信号传播十分复杂，这个假设可能无法成立，因此需要更加合理的方法来解决这一问题。

4.2 决策树模型

在这一部分，本文提出一种基于决策树的通用非线性估计方法。决策树的每个叶结点都是基于二叉树线性回归方法。因此，可以在不知道线性关系的前提下来估计任何线性变量。图 3 所示为多变量回归分析与决策树之间的区别。正如图中所示，多变量回归是采用同一个线性关系来估计所有的点的回归系数，而决策树采用不同的线性关系来估计不同状态空间的回归系数。

对于每一个接入点都建立一个决策树来预测参考点与移动终端接收到的信号强度之间的关系。图 4 所示为四个参考点的条件下的决策树。图 4 中所示的结构与图 3 所示的状态空间图是等价关系。图中，每个内部结点对应于一个特殊的参考点接收到信号强度。2 棵子树都是同一个内部结点的分支。从根结点出发，根据接收到的信号强度来决定分支，最后到达叶结点。每个叶结点都对应于某一个状态空间的多变量回归模型。由此估计移动终端接收到某个接入点的信号强度值。

下面分析决策树的构造方法。决策树通常采用二元递归法来完成。递归过程是将所有样本二分，再对另一部分二分，由此形成决策树。最重要的是分类准则，去除采集信号不准确的部分。由于分类数可以连续预测，因此类的估计方差可以作为判定准则。在结点处最佳分类点是使最小期望方差，由下式计算：

$$V_{exp} = \frac{1}{N_L + N_R} (N_L \hat{\sigma}_L^2 + N_R \sigma_R^2).$$

其中 N_R, N_L 表示左右子树中的采样数，其和等总的采样数 q ，因此， $\hat{\sigma}_L^2$ 和 $\hat{\sigma}_R^2$ 两个子树方差，由下式计算：

$$\hat{\sigma}_L^2 = \frac{1}{N_L} \sum_{n \in L} (y_n - \mu_L)^2, \sigma_R^2 = \frac{1}{N_R} \sum_{n \in R} (y_n - \mu_R)^2.$$

其中 y_n 表示：每个训练样本中的类值， $\hat{\mu}_L$ 和 $\hat{\mu}_R$ 为左右子树类值的平均值。

根据最佳分类点的定义，决策树的构造算法如下所述：初始，所有训练样本全部放在根结点，对所有参考点应用所有可能的二分类。算法中选择的分类点为最小化左右子树的方差的点。然后对子树采用相同的方法进行分类。连续执行直到用户指定的最小结点数目停止。如果期望方差为零，则该结点被认为是叶结点，即使结点内参考点的数目没有达到用户指定的数目。然后采用算法对决策树进行修剪，用相应的线性回归函数替代合适的叶结点。

1. 决策树的离线学习：离线阶段，在每个位置用移动终端和 m 个参考点接收到的 q 个连续采样得到的信号强度值产生 p 个决策树。
2. 决策树的在线应用：在线阶段，对于每个接入点，根据参考点接收到信号强度值找到相应的叶结点。通过把线性模型加入到叶结点的合适位置来估计移动终

端的位置。通过这种方式，得到每个接入点的估计接收到的信号强度值。最后通过最小距离来预测移动终端的位置。

3. 在线时间复杂度分析：在线阶段应用决策树时，其时间复杂度为 $O(p'mn)$ ，其中 $p' \leq p$ 。因为根据决策树算法，通常选择较的参考点构成决策树，而不是所有的点。因为 m, p' 都是小的整数，所以在实验中，定位估计算法能够有效运行。

5 实验结果

在这一部分，首先讨论实验环境和数据采集过程。然后评估本文提出的算法的性能并与其它 WLAN 定位算法性能进行比较。

5.1 实验环境

实验在香港科学技术大学的计算机科学系学院大楼三楼的一部分区域进行。实验环境的布置如图 5 所示。实验中，选择八台 PC 机水平放置在走廊中，每台电台采用 Linksys-B USB 适配器作为参考点。图中用实点来表示。在实验环境中，存在九个接入点，在图中用箭头表示。另外四个接入点位于其它楼层或者实验区以外。区域内可见的接入点平均为 5~7 个。此外，采用 IBM 手提电脑作为移动终端来测试定位结果。

如图中放置的参考点所示，在上午 8 点到上午 12 点连续采集八个参考点上接收到的接入点的信号强度值，同时用 IBM 手提电脑在不同地点采集接入点的信号强度值。更具体的说，每个采集点相距 1.5 米，在每个位置采集 450 个样本点，每秒采集 10 个。因此，可以在一小时内得到九组数据。实验目标是测试本文提出的算法的自适应性。因此，用半夜 12 点测试的一组数据作为训练样本，其它各组作为测试点。

5.2 环境因素的影响

在本部分，本文评估了第四部分提出的多变量回归算法和决策树算法。特别地，用最大似然估计算法的定位结果相比。

表 1 所示为用三种定位方法在不同时段的总体定位精度。表中比较了三种方法在误差半径为 0.5 米、1 米和 1.5 米时的定位精度。对一进了大似然估计方法，每个位置在午夜 12 点采集 450 个样本点来作为训练的无线电地图，然后用其它数据作为测试点。相比这下，多变量回归算法和决策树算法则采用移动终端和参考点同时接收到的接入点的信号强度值来预测其关系。测试在不同时段进行。从表中可以看出，不同时段采用同样的定位算法的定位结果不一样，即表明实验结果具有环境多变性。对于本文提出的方法，1.5 米的定位精度达到 80%。因为在夜间，实验环境相对稳定，变化小。对于最大似然估计算法，在离线阶段得到的数据作

为定位阶段的实测数据。因此，这一部分的定位结果中，本文提出的方法和最大似然估计算法并不没太大的差别。

在白天，定位结果就不一样。多变量回归算法和决策树算法的定位结果明显优于最大似然估计算法。部分白天时段测试的定位结果在图 6 中表示。在图中，最大似然估计算法的定位精度在白天时候变化明显，而另外两种算法的定位精度相对稳定。更准确地说，对于最大似然估计算法，定位精度的方差为 0.0081，而另外两种算法的定位精度的方差分别为 0.0016 和 0.0012，大大低于最大似然估计算法的方差。这是因为在白天，定位区域的环境变化比夜间更加复杂。因此，最大似然估计算法的性能下降。相比这下，通过参考点来校正由于环境变化带来的信号强度的波动。从图中可以看，多变量回归算法和决策树算法的定位精度均高于最大似然估计算法。这也表明，对于多变量回归分析中的线性估计在复杂环境下不成立。在 1.5 米的定位精度条件下，决策树算法的定位精度平均为 76%，而最大似然估计算法的定位精度比其低 15%。

5.3 参考点的影响

在这一部分，本文讨论了参考点的放置及参考点数目对本文提出的算法性能的影响。从直观上看，参考点的放置和数目与现在的建模技术相关。对于多变量回归算法，采用 4.1 中的线性模型，因此至少需要两点，而且移动终端至少能看到两个参考点。对于 4.2 中所述的决策树算法，对于回归至少需要两个参考点，因此在本文中工程算法来解决参考点的放置和数目问题。在本文中，将定位区域分为四个方形区域，在每个区域至少有两个参考点。

图 7 比较了在不同参考点数目时两种定位算法的定位精度。对于给定的参考点数目 m ，平均定位精度定义为：每个子集内参考点的定位精度的平均。从图中可以看出，决策树算法的定位精度对参考点的数目不敏感。这是因为模型树选择最优的子集来进行估计与定位。然而，多变量回归算法的定位精度与参考点的数目相关。最优的参考点数目为 2 点。这是因为多变量回归算法总是选择一种方法来得到相应的线性拟合，因此两点时有最好的效果。对于多变量回归算法倾向于选择尽量少的参考点，而决策树算法一般选择 5~6 个参考点。考虑到系统设计，最困难的是在系统工作前选择合适的参考点的数目，因此决策树算法更具有可行性。

此外，从图 5 中可以粗略地得出 5 平方米内放置一个参考点。采用决策树算法在 1.5 米内定位精度为 76%。而对 LANDMARC 系统，每平方米需要一个参考点。因此，本文提出的方法更具可行性。

5.4 接入点的影响

在这一部分，本文研究接入点数目对提出的算法的定位精度的影响。

图 8 给出了在不同数目的接入点时，1.5 米内的平均定位精度。相似地，对于给定的数目的接入点 p ，定义平均定位精度：当接入点数目为 p 时的定位精度。从图中可以看出，当接入点的数目较小时，三种方法的定位精度都上升。这是因为当接入点的数目增加时，对于定位就有更多的信息可以使用。然而，当接入点的数目达到 6 或 7 时，对于某一定位方法的定位精度保持不变。这也表明，当有足够的信息来区分不同的位置时，增加接入点的数目并不能增加定位精度，而只是增加了计算复杂度。因此，粗略估计在定位区域内需要六个接入点。

6 结论与未来工作

本文提出了一种新颖的基于射频频的室内估计系统，这个系统能够自适应环境变化的影响。本文提出多变量回归算法和决策树算法。前一种算法基于信号强度之间为简单的线性关系的假设，而后者采用决策树方法生成非线性估计方法。实验表明，本文提出的算法的性能比最大似然估计算法好。此外，决策树算法的性能更加稳定且更具可行性。对于提出的算法，其所需的接入点的数目和参考点的数目已知，而且没有必要知道参考点和接入点的物理位置。本文提出的算法能低密度参考点的条件下估计移动终端的位置。

本文的工作可以向以下几个方面推广。一，采用更有效的概率方法来建立无线电地图，二，希望能引入用户的移动跟踪来提高定位精度。此外，需要测试本文提出算法在大规模条件下的合理性。

注：对于文献中的致谢和参考文献部分仅给出原文，并没有翻译。

附录 二

Adaptive Temporal Radio Maps for Indoor Location Estimation

Jie Yin, Qiang Yang, Lionel Ni

Department of Computer Science

Hong Kong University of Science and Technology

Clearwater Bay, Kowloon, Hong Kong, China

{yinjie, qyang, ni}@cs.ust.hk

Abstract

In this paper, we present a novel method to adapt the temporal radio maps for indoor location estimation by offsetting the variation environmental factors using data mining techniques and reference points. Environmental variations, which cause the signals to change from time to time even at the same location, present a challenging task for indoor location estimation in the IEEE 802.11b infrastructure. In such a dynamic environment, the radio maps obtained in one time period may not be applicable in other time periods. To solve this problem, we apply a regression analysis to learn the temporal predictive relationship between the signal-strength values received by sparsely located reference points and that received by the mobile device. This temporal prediction model can then be used for online localization based on the newly observed signal strength values at the client side and the reference points. We show that this technique can effectively accommodate the variations of signal-strength values over different time periods without the need to rebuild the radio maps repeatedly. We also show that the location of mobile device can be accurately determined using this technique with lower density in the distribution of the reference points.

1 Introduction

Location estimation is an important task in today's pervasive computing applications that range from context dependent content delivery to people monitoring. In an indoor environment, increasing attention is paid to location estimation using the inexpensive and popular IEEE 802.11b wireless networks as the fundamental infrastructure. Many systems utilize the signal-strength values received from the access points to infer the location of mobile device, based on deterministic or probabilistic techniques

In general, location-estimation systems using radio frequency (RF) signal strength function in two phases: an offline training phase and an online localization phase. In the offline phase, a radio map is built by tabulating the signal strength values received from

the access points at selected locations in the area of interest. These values comprise a radio map of the physical region, which is compiled into a deterministic or statistical prediction model for the online phase. In the online localization phase, the real-time signal strength samples received from the access points are used to search the radio map to estimate the current location based on the learned model.

In most of the previous work, the radio maps are assumed to be static, which means that once learned in the offline phase, a radio map is applied thereafter to estimate the various locations in later time periods without adaptation. This simplistic approach poses a serious problem. In a dynamic environment caused by the unpredictable movements of people, layout changes, radio interference and signal propagation, the signal-strength samples measured in the online phase may significantly deviate from those stored in the radio map. As a result, location estimation based on a static radio map may be grossly inaccurate. Even if we attempt to deal with the variations of signal strength using additional reference points, as is done in [11] with RFID networks, the accuracy can be guaranteed only when the reference points are densely distributed, which drives up the cost of hardware and increases the potential computational time of location estimation. Conversely, a sparse distribution may not solve the problem satisfactorily. Therefore, it is a challenging task to design a location-estimation system which is both accurate and robust with respect to dynamic environmental changes.

In this paper, we present a novel method to adapt the radio map along the time dimension by offsetting the environmental dynamics using a regression analysis. Figure 1 illustrates the idea behind our proposed method. As in previous work, we start by collecting data to construct a static radio map in time period t_0 . In any later time period t_i , where $i > 1$, instead of rebuilding the radio maps repeatedly, we place radio frequency (RF) receivers which act as dynamic reference points in the environment. Based on the signal-strength values received by the reference points, we apply a regression analysis to obtain the estimated radio maps which comprise the corrections we need to make to the static radio map. In our approach, the static radio map is compiled into regression models that predict the client locations using both the signal-strength values collected by the mobile client and those collected by the reference points. During the localization phase, the regression models are used to predict the most likely location of the mobile client. To the best of our knowledge, the Maximum Likelihood (ML) method is one of the best approaches to location estimation. We show that, using adaptive temporal maps through reference points, our approach can gain a higher average accuracy of localization over different time periods, which is 15% higher

than the ML method (within 1.5 meters). In addition, our approach does not require that the physical positions of the reference points are known. Therefore, the location of mobile device can be accurately determined even with a lower density of reference points as compared with previous reference-point based methods such as the LANDMARC system.

The novelty of our work can be summarized as follows:

- Compared with previous static radio map-based techniques, our proposed method can adapt well to the variations of signal-strength values caused by the environmental dynamics.
- By capturing the dynamic relationship between signal strength values received by the reference points and that received by the mobile device using regression models, the reference points can be sparsely distributed in the environment.

The rest of the paper is organized as follows. Section 2 discusses related work on location-determination systems using RF signal strength. Section 3 provides an overview of the problem domain. Section 4 presents the algorithms used in our analysis. Section 5 describes the experimental results. Section 6 concludes the paper and discusses directions for future work.

2 Related Works

Significant work has been done in the area of location estimation using RF signal strength. Most of the previous work are based on the techniques of using radio map, which can be classified into two broad categories: deterministic techniques and probabilistic techniques. Deterministic techniques apply deterministic inference methods to estimate a client's location. For example, the RADAR system by Microsoft Research uses nearest neighbor heuristics and triangulation methods to infer a user's location. Each signal-strength sample is compared against the radio map and the coordinates of the best matches are averaged to give the location estimation. The accuracy of RADAR is about three meters with fifty percent probability. In [3], an online procedure based on feedback from users was employed to correct the location estimation of the system.

Probabilistic techniques construct the signal-strength distributions over different locations in the radio map and use probabilistic inference methods for localization. The robotics-based location sensing system in applies Bayesian inference to compute the conditional probabilities over locations based on received signal strength samples from

various access points. Then a post processing step, which utilizes the spatial constraints of a user's movement trajectories, is used to refine the location estimation and to reject the estimates showing significant changes in the location space. Depending on whether the post processing step is used or not, the accuracy of this method is 83% or 77% within 1.5 meters respectively. Likewise, Youssef et al. used a joint clustering technique to group locations together to reduce the computational cost of the system. The method first determines a most likely cluster within which to search for the most probable location, and then applies a Maximum Likelihood (ML) method to estimate the most probable location within the cluster. A time-series analysis technique is introduced in [17] to study the correlation among consecutive samples received from the same access point over time. In this way, higher accuracy is obtained by taking the information about sample sequences into account. Most of the above works are all based on a common assumption that the radio map collected in the offline phase does not change much later in the online phase. A major limitation with this assumption stems from the dynamic characteristics of signal propagation and the environment, where the signal-strength values measured in the online phase can significantly deviate from those that are stored in the radio map, thereby limiting the accuracy of such systems.

Another related work is the LANDMARC system which is based on the RFID technology. LANDMARC utilizes the concept of reference tags to alleviate the effects caused by the fluctuation in RF signal strength. The method first computes the distance between the signal-strength vector received from the tracking tag and those from different reference tags respectively. It then uses k nearest reference tags' coordinates to calculate the approximate coordinate of the tracking tag. However, the accuracy of LANDMARC can be guaranteed only when the reference tags are densely distributed. The authors report that one reference tag is needed for each square meter to accurately locate the objects within the error distance between one and two meters. However, in many location-based applications, the deployment of a dense infrastructure for location estimation is not feasible. Moreover, the RFID readers are expensive, making them cost-prohibitive for localization in a large area. In contrast, our adaptive system utilizes the IEEE 802.11b wireless network which is already widely available and are relatively inexpensive. More importantly, the location of mobile devices can be determined even with a lower density of the reference points. In addition, our work is related to the LEASE system, which employs a few stationary emitters and sniffers to assist location estimation for indoor RF wireless networks. In this work, a synthetic model is generated for each sniffer, which estimates the signal-strength value at each grid point based on the coordinates of the stationary emitters and the signal-strength values received from them. The authors evaluate the performance of LEASE in two different experimental

test-beds. However, the focus of our work is mainly to demonstrate the adaptivity of our proposed method to dynamic environmental changes over different time periods. Moreover, our approach, as compared with [9], does not require that the physical positions of reference points are known.

3 Wireless Environment

In this section, we describe our experimental setup and the noisy characteristics of wireless channel which makes the problem of location determination a challenging task.

3.1 Experimental Setup

Our experiments were conducted in a real environment which is equipped with an IEEE 802.11b wireless Ethernet network. We will discuss the experimental test-bed in detail in Section 5.1. The conditions of our experimental setup are as follows: (1) The number of reference points is known while the physical positions of reference points are not necessarily given. (2) The number of access points that can be detected in the environment is known, but we need not know the layout of the access points. (3) We performed the experiment in a two-dimensional location space, but it can be easily extended to a three-dimensional location space.

We developed a wireless API under the Window XP operating system to record the signal-strength values from all detectable access points along with their MAC addresses using the mode of active scanning. Using this API, the mobile client and reference points can receive the signal-strength values from the access points simultaneously. As the mobile client and reference points have the capabilities of communicating with the Internet using IEEE 802.11b wireless network, all the information received from the access points is first sent over to a specific program running on the location server. After the information is received by the server, the signal-strength values received by each reference point are packaged and transmitted to the mobile client via a wireless network socket. The location computation is done in an online manner by the mobile client.

3.2 Noisy Characteristics

The IEEE 802.11b standard uses radio frequencies in the 2.4 GHz band, which is attractive because it is license free in most places around the world. However, it does suffer from inherent disadvantages. In the 2.4 GHz band, microwave ovens, Blue Tooth devices, 2.4 GHz cordless phones and other devices can be sources of interference. Moreover, 2.4 GHz is the resonant frequency of water and human bodies can absorb RF signal strength.

Subject to reflection, refraction, diffraction and absorption by structures and humans, signal propagation suffers from severe multi-path fading effects in an indoor environment. As a result, a transmitted signal can reach the receiver through different paths, each having its own amplitude and phase. These different components are combined to reproduce a distorted version of the original signal. Moreover, changes in the environmental conditions such as the change in temperature and humidity affect the signals to a large extent. As a result, the signal-strength values received from an access point at a fixed location varies with different time periods as well as physical surroundings.

Figure 2 gives a typical example of three normalized histograms of the signal-strength values received from an access point at a fixed location over different time periods. To build each particular histogram, 450 samples were taken in about 45 seconds over different time periods. It is clear that the signal-strength values received from the same access point varies with time even at a fixed location. Previous work [10, 18, 19] showed that it would be better to directly use these histograms rather than reduce the data into average values. By doing this, the essential assumption is that the histograms constructed in the training phase does not change much over time. However, in reality, as shown in Figure 2, the signal-strength histograms vary noticeably over different time periods, with significantly higher noise levels when more people are moving in the building. These variations suggest that, depending on the signal-strength histograms trained in the offline phase, the results of location estimation might be inaccurate if the signal-strength samples measured in the online phase deviate significantly from those collected in the offline phase. This motivates us to make use of reference points to adaptively offset the environmental dynamics that cause the variations in signal strength.

4 Methodology

We first define the location-state space L as a set of n physical grid points on the floor map. L is denoted as $L = \{l_1 = (x_1, y_1, \theta_1), \dots, l_n = (x_n, y_n, \theta_n)\}$, where each tuple $(x_i, y_i, \theta_i), i \in [1, n]$, represents a mobile user's location and orientation. Note that we describe the algorithm in a two-dimensional location space, but our algorithms can be easily extended for three-dimensional localization.

Suppose that there are p access points that can be detected in the environment. The

signal-strength vector received by a mobile device is defined as $s = (s_1, \dots, s_p)$, where $s_j, j \in [1, p]$, represents the signal-strength value received from the j th access point. Note that if the signal strength value of an access point is too weak to be detected by the mobile device, we assign s_j with a small signal strength value, e.g., -95 dBm. Suppose that there are m reference points placed in the environment. We define the signal-strength vector received by the k th reference point as $r_k = (r_{k1}, \dots, r_{kp})$, where $r_{kj}, k \in [1, m], j \in [1, p]$, represents the signal-strength value received by the k th reference point from the j th access point. As described in Section 3.1, for each location l_i , we can obtain the signal strength vector s received by a mobile user at this location, along with m signal-strength vectors r received by m reference points in the same time period.

Since our objective is to determine the location of a mobile user using reference points in an adaptive way, the vital issue is how to correlate the signal-strength values received by the reference points with that of the mobile device over different time periods. For this purpose, our proposed approach works in two phases:

(1) During the offline phase, which corresponds to time period t_0 , we apply a regression analysis to learn the predictive relationship of signal-strength values between the reference points and the mobile device which is tracked at each selected location. First, if we consider a location to be $l_i, i \in [1, n]$, where for the j th access point, $j \in [1, p]$, we learn the corresponding relationship f_{ij} . Here f_{ij} indicates the relationship between the signal-strength values $r_{kj}(t_0)$ received by each of the k reference points, $k \in [1, m]$, and the value received by the mobile device $s_j(t_0)$. In particular, we build a regression relationship using the following function, which we will discuss in detail in the next two subsections: $s_j(t_0) = f_{ij}(r_{1j}(t_0), r_{2j}(t_0), \dots, r_{mj}(t_0))$, $k, i \in [1, m], j \in [1, p]$. While this function is learned in time period t_0 , the functional relationship inherent in f captures the dynamic relationship between the signal-strength values received by reference points and the predicted signal strength value received by the mobile device at each location.

(2) During the online phase in time period t , based on the signal-strength vectors received from the reference points, we compute the estimated signal-strength vector $s_{est} = (s_1(t), \dots, s_p(t))$ that may be received at each location using the corresponding

function f_{ij} . We refer to the signal-strength vector that is computed using the function f_{ij} as an estimated signal-strength vector s_{est} and the signal-strength vector that is actually received by the mobile device as an actual signal-strength vector as s_{act} . Then for each location l_i , we compute the Euclidian distance $D_i(t)$ between its corresponding estimated signal-strength vector $s_{est} = (s_1(t), \dots, s_p(t))$ and the actual signal-strength vector $s_{act} = (s_1(t), \dots, s_p(t))$ as follows:

$$D_i(t) = \sqrt{\sum_{j=1}^p (s_j(t) - s_{s_j}(t))^2}$$

Finally, a location li is predicted to be the most probable location if its corresponding distance $D_i(t)$ is minimized. Since the reference points are subject to the same effect in the environment as the tracked mobile device, the newly observed signal-strength values received by the reference points can be used to dynamically update the information for localization in real time. Therefore, this approach is more flexible and adaptive to the environmental dynamics. However, to achieve high accuracy, the critical issue is how to model the relationship between the signal-strength values received by the reference points and that received by the tracked mobile device during the offline phase. In the next two subsections, we will discuss two different algorithms to learn the function f_{ij} .

4.1 Multiple Regressions

Our first attempt is to apply multiple regression to model the relationship of signal-strength values between reference points and the mobile device. Multiple regression is a generalization of simple linear regression which allows for the modeling of the relationship between a dependent variable and more than one independent variable. Based on the multiple regression model, at each location, for each access point, we compute the signal-strength value received by the mobile device as a linear aggregate of the signal-strength values received by m reference points, as follows:

$$s_j = \alpha_{0j} + \alpha_{1j}r_{1j} + \alpha_{2j}r_{2j} + \dots + \alpha_{mj}r_{mj} + \varepsilon_j$$

In this equation, s_j represents the signal-strength value received by the mobile device from the j th access point and r_{kj} , $k \in [1, m]$, represents the corresponding signal strength value received by the k th reference point from the j th access point. The regression coefficients α_{kj} , represent the independent contributions of each reference point to the prediction of signal-strength value received by the mobile device. When all the r 's are equal to 0, α_{0j} is called the intercept. In addition, ε_j is the random error, which is

usually assumed to be normally distributed with mean zeros and variance σ^2 . 1. **Offline Learning of Multiple Regression Model:** During the offline phase, we perform the least square estimation method to compute the regression coefficients $\alpha_j = (\alpha_{0j}, \alpha_{1j}, \alpha_{2j}, \dots, \alpha_{mj})^T$ for p access points respectively. Specifically, we collected a series of q signal-strength samples received by the mobile device and m reference points simultaneously at each location. Note that we assume $q > m + 1$ so that for the j th access point, we have q linear equations, which are more than $(m + 1)$ parameters to be estimated, $\alpha_j = (\alpha_{0j}, \alpha_{1j}, \alpha_{2j}, \dots, \alpha_{mj})$. Then the solutions to these linear equations provide the least squares estimates of the coefficients. Therefore, at each location, we can obtain a set of regression coefficients, $\alpha_j = (\alpha_{0j}, \alpha_{1j}, \alpha_{2j}, \dots, \alpha_{mj})$, where α_j corresponds to the j th access point.

2. Online Application of Multiple Regression Model:

During the online phase, based on the signal-strength values received by the reference points, the regression coefficients can be used to calculate the estimated signal-strength vectors s_{est} for each location. Finally, the location with the smallest distance between s_{est} and ss_{act} is selected as the final predicted location.

3. Analysis on Online Time Complexity:

When we apply multiple regression model during the online phase, the time complexity is $O(pmn)$, which is linear with the number of locations n , the number of reference points m and the number of access points p . However, in most cases m and p are small integers. In our experiment, $m \in [1, 8]$; $p \in [1, 9]$, therefore, the location estimation can be done efficiently. The multiple-regression based algorithm is simple and straightforward; however, it assumes that the relationship of signal-strength values between the mobile device and reference points can be well approximated by a linear model. In an indoor environment where the signal propagation is quite complex, this assumption may not hold and more effective approaches are therefore desired.

4.2 Model Tree

In this section, we propose a general nonlinear approximation approach based on a model tree. A model tree is a binary decision tree with linear regression functions at the leaf nodes. Thus it can represent any piecewise linear approximation to an unknown function. Figure 3 illustrates the difference between a multiple regression and a model tree. As shown in the figure, a multiple regression uses a single linear model to fit the

whole reference-point value space while a model tree divides the whole state space into several regions, in each of which a different linear model is used for relating the signal-strength values received by reference points with the value received by the mobile client.

For each access point, we build a model tree to learn the predictive relationship of signal-strength values between reference points and the mobile device. As an example, Figure 4 shows a model tree which is built over four reference points (RP1 ~ RP4) to predict the signal-strength value received by the mobile device. Note that this tree structure is equivalent to the state-space structure in Figure 3(b). In the figure, each internal node corresponds to a test on the signal-strength value received by a particular reference point. Two sub-trees are branched from an internal node, each corresponding to a binary range of values. Starting from the root node, a test sample will be asked a sequence of questions until it reaches a leaf node. Each leaf node at the lowest level is attached with a linear regression function from which the estimated signal-strength value received by the mobile device can be calculated.

Now let us explain the construction process of a model tree. A model tree is built through a process known as binary recursive partitioning. This is an iterative process of splitting the samples into two partitions and then splitting each partition further into sub-trees. The vital part of the algorithm is the splitting criterion, derived from the measure of the impurity of a sample set. Since the class value to be predicted is continuous, the estimated variance of the class values is used as the impurity measure. The best splitting point of the samples in a node is chosen as the one that minimizes the expected variance V_{exp} , given by:

$$V_{\text{exp}} = \frac{1}{N_L + N_R} (N_L \sigma_L^2 + N_R \sigma_R^2)$$

Where N_L, N_R denote the number of samples falling into the left child node and the right child node. Accordingly, σ_L^2, σ_R^2 are the variance of predicted values at two children nodes respectively, computed by:

$$\sigma_L^2 = \frac{1}{N_L} \sum_{n \in L} (y_n - \mu_L)^2, \sigma_R^2 = \frac{1}{N_R} \sum_{n \in R} (y_n - \mu_R)^2$$

Where y_n is the class value of each training sample, and μ_L, μ_R are the means of the class values at left child node and the right child node.

Based on the definition of the best splitting point, the algorithm of building a model tree

works as follows: Initially, all of the training samples are placed together in the root node. The algorithm then tries breaking up the samples, using every possible binary split on every reference point. The algorithm chooses the splitting point that partitions the samples into two parts such that it minimizes the expected variances for each part. This splitting is then applied to each of the new branches. The process continues until each node reaches a user-specified minimum node size and becomes a leaf node. If the expected variance in a node is zero, then that node is considered a leaf node even if it has not reached the minimum size. Then the algorithm prunes the tree by replacing sub-trees with linear regression functions whenever this seems appropriate.

1. Offline Learning of Model Tree: During the offline phase, at each location, we use a series of q signal strength samples received by the mobile device and m reference points simultaneously to learn p different model trees, one for each access point.
2. Online Application of Model Tree: During the online phase, for each access point, we walk along the corresponding model tree until a leaf node is reached, based on the signal-strength received by reference points. Through the linear model attached to that leaf node, we calculate the estimated signal-strength received by the mobile device. In this way, for each location we obtain an estimated signal-strength vector s_{est} . Finally, the location with the smallest distance is predicted.
3. Analysis on Online Time Complexity: When we apply model tree for localization during the online phase, The Time complexity is $O(pmn)$, where $q < p$ because the model-tree based algorithm, instead of using all the reference points, always chooses an optimal subset of reference points to build the tree. Similarly, since m and p are small integers, the location estimation can be done efficiently in our experiment.

5 Experimental Results

In this section, we first discuss our experimental test-bed and the procedure for data collection. Then we evaluate the performance of our proposed algorithms and compare them with previous methods for indoor WLAN-based location estimation.

5.1 Experimental Test-bed

We conducted our experiment in a section of the third floor of the Academic Building where the Computer Science Department at Hong Kong University of Science and Technology is located. The layout of the experimental test bed is shown in Figure 5. In our experiments, we chose eight available PC machines along the horizontal hallway, each of which is equipped with a Linksys Wireless-B USB Network adapter, as the

reference points. The placement of reference points is marked with solid circles in the figure. In this environment, there are nine access points that can be detected, of which five access points distributed within this areas are marked with blank triangles in the figure. The other four access points are located either on the same floor outside this area or on the different floors. On average, the number of access points covering a location varies from five.

With the placement of the reference points shown in the figure, we repeatedly collected signal-strength samples received by the eight reference points from the access points on every other hour from early morning to midnight (8:00 AM – 12:00 AM). Within each hour in which data are continuously collected, we simultaneously used an IBM laptop computer to collect signal-strength samples at various positions in the horizontal hallway, along which reference points are placed. More specifically, we collected samples at the positions every 1.5 meters apart from one end of the hallway to the other facing both directions (each grid cell is 1.5 meters). At each position, we took 450 samples at ten samples per second. Thus we obtained nine groups of one-hour data. The objective of our experiment is to test the adaptive abilities of our proposed algorithms. Therefore, we used one group of data collected at midnight 12:00 AM for training and other independent groups of data for testing.

5.2 Impact of Environmental Factors

In this section, we evaluate the performance of the multiple regression algorithms and the model-tree based algorithm discussed in Section 4. In particular, we compare the two algorithms with the Maximum Likelihood (ML) method in with respect to their ability to adapt to the environmental factors.

Table 1 shows the overall accuracy using the three approaches over different time periods. As shown in the table, we compare the accuracy of three approaches within different distances: 0.5, 1.5 and 3 meters. In this experiment, for the ML method, we used 450 samples collected at each location at midnight 12:00 AM to train the radio map which was later used for testing over different time periods, as described in [10, 19]. In contrast, for the multiple regression and the model-tree based algorithms, we used signal-strength samples received by both the mobile device and reference points at the same time to learn the predictive relationships among them. The testing was also performed over different time periods. We can see from the table that the three approaches perform approximately the same at 10:00 PM, a quiet time in the department. For example, the accuracy within 1.5 meters is nearly 80%. This is because the

environmental conditions at night are relatively static. For the ML method, the radio map built in the training phase can accurately model the signal-strength samples observed in the localization phase in these quiet time periods. Therefore, in this part of the experiments, there is not much difference in accuracy between the ML methods with the two algorithms using reference points.

The situation is quite different during the daytime, when the multiple-regression and the model-tree based algorithms can be seen to outperform the ML method by a large margin. Part of the results with respect to the accuracy within 1.5 meters in the daytime periods are shown in Figure 6 for illustration. In the figure, the accuracy of the ML method varies a lot over different daytime periods, while the accuracy of multiple-regression and model-tree based algorithms are relatively stable. More specifically, for the ML method, the variance of accuracy over different daytime periods is 0.0081. For the other two algorithms using reference points, the variances of accuracy are 0.0016 and 0.0012 respectively, which are much lower than that of the ML method. This is because the environment during the daytime is much more complex than at night due to people moving, door opening and closing. This causes the signal strength samples measured during the daytime to significantly deviate from those in the radio map. Therefore, the performance of the ML method may decrease dramatically depending on the environmental dynamics. In contrast, by using reference points, both of our proposed algorithms can better adapt to the dynamics of environmental conditions. Moreover, we can see from the figure that the accuracy of the model-tree based algorithm is higher than that of the multiple-regression based algorithm. This shows that the linear assumption made by the multiple-regression algorithm may not hold in a complex indoor environment. The average accuracy of the model-tree based algorithm over different times is about 76% within 1.5 meters, which is increased by 15% as compared with the ML method.

5.3 Impact of Reference Points

In this section, we investigate the effect of the placement and number of reference points on the performance of our proposed algorithms. Intuitively, the placement and number of reference points are related to the technique used to build the model. For the multiple-regression based algorithm, the model is built using a linear function as described in Section 4.1; therefore, at least two points are needed for reasonable smoothing. This implies that a mobile device at any position should see at least two reference points. For the model-tree based algorithm, as described in Section 4.2, the model is built by first dividing the whole reference-point value space into sub regions

and then fitting a different linear function to each sub region. Similarly for each sub region, at least two reference points are needed. Therefore, we use an engineering solution to place the reference points in this paper. In our case, we divide the horizontal hallway into four sub squares with approximately equal area, in each of which at least two reference points are placed on two sides respectively along the hallway.

Figure 7 compares average accuracy within 1.5 meters using the two approaches with respect to different numbers of reference points. For a specific number of reference points m , we define average accuracy as the accuracy averaged on all the possible subsets of reference points corresponding to m . We can see from the figure that, the accuracy of the model-tree based algorithm is not very sensitive to the number of reference points. This is because the model-tree based algorithm always chooses an optimal subset of reference points to build the tree according to their capability in predicting the signal-strength value received by the mobile device, even if more reference points are provided. However, the accuracy of the multiple-regression based algorithm depends on the number of reference points to a large extent. An interesting observation is that, the best accuracy of the multiple-regression based algorithm is usually obtained when the number of reference points is two. This is because the multiple-regression based algorithm always finds a linear model to approximately fit the relationships between signal-strength values received by reference points and that received by the mobile device. However, in reality, such relationships are nonlinear. As a result, the multiple-regression based algorithm tends to select as few reference points as possible to construct an optimal linear model. In contrast, the best accuracy of the model tree based algorithm is achieved when five or six reference points are evenly distributed on two sides along the hallway. From the perspective of system design, it is difficult to specify the appropriate number of reference points before the system starts to work, therefore, the model-tree based algorithm is more feasible than the multiple-regression based algorithm since its performance is less sensitive to the placement and number of reference points.

In addition, as shown in Figure 5, we roughly place one reference point at about five square meters to locate the mobile device. The average accuracy is 76% within 1.5 meters using the model-tree based algorithm. However, in the LANDMARC system, one reference tag is needed per square meter such that the worst error is two meters and the average is about one meter. Therefore, it is more feasible to implement our proposed algorithm in many location-based applications.

5.4 Impact of Access Points

In this section, we study the effect of the number of access points p on the performance of the multiple-regression based algorithm, the model-tree based algorithm and the ML method.

Figure 8 shows the average accuracy within 1.5 meters using three approaches with respect to different numbers of access points at 12:00 PM. Similarly, for a specific number of access points p , we define average accuracy as the accuracy averaged on all the possible subsets of access points corresponding to p . We can see from the figure that, initially, the accuracy of three approaches increases as the number of access points increases. This is because when more access points are used, we have more information for localization. However, when the number of access points increases to six or seven, for a particular approach, the accuracy remains approximately the same. This shows that when we have enough information to distinguish different locations, the added access points do not contribute to an increase in accuracy but to an increase in the computational complexity. Therefore, we roughly need six access points to locate a mobile device in our environment. However, the model-tree based algorithm outperforms the other two approaches at different numbers of access points since it can adapt better to dynamic environmental changes.

6 Conclusions and Future Work

In this paper we have proposed a novel RF-based indoor location-estimation system which can adapt to dynamic environmental changes. We proposed a multiple-regression based algorithm and a model-tree based algorithm. While the former is based on a simple linear relationship between the signal-strength values received by the reference points and that received by the client device, the latter represents an improvement using a nonlinear function. Our experiments show that the proposed algorithms achieve a large advantage over the Maximum Likelihood method in terms of estimation accuracy by using adaptive temporal maps through reference points. Furthermore, we show that the model-tree based algorithm is much more robust with respect to reduction in the number of reference points. For the proposed algorithms, the number of reference points and the number of access points are known, but we need not know the physical positions of reference points and access points. In addition, the proposed algorithms can determine the locations of mobile devices even with a lower density of reference points.

Our work can be extended in several directions. First, we will consider applying more

effective probabilistic methods to build the radio map at each grid point using the signal strength values received by the reference points. Second, we wish to incorporate the user's movement trajectories to further improve the accuracy of location estimation. In addition, we also wish to test the validity of our proposed algorithms in a larger-scale environment.

- [1] Axel K. Location-based Services: Fundamentals and Operation[M]. New York: Wiley: 2005.
- [2] Kolodziej K.W., Hjelm J. Local Positioning Systems: LBS Applications and Services[M]. New York: Taylor & Francis: 2006.
- [3] Want R., Hopper A., Falco V. et al. The Active Badge Location System[J]. ACM Transactions on Information Systems[J]. 1992: 10(1): 91-102.
- [4] Hightower J., Borriello G. Location Systems for Ubiquitous Computing[J]. IEEE Computer, 2001: 34: 57-66.
- [5] Darrell T., Gordon G., Harville M., Woodfill J. Integrated Person Tracking Using Stereo Color, and Pattern Detection[J]. International Journal of Computer Vision, 2000: 175-185.
- [6] Bahl P., Padmanabhan V. N. RADAR: An In-building RF-based User Location and Tracking System[C]. In: INFOCOM 2000. NiNETEENTH Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings of IEEE. 2002: 775~784.
- [7] Bahl P., Padmanabhan V. N., Balachandran A. Enhancements to the RADAR User Location and Tracking System[R]. Microsoft Research Technical Report: MSR-TR-00-12, Fed., 2000.
- [8] Roos T., Myllymaki P., and Tirri H. et al. A Probabilistic Approach to WLAN User Location Estimation[J]. International Journal of Wireless Information Networks. 2002: (9) 155~164.
- [9] Martinez E. A., Cruz R., Favela J. Estimating User Location in a WLAN Using Back-propagation Neural Networks[J]. Advances in Artificial Intelligence-IBERAMIN 2004. 2004: 737~746.
- [10] Pan J.J., Kwok J. T., and Yang Q. et al. Multidimensional Vector Regression for accurate and Low-cost Location Estimation in Pervasive Computing[J]. IEEE Transactions on Knowledge and Data Engineering. 2006: 1181~1193.
- [11] Kushki A., Plataniotis K. N., Venetsanopoulos A. N. Kernel-based Positioning in Wireless Local Area Networks [J]. IEEE Transaction on Mobile Computing. 2007: 689~705.
- [12] Guvenç I., Abdallah C., and Jordan R. et al. Enhancements to RSS Based Indoor Tracking System Using Kalman Filter [C]. In: Proceedings of International Signal Processing Conference(ISPC), Dallas, TX, US, 2003.
- [13] Oreshkin B., Coates M. J.. Bootstrapping Particle Filters Using Kernel Recursive Least Square [C]. In: Proceedings of the IEEE Aerospace Conference, Big Sky, MT, USA, March, 2007.
- [14] Gentile C., Klein-Berndt L. Robust Location Using System Dynamics and Motion Constraints [C]. In: IEEE International Conference on Communications. 2004: 1360~1364.
- [15] Niculescu D., Nath B. VOR Base Stations for Indoor 802.11 Positioning [C]. In: Proceedings for the 10th Annual International Conference on Mobile Computing and Networking, 2004: 69.
- [16] Dartmouth College WLAN. <http://crowdad.cs.dartmouth.edu>.
- [17] Aruba Selected By Microsoft For Next Generation Wireless LAN. <http://www.arubanetworks.com/company/press/2005/06/13/>.
- [18] Jardosh A.P., Iannaccone G., Papagiannaki K., Vinnakota B. Towards an Energy-Star WLAN Infrastructure [J]. Eighth IEEE Workshop on Mobile Computing Systems and Applications. 2007: 85-90.
- [19] 刘兴堂, 周自全, 李为民, 孙德建. 现代导航与测控 [M]. 北京: 科学出版社. 2010: 283-284.
- [20] Maaten L.J.P., Postma E.O., and Herik H.J.. Dimensionality reduction: A comparative review [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007.
- [21] Kung S.Y., Diamantaras K.I., and Taur J.S. Adaptive Principal component EXtraction (APEX) and Applications [J]. IEEE Transactions on Signal Processing, 42(5):1202~1217, 1994.
- [22] Duda R. O., Hart P. E., and Stork D. G. Pattern Classification [M]. Wiley Interscience Inc., 2001.
- [23] R.A. Fisher. The use of multiple measurements in taxonomic problems [J]. Annals of Eugenics, 7:179~188, 1936.
- [24] H. Hotelling. Analysis of a complex of statistical variables into principal components [J]. Journal of Educational Psychology, 24:417~441, 1933.
- [25] Tenenbaum J.B. Mapping a manifold of perceptual observations [J]. In Advances in Neural

- Information Processing Systems, MA, USA,. The MIT Press. 1998: (10) 682–688.
- [26] Roweis S.T. and Saul L.K. Nonlinear dimensionality reduction by Locally Linear Embedding [J]. Science, 290(5500):2323–2326, 2000.
- [27] Jardosh A., Papagiannaki K., Belding E. Green WLANs: On-Demand WLAN Infrastructures [J]. Mobile Networks and Applications, 2009, 14(6): 798-814.
- [28] Lin Ma, Yubin Xu, Di Wu. A Novel Two-Step WLAN Indoor Positioning Method [J]. Journal of Computational Information Systems. 2010: 6(14) 4627-4636.
- [29] 万鸣华. 基于图嵌入的特征提取与人脸识别研究 [D]. 南京理工大学, 2011.06: 27~29.
- [30] 李波. 基于流形学习的特征提取方法及其应用研究 [D]. 中国科学技术大学, 2008.08: 43~62.
- [31] I. Hattie. Principal Curves and Surfaces [R]. 1984. 055-056.
- [32] S. K. Nayar, S. A. Nene, H. Murase. Subspace Methods for Robot Vision[R]. 1995. 06-95.
- [33] C. Bregler, S. Omohundro. Nonlinear Manifold Learning for Visual Speech Recognition [C]. In Proceedings of 5th International Conference on Computer Vision. 1995: 494–499.
- [34] C. Bregler, S. M. Omohundro. Nonlinear Image Interpolation Using Manifold Learning [C]. Advances in Neural Information Processing Systems. Cambridge, MA, 1995.
- [35] H. S. Seung, D. D. Lee. The Manifold Ways of Perception [J]. Science.2000, 290(5500): 2268–2269.
- [36] M. Belkin, P. Niyogi. Laplacian Eigen-maps for Dimensionality Reduction and Data Representation [J]. Neural Computation. 2003: 15(6): 1373–1396.
- [37] K. Q. Weinberger, F. Sha, L. K. Saul. Learning A Kernel Matrix for Nonlinear Dimensionality Reduction [C]. Proceedings of the 21st International Conference on Machine Learning. 2004: 106–113.
- [38] Z. Zhang, H. Zha. Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment [J]. SIAM Journal of Scientific Computing.2004, 26(1): 313–338.
- [39] L. Yang. Alignment of Overlapping Locally Scaled Patches for Multidimensional Scaling and Dimensionality Reduction [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2008: 30(3) 438–450.
- [40] T. Lin, H. Zha. Riemannian Manifold Learning [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2008, 30(5): 796–809.
- [41] 詹德川, 周志华. 基于集成的流形学习可视化[J]. 计算机研究与发展. 2005, 42(9): 1535–1537
- [42] 续爽, 贾云得. 基于表情相似性的人脸表情流形[J]. 软件学报. 2009: 20(8) 2191–2198.
- [43] 刘法旺, 贾云得. 基于流形学习与隐条件随机场的人体动作识别[J]. 软件学报. 2008: 19(12) 69–77.
- [44] 詹德川, 周志华. 基于流形学习的多示例回归算法[J]. 计算机学报. 2006:29(11) 1948–1955.
- [45] 杨剑, 王珏, 钟宁. 流形上的Laplacian 半监督回归[J]. 计算机研究与发展. 2007: 44(7) 1121–1127.
- [46] Qiong Yang and Xiao qing Ding. “Symmetrical PCA and its application to face recognition”, Chinese journal of computers, Vol.26, No. 9, pp.1146-1151,2003(In Chinese).
- [47] T. Cox and M. Cox. Multidimensional scaling [M]. Chapman & Hall, London, UK, 1994.
- [48] J.B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a non-metric hypothesis [J]. Psychometrika,29:1–27, 1964.
- [49] 李养成, 郭瑞之, 崔登兰. 微分流形基础[M]. 北京: 科学出版社, 2011.
- [50] G.E. Hinton and S.T. Roweis. Stochastic Neighbor Embedding [J]. In Advances in Neural Information Processing Systems, volume 15, pages 833–840, Cambridge, MA, USA, 2002. The MIT Press.
- [51] 吴勃英, 王佳明, 丁效华, 李道华. 数值分析原理[M]. 北京: 科学出版社, 2003: 39~45.