

学校代号 10532

学 号 G12266110

分 类 号

密 级 普通



湖南大学
HUNAN UNIVERSITY

工程硕士学位论文

基于 Android 的语音智能收银 系统设计与实现

学位申请人姓名 周 昀

培 养 单 位 软件学院

导师姓名及职称 向志强 教授 张华立 高级编辑

学 科 专 业 软件工程

研 究 方 向 传媒管理

论 文 提 交 日 期 2015 年 4 月

学校代号：10532

学 号：G12266110

密 级：普通

湖南大学工程硕士学位论文

基于 Android 的语音智能收银系统 设计与实现

学位申请人姓名：周 昀

导师姓名及职称：向志强 教授 张华立 高级编辑

培 养 单 位：软件学院

专 业 名 称：软件工程

论文提交日期：2015 年 4 月

论文答辩日期：2015 年 6 月

答辩委员会主席：李正良教授

Research and Implementation system of voice intelligent
cash register based on Android

by
Zhou Yun

B.E. (XiangTan University) 1999

A thesis submitted in partial satisfaction of the

Requirements for the degree of

Master of Engineering

in

Software Engineering

in the

Graduate School

of

Hunan University

Supervisor

Professor Xiang Zhiqiang

Researcher Zhang Huali

June, 2015

湖南大学

学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律后果由本人承担。

作者签名：

日期： 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权湖南大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于

1. 保密□，在____年解密后适用本授权书。
2. 不保密□。

(请在以上相应方框内打“√”)

作者签名：

日期： 年 月 日

导师签名：

日期： 年 月 日

摘 要

随着科学技术的进一步发展，人们的生活日益趋向智能化。银行自助存取款机、手机网上购物等自助生活消费方式进入人们的生活。目前，手机成为大多数人的日常生活的必需品之一。它不仅仅有联系的功能，而且可以帮助人们购物付款，在人们的日常生活中有很大的帮助。人们的生活节奏越来越快，对于超市购物的便捷性要求越来越高。结合当下的信息化、智能化方面的科学技术和人们对超市购物的便捷性需求，本文提出了基于 Android 的语音智能收银系统。在系统中采用自助购物机，使用手机支付，不需要收银员收银找零，且可以 24 小时运行，极大的提高了购物的便捷性和效率，减少了超市的运营成本。同时，系统在自助购物机中加入语音引导购物功能，进一步方便广大消费人群。自助购物机与手机支付配合形成一个智能化的收银系统。

本文的主要工作如下：

（1）深入阐述了该系统的研究背景和在现实生活中的意义。详细说明了国内外语音智能收银系统的状况和发展方向，及急需解决的问题。

（2）详细介绍了系统涉及的相关技术和语音识别关键技术研究情况。对系统自助购物端使用的 QT 技术平台的各方面进行了介绍，以及系统手机客户端的 android 技术平台进行了介绍，对语音识别的关键技术进行了分析研究。

（3）对语音智能收银系统进行了系统的可行性分析和需求分析，在系统的分析基础上，对系统进行了详细的设计。明确了系统的可行性和各项实际的需求，得到了系统的架构设计和功能模块设计结果，为随后的实现及测试打下了坚实的基础。

（4）基于设计结果对系统进行了编码实现以及测试。包括环境的搭建、自助购物服务端、手机客户端，然后对各个模块进行了全面的用例测试，保证了系统的实现质量。

（5）对本文所做的工作以及所研究的课题进行了总结，并对下一步所需进行研究工作进行了说明。

本论文很好的设计并实现了超市的智能收银过程。在该系统中，充分利用了当前的信息技术，对智能化生活起到了积极作用。

关键词：自助购物；手机支付；收银系统；语音操作；Android

Abstract

With the further development of science and technology, the life of people tend to be more intelligent. Bank Teller machines, mobile online shopping and other self-help spending way into people's lives. Currently, the phone has become one of the necessities of daily life for most people. It is not only a tool of communication, but also help people in payment when shopping, and it is very helpful in people's daily life. People's life pace faster and faster, so the demand of the supermarket shopping convenience is higher and higher. In combination with informationized, intellectualized aspect of the present science and technology and people's demand of the supermarket shopping convenience, this paper puts forward the voice intelligent cash register system based on Android. The system uses a self-service shopping machine, using mobile payment, not requiring the cashier, and the 24-hour operation, which greatly improves the shopping convenience and efficiency, reduces operating costs of the supermarket. At the same time, the system adds voice functions in the self- service shopping machine, to further facilitate the majority of consumer groups. Self-service shopping machine and mobile payment cooperate to form an intelligent cash register system. The system consists of two parts, the self-service shopping server and mobile client.

In this paper, the main work is as follows:

(1)Further expounds the research background of this system and its significance in real life. Details the phonetic system of smart home and abroad and the development direction, and the problems needed to resolve.

(2)Detailed introduces related technologies in the system and key techniques of speech recognition. QT technology platform for system self-help shopping end use of various aspects are introduced, as well as the system introduces the mobile client android technology platform, analyses the key technology of speech recognition research.

(3)The feasibility and requirement analysis is implemented in voice intelligent cash register system. Based on the analysis of the system, we have designed the system in detail. Clearly the feasibility of the system and the demand, and get the architecture of the system design and function module design, which laid a solid foundation for the implementation and testing.

(4)Implement and test system based on design results, including environmental

construction, self-help shopping service side, mobile phone client, then the comprehensive use of various modules test, ensure the quality of the implementation of the system.

(5)The paper work and research topics are summarized and the needed for the next step research work are illustrated.

This paper designed and implemented the intelligence of supermarket cashier process very well. In the system, we make full use of the current information technology, which has played a positive role in the intelligent life.

Keyword: Self-help shopping; Mobile payment; cash register system; Voice operation; Android

目 录

学位论文原创性声明和学位论文版权使用授权书.....	I
摘 要.....	II
Abstract.....	III
插图索引.....	VII
第 1 章 绪论.....	5
1.1 课题背景及意义.....	1
1.2 国内外研究状况.....	2
1.3 本文主要研究内容.....	3
1.4 本文组织结构.....	3
第 2 章 相关技术介绍和关键技术.....	5
2.1 Android 介绍.....	5
2.1.1 Android 系统架构.....	5
2.1.2 平台优势.....	6
2.1.3 平台的安装.....	7
2.2 QT 介绍.....	10
2.2.1 Qt 平台特点.....	11
2.2.2 Qt 组成模块.....	12
2.2.3 Qt 的安装.....	13
2.3 客户/服务器架构.....	13
2.4 语音识别关键技术研究.....	15
2.4.1 语音识别技术.....	15
2.4.2 动态时间规整.....	17
2.4.3 隐马尔可夫模型.....	18
2.5 本章小结.....	20
第 3 章 系统分析.....	21
3.1 可行性分析.....	21
3.1.1 技术可行性.....	21
3.1.2 经济可行性.....	21
3.1.3 环境可行性.....	22
3.2 系统需求分析.....	22
3.2.1 角色分析.....	22

3.2.2 功能需求.....	24
3.2.3 性能需求.....	25
3.3 本章小结.....	25
第 4 章 系统设计.....	26
4.1 系统架构设计.....	26
4.2 系统功能模块设计.....	27
4.2.1 功能模块划分与组织结构.....	27
4.2.2 功能模块详细设计.....	28
4.3 本章小结.....	30
第 5 章 系统实现与测试.....	31
5.1 开发环境与运行环境.....	31
5.1.1 开发环境.....	31
5.1.2 运行环境.....	31
5.2 自助购物服务端实现.....	32
5.2.1 条形码扫描实现.....	32
5.2.2 购物车实现.....	35
5.2.2 账单生成发送实现.....	38
5.2.3 语音操作实现.....	40
5.3 手机客户端实现.....	42
5.3.1 XML 数据说明.....	43
5.3.2 WiFi 连接实现.....	44
5.3.3 账单获取实现.....	45
5.3.3 支付实现.....	46
5.4 系统测试.....	47
5.4.1 测试环境.....	48
5.4.2 测试过程.....	48
5.4.3 测试结果.....	52
5.5 本章小结.....	52
结论.....	53
参考文献.....	55
致谢.....	58

插图索引

图 2.1 JAVA_HOME 系统变量设置.....	7
图 2.2 CLASSPATH 系统变量设置.....	8
图 2.3 JAVA JDK 版本信息查询.....	8
图 2.4 Android SDK Manager.....	9
图 2.5 Android 开发环境.....	9
图 2.6 创建虚拟机.....	10
图 2.7 Android 虚拟机.....	10
图 2.8 Qt 的跨平台和 UI 框架.....	11
图 2.9 QT 主界面.....	13
图 2.10 客户/服务器架构.....	14
图 2.11 语音识别技术模型.....	15
图 2.12 二维坐标表示.....	18
图 3.1 系统角色及功能的用例图.....	23
图 3.2 系统整体图.....	24
图 4.1 系统网络拓扑图.....	26
图 4.2 系统软件架构图.....	27
图 4.3 系统功能模块图.....	28
图 4.4 自助购物车工作流程图.....	29
图 4.5 手机客户端工作流程图.....	30
图 5.1 新建 Qt Quick UI 工程.....	33
图 5.2 新商品存档交互界面.....	34
图 5.3 顾客商品扫描界面.....	34
图 5.4 购物车界面.....	35
图 5.5 main.qml.....	36
图 5.6 bill_layout.xml.....	44
图 5.7 head.xml.....	44
图 5.8 WIFI 网络没开提示.....	45
图 5.9 WIFI 网络没开提示.....	47
图 5.10 测试平台网络拓扑图.....	48
图 5.11 QT 自助购物端初始化界面.....	49
图 5.12 QT 端为侦听状态.....	49

图 5.13 新商品添加.....	50
图 5.14 购物车信息.....	50
图 5.15 应用程序账单接受.....	51
图 5.16 账单信息.....	51

第 1 章 绪论

1.1 课题背景及意义

随着现代通信与信息、计算机网络等技术的发展，智能化已成为不可阻挡的趋势。智能化正渐渐的渗入人们的生活，像：建筑智能化、智能家居化等。超市是人们日常生活用品的主要来源场所，是让智能化进一步走入人们生活的重要发展点。

在人们购物时，经常遇到这样一个问题：挑选好了一大堆自己需要的商品后，却发现不得不加入长长的结账队伍中去，花费十几分钟甚至是几十分钟来等待结账。如何解决这个问题呢？我们先来看看银行这一例子，在过去，人们常需要花费很长的时间在排队上，为了解决这一问题，在文献^[1]中提到，大力推广自主服务这一策略。自助服务技术^[2]在生活中的很多方面得到了推广，像自助餐厅，自助营业厅等。自助服务可以体现一个国家的素质的高低，激发国人的自觉性。从现实观察中可以看出，自助服务技术在一定程度上缓解了排队压力。是否可以在购物中引进自助服务技术呢？采用自助购物的模式？根据自助服务在生活其它方面的应用，可以看出，自助服务可以缓解一定程度的排队压力。

采取自助购物模式，不仅仅只是缓解了排队压力，还具有许多其它的优势。对卖家来说，可以减少人工成本，提高结算效率以及增加营业时长，可 24 小时营业。对买家来说，可快速结算，节约时间。在传统的收银员结算模式下，效率往往偏低。人的精力是有限的，长时间的工作往往会降低工作效率，而且长时间的重复相同的操作，扫描、收银、找零等会造成收银员的情绪低落，进而影响服务态度和工作效率。而计算机是可以 24 小时工作的，它们是没有情绪的，只需要定时维护，保证其正常工作即可。再加上其 24 小时工作，为急需在半夜购买商品的顾客提供了许多便利。顾客全程都是自己服务自己，给他们更多的自主权，给人以更亲切的感觉。

超市是面向广大客户的，而客户的教育水平高低不一。为了使得更多的客户能够更好的使用该系统，在本次设计中，加入了语音部分，使用语音来引导客户购物。语言是人们日常生活的最简单的交流模式，在许多电影中，我们常常可以看到人机对话^[3-5]的炫酷场景，这是对未来生活的计划，许多科学研究人员正投身这一领域的研究。将语音应用到本系统中，让客户亲身体验人机交互所带来的便利。

在大多数超市中，结算都需要在结算台结账，获取账单。双方交易时，会花

费许多时间在找零，打印账单上面，另外纸质账单携带不便，还不利于环保。为解决这个问题，本系统采用手机支付，在扫描需要购买的商品后，将账单信息发送到手机上，利用手机支付，完成交易，生成最后的电子账单，保存在手机中，便于客户日后查看。目前，手机支付^[6-7]获得了许多企业的支持，例如：腾讯和阿里巴巴目前在打车方面的投入。手机支付将成为每人必会的支付方式。

1.2 国内外研究状况

随着时代的发展，信息技术的发展，购物模式也在呈现多样化、高效化自动化和信息化。在国外，许多国家都已在超市中开启了自助购物模式，像英国超市巨头乐购（Tesco），美国和新西兰等，自助购物在许多国家已是一种普遍的购物形式。在国内，自助购物还是“新鲜事”，近年来发达地区也在积极的尝试，如深圳市在 2014 年 1 月份在某些超市引进自助购物模式：没有收银员，顾客在挑选好上自己想要的商品后，自行使用条形码扫描器扫描商品，并刷卡完成支付交易，全程都需自助完成。

根据目前国外已有的自助购物系统来看，系统从功能上分为如下两部分：自助商品识别和自助支付。在国内外的大多数超市里，商品识别采用条形码扫描方式，本系统仍采用条形码扫描。自助支付方式却是多样性的，有现金支付、银行卡支付、会员卡支付以及手机支付等，本系统采用手机支付方式。系统从技术上分为如下两部分：自助购物服务端和手机客户端。自助购物服务端用于自助商品识别及确认商品和生成账单等工作。手机客户端用于账单确认和支付等工作。现有系统都是购物者手工操作，本系统为更加友好和智能化，加入语音^[8]引导购物，使其面向大众化，有更多的人能够理解系统的使用流程。

在自助购物服务端界面方面，有许多用于做界面的语言，像 VB.Net、VC.Net、C#.Net、C++ Builder、C# builder、delphi.net 等等。它们各有自己的特色。QT^[9-10]是基于 C++ 语言的界面开发工具，它曾作为诺基亚旗下的编程工具而风行一时，后被 Digia 收购，于 2014 年，发布了 Qt Creator 3.1.0^[11]，它是一款跨平台的集成开发环境，全面支持 iOS、Android^[12-14]、WindowPhone，QT 在界面开发方面，支持 XML^[15-18]，有大量的支持文档等，具有许多优势。

在手机系统方面，目前的智能手机的主流操作系统有 Android、iOS、Ubuntu、WindowPhone、Linux、Symbian^[19]等。在全球，Android 占领先地位。手机应用^[20]为人们的日常生活提供了巨大的帮助。据最新数据显示，Android 操作系统在德国占比达 75.4%，西班牙达 86.2%，在我国，Android 操作系统占 78.8%，而苹果的 iOS 操作系统在许多国家的市场份额占比有所下降。

再来看语音识别^{[21][22][23]}方面，针对特定应用的少量词汇的语音识别技术发展较为成熟，孤立词的特定人语音识别也较为成熟。大词汇量、非特定人、连续语

音识别技术进展缓慢，未来将是研究的难点和重点。语音识别技术有线性预测分析技术、动态时间规整、矢量量化、隐马尔科夫模型和人工神经网络等。在我国，有许多公司投入语音识别的研究领域之中，比如：捷通华声、尚科语音和科大讯飞^{[24][25]}等，该领域做得最好的要属科大讯飞了。在国外也有许多公司的大量科研投入，像微软、Intel 等。

在自助支付方面，本系统采用手机支付的方式。手机支付工作方式是将用户的银行卡与 SIM 卡建立联系，并通过短信接受与发送的形式完成交易。为完成支付，商家应提前签约相应的银行和信用中心。

1.3 主要研究内容

本文的主要研究内容包括自助购物服务端设计和实现、手机客户端设计和实现、语音操作设计和实现三大块，详细内容如下：

（1）自助购物服务端设计和实现

首先扫描条形码，转换成计算机可读的二进制，匹配数据库，得出商品相应的信息。说明采用的软件系统的架构，对自助购物服务端模块设计和编码实现以及测试。

（2）手机客户端设计和实现

客户端利用 Android 实现，并连接到服务端发射的 WLAN 网络中，实现两者的数据互传。客户端接受到账单后，确认付款。付款成功后，返回给服务端付款成功的消息，服务端在发送交易结束的信息给客户，客户可以带走购买的商品。

（3）语音操作设计和实现

基于 QT 实现人机交互界面。加上语音识别，引导客户进行操作。使用 QT 的 QAudioInput 输入语音，生成.pcm 格式编码的音频文件，再将其转换为.wav 格式编码的音频文件；登录科大讯飞语音识别服务端（首次使用时，上传 abnf 格式的语音语法），上传.wav 音频文件，由科大讯飞服务端识别并返回识别结果，接受返回结果；处理相应的操作。以上过程是在系统服务端上实现，系统基于 window7 平台实现，在服务端设置 WIFI 热点。

1.4 本文组织结构

论文分为 5 个章节对基于 Android 的语音智能收银系统进行分析、研究，以下是各章节的主要研究内容：

第 1 章绪论，阐述了本系统的课题背景和现实意义，综合概述了国内外语音智能收银系统的发展情况，研究现状及存在的问题，及本文的组织结构。

第 2 章相关技术介绍和关键技术研究，对 Android 平台、Qt 平台以及基于 C/S

模式的语音智能收银系统和语音识别所用到的相关技术和理论进行了介绍性的说明。展现了 Qt 平台开发的实用性及灵活性。

第 3 章系统分析，对基于 Android 的语音智能收银系统从经济、技术和环境等方面进行了可行性分析，然后从系统的整体需求出发，对系统进行了角色的分析、具体功能需求的分析和性能需求的分析。

第 4 章语音智能收银系统设计，根据前一章所进行的系统需求的分析，对于系统的架构进行了详尽的阐述。具体的分析说明了系统中具体模块功能，并对各模块作用进行了说明。然后分别设计了客户端系统和服务器系统的实现流程。

第 5 章系统的实现与测试。对于系统运行和测试的环境进行了描述，同时介绍了系统搭建的硬件和软件的配置，以及各模块功能的具体实现步骤。对系统各个模块进行了全面的用例分析测试。最后根据测试过程得到了测试结果。

最后对于本文所做的工作以及所研究的课题进行了总结，并对下一步所需进行研究工作提出了展望。

第 2 章 相关技术介绍和关键技术研究

本章对智能化收银系统相关技术进行了详细的介绍，相关技术主要包括 Android 系统架构，QT 等。深入探讨和研究了智能化收银系统中的语音识别关键技术。

2.1 Android 介绍

Android 一词的本义指“机器人”，Android 一词最早出现于法国作家利尔亚当在 1886 年发表的科幻小说《未来夏娃》中。他将外表像人的机器起名为 Android。2012 年 7 月美国科技博客网站 Business Insider 评选出二十一世纪十款最重要电子产品，Android 操作系统和 iPhone 等榜上有名。

Android 是一个基于 Linux 内核的自由和开放源代码的操作系统，主要使用目标为移动设备，比如：平板电脑、智能手机等。Android 操作系统最开始由 Andy Rubin 开发，2005 年 8 月由 Google 收购并投资开发。2007 年 11 月，Google 与 84 家硬件制造商、软件开发商及电信营运商组建开放手持设备联盟共同设计和研发以及改良该系统。随后 Google 以 Apache 开源许可证的授权方式，发布了 Android 的源代码。第一部 Android 智能手机于 2008 年 10 月发布。Android 系统的应用已经逐渐扩展到其他领域如电视、游戏机等。从 2011 年第一季度到 2013 年第四季度，Android 系统手机的市场份额一直在增长，以至于现在占据着绝对的市场优势^[26]。

2.1.1 Android 系统架构

Android 系统架构采用了分层的架构。Android 主要分为四个层，分别是 Linux 内核层、系统运行库层、应用程序框架层和应用程序层，前一层作为后一层的基础。

1. 应用程序

在发布时，Android 包含一系列重要的应用程序一起发布，比如浏览器、短消息程序、联系人管理程序等等。由于 Java 语言适合应用高层应用开发，所以都使用 JAVA 语言开发而成。

2. 应用程序框架

应用程序框架是一系列的 API，提供给开发人员开发应用程序使用。通过一个开放的平台，Android 提供给开发者创建丰富和创新应用的能力。开发者可以

自由的访问本地信息，向状态栏添加通知，运行后台服务等。应用程序架构简化了组件的重用，该重用机制使用户可以方便的加入、更换、删除程序组件。在每个应用后面的是一系列的服务，其中包括：丰富且可扩展的可视控件，可以用来构建应用程序的 UI 和用户进行交互，其中包括列表，网格，文本框，按钮等。内容提供者使得一个应用可以获取另一个应用的数据，或者将它们自己的数据共享给其他的应用。资源管理器完成字符串、图形和布局文件等资源的管理。通知管理器完成在状态栏中显示自定义的通知信息。活动管理器对应用程序的生命周期进行管理和提供常用的导航功能。

3. 系统运行库

Android 系统运行库主要是一些 C/C++ 库，这些库被 Android 系统中不同的应用程序框架组件使用，通过 Android 应用程序框架为开发者提供服务。主要有如下核心库。系统 C 库，标准 C 系统函数库 Libc；媒体库，支持音频、图像、视频录制和回放。Surface Manager，对外观子系统进行管理；LibWebCore，web 浏览器引擎用。

4. Linux 内核层

Android 的核心系统服务来源于 Linux 内核，所以 Android 很多的 core 服务都依赖于 Linux 内核。同时，Linux 内核主要是作为软件栈和硬件之间的抽象层，用于消除具体硬件细节为软件提供一致的接口^[27]。

2.1.1 平台优势

1. 开放性利于创新

由于 Android 平台是开放源码的，并且手机厂商可以自由的使用。因此，吸引了众多的开发者和投资者参与，使得 Android 的发展速度极其的迅猛，显著的提高了创新率。

2. 丰富的选择

由于开放性，Android 一直以来得到充足的发展，参与的硬件厂商很多，开发人员也多，也因此，Android 平台上各种应用都有。所以站在用户的角度上看，对硬件和软件都有着很丰富的选择。

3. 开发资源丰富

Android 平台主要使用 Java 语言开发应用程序，Java 语言是一种易用且功能强大的语言，而 iOS 的平台是封闭的，且开发语言使用的是 objective-C，相对来说 Java 的使用者更广泛。因此，开发资源丰富，不易受到各种开发问题所阻挠，所以，Android 平台上的新软件产生速度非常的惊人。

4. Google 各种服务支持

Google 是当前最大的互联网公司，拥有强大的技术团队，研发并提供大量基

于互联网的服务，如搜索引擎、Google 地图、Google 翻译等等优秀的网络服务。而 Android 本身出自 Google 公司，那么和自己公司其他产品服务的融合，将会更加的完整和流畅，大大的增强了 Android 平台的易用性，服务的多样性，极大的增强了平台的竞争力。


2.1.3 平台的安装

Android 开发环境搭建需要：

下载 JAVA^[28-29] JDK: jdk-8-windows-i586.exe。

下载 Eclipse+ADT 的集成: adt-bundle-windows-x86-20131030.zip。

首先，安装 JAVA: jdk-8-windows-i586.exe:

第一步 双击  图标，选择安装路径（本人安装在 E:\Program Files\Java 中）进行安装，过程简单。

第二步 设置环境变量：

1. 打开计算机属性->高级系统设置->环境变量。
2. 新建系统变量：变量名为 JAVA_HOME，变量的值为 Java 的安装的 jdk 路径，例如，我的安装路径是：E:\Program Files\Java\jdk1.8.0，如图 2.1 所示。
3. 编辑系统变量 CLASSPATH（如果没有，则新建该变量），在变量值后追加路径为：.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar；如图 2.2 所示。
4. 在系统变量 Path 后面追加路径：%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin；
5. 打开 cmd 窗口，输入：java -version，查看 Java 的信息如下图 2.3 所示。



图 2.1 JAVA_HOME 系统变量设置



图 2.2 CLASSPATH 系统变量设置

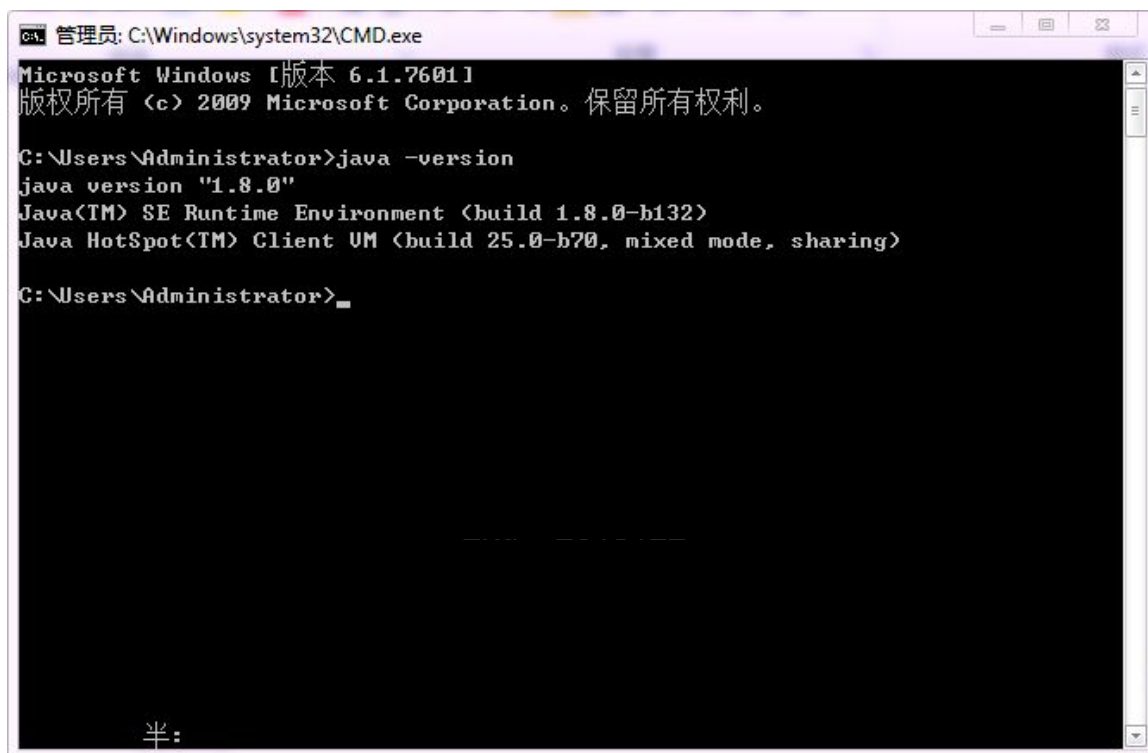



图 2.3 JAVA JDK 版本信息查询

再次，解压 adt-bundle-windows-x86-20131030.zip，双击  SDK Manager.exe，下载 SDK，如下图 2.4 所示：

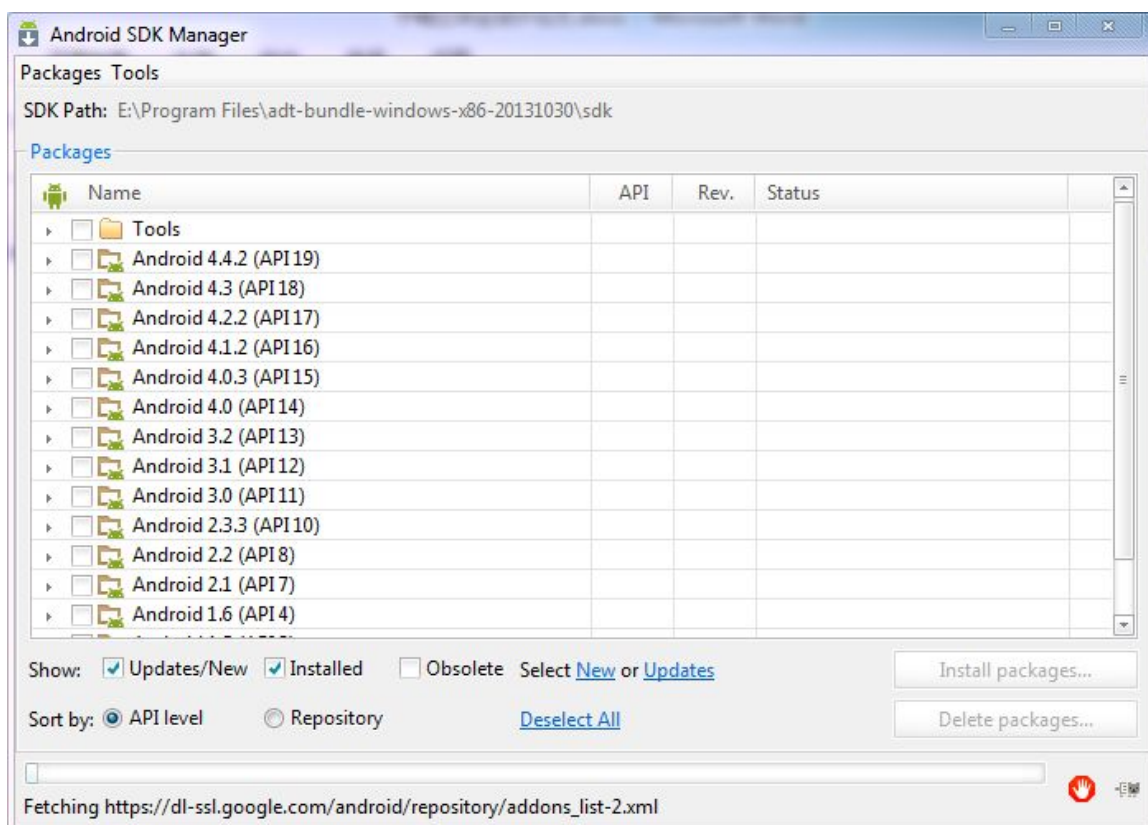



图 2.4 Android SDK Manager

Android 开发环境安装完成，打开：adt-bundle-windows-x86-20131030\eclipse 文件夹下的  eclipse.exe 便可以开发 Android 应用程序，如图 2.5 所示。

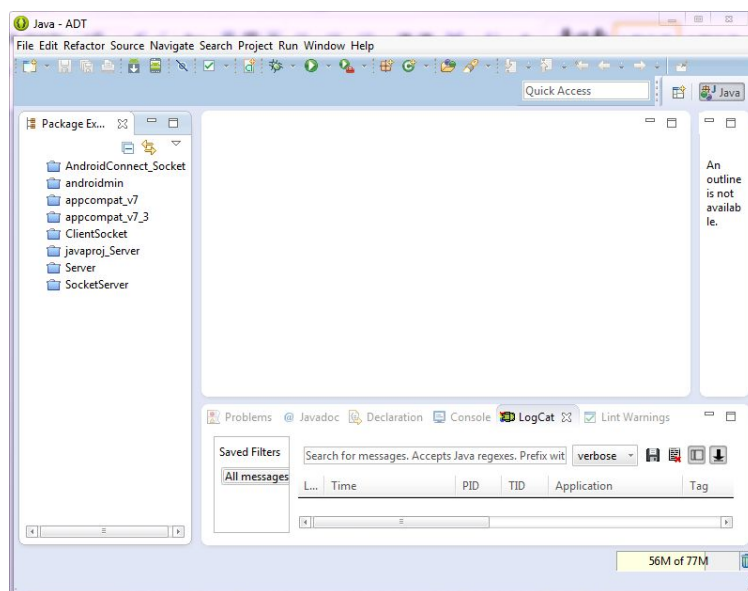


图 2.5 Android 开发环境

为 Android 创建虚拟机，打开 Window->Android Virtual Device Manager，新建一个模拟器，输入相应的信息，点击确定，创建模拟器，如图 2.6 所示。

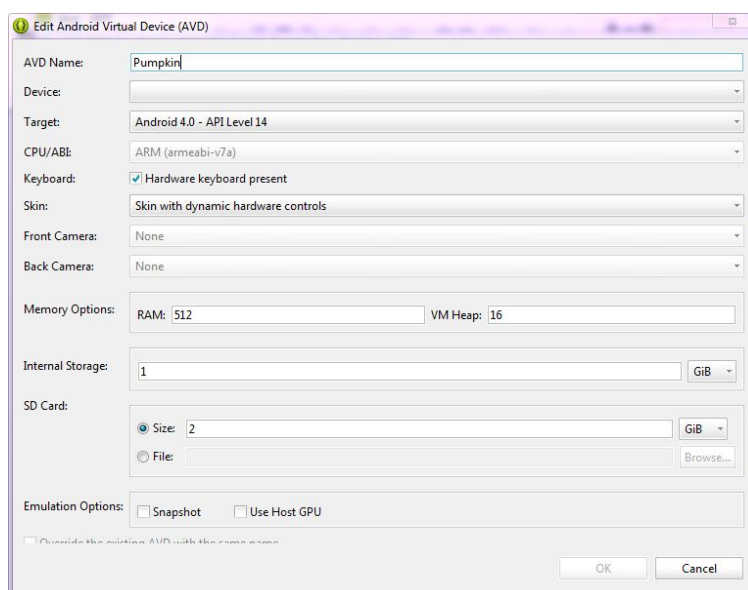


图 2.6 创建虚拟机

打开虚拟机，如图 2.7 所示。

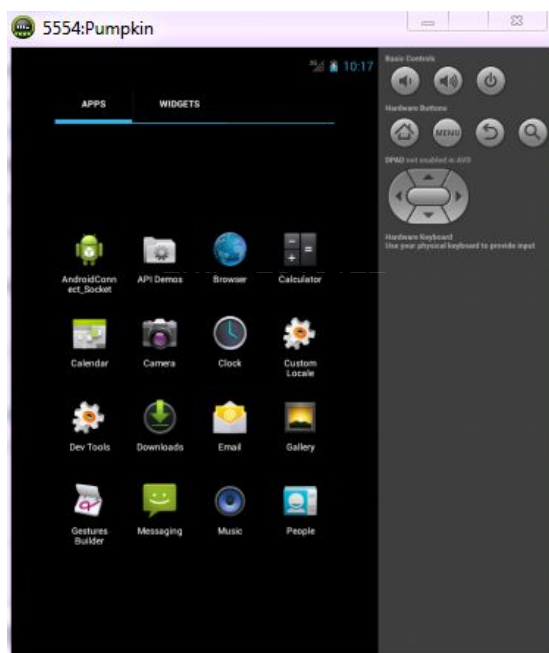


图 2.7 Android 虚拟机

2.2 QT 介绍

Qt 是一个著名的高性能、跨平台的 C++ 应用程序框架。不仅是 GUI 组件，使用 Qt 在一定程度上不再需要研究 STL，不再需要到处去找解析 XML、连接数据库、访问网络等各种第三方库，因为 Qt 自己内置了这些技术。它给图形用户界面开发者提供了所需的所用功能^[30]。

Qt 和 wxWidgets 一样，也是一个标准的 C++ 库。它的语法十分清晰，使用信号槽（signal/slot）机制，让程序看起来很明白，这也是很多人优先选择 Qt 的一

个很重要的原因。这种机制虽然很清楚，但是它所带来的后果是需要进行预处理，才能够再使用标准的 make 或者 nmake 进行正常编译，并且信号槽的调用要比普通的函数调用慢很多。

Qt 由 Trolltech（奇趣科技）发布，之后被 Nokia 收购，将 Qt 应用于 Symbian 程序开发，后来，Nokia 将 Qt 出售给了 Digia。伴随 Qt 一直有两种授权协议：商业授权和开源授权。在 Qt 的早期版本，商业授权包含一些开源授权不提供的组件，但是在近期版本则不存在这个问题。以往人们对 Qt 的开源授权多有不满。不过，现在 Qt 的开源版本使用的是 GPLv3 以及 LGPL 协议。这意味着，你可以将 Qt 作为一个库链接到一个非开源软件里面。所以，Qt 协议的争议已不存在。

Qt 应用程序框架是面向对象且容易扩展，且允许组件编程。Qt 包含丰富的 GUI 元素，提供了易用的集成开发环境工具，能极大的加速图形用户界面的开发。Qt 的跨平台和 UI 框架如图 2.8 所示。

Qt SDK

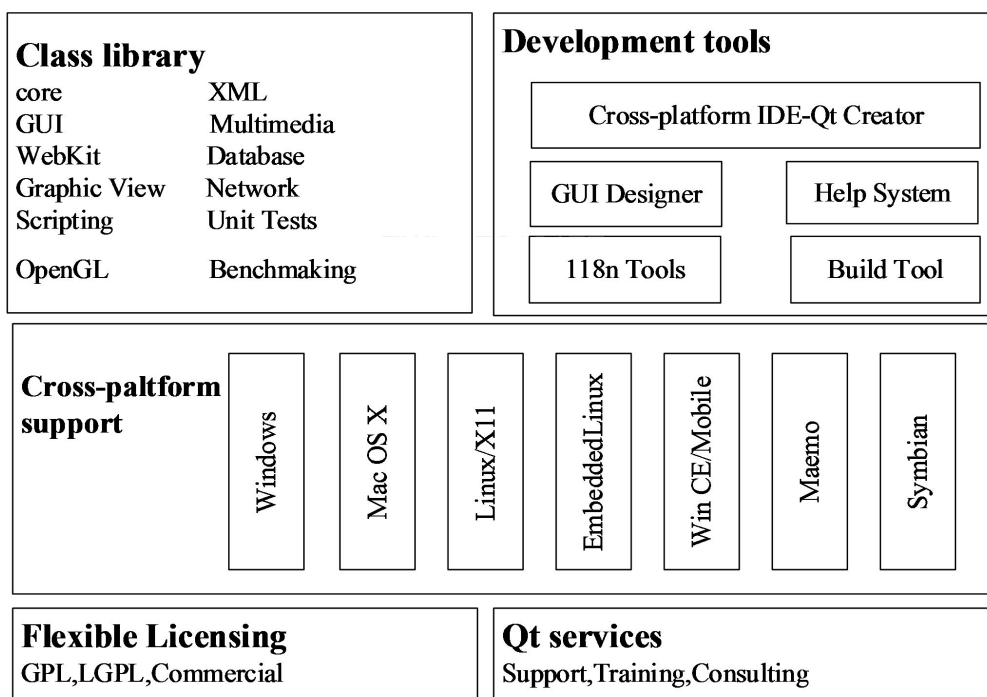


图 2.8 Qt 的跨平台和 UI 框架

1.2.1 Qt 平台特点

在 X-Window 上有 Open win、Motif、GTK 等与 Qt 相似的图形界面库；Windows 平台上有 ATL、MFC 和 VCL 等与 Qt 相似的图形界面库，但是和 Qt 相比，都具有很大的差距，相对来说，Qt 具有更多的优点：

(1) Qt 具有很好的跨平台性，而其他的图形界面库则多数没有跨平台性，或者跨平台性不是很好。Qt 支持各种操作系统，比如：windows、linux 等 pc 操作系

统、手机操作系统等等。支持的详细操作系统如表 2.1 所示。

表 2.1 Qt 支持的平台

操作系统类型	具体操作系统
pc 操作系统	windows、Solaris、AIX、HP-UX、Linux、BSD、Mac
嵌入式操作系统	Embedded Linux
手机操作系统	Windows CE / Mobile、Symbian、Maemo/MeeGo

(2) Qt 完全的面向对象，具有良好的封装机制。因此，使用 Qt 进行开发的应用具有高度模块化的结构，并且应用易于修改，可重用性显著提高。另外，Qt 采用了信号槽(Signals/Slots)机制，抛弃了传统的回调(Callback)函数机制^[31]，相对于回调函数机制，它更加的安全，Qt 类库均是基于信号槽机制实现。

(3)支持 2D/3D 图形渲染，支持 OpenGL。

(4) Qt 拥有几百个 C++类库，具有丰富的 API 函数，同时，伴随有大量的开发文档，支持 XML 数据解析，支持正则表达式等功能。

1.2.2 Qt 组成模块

经过许多年的发展，Qt 不但拥有了完善的 C++图形库，而且最新的版本整合了数据库、网络、OpenGL 库、多媒体库、XML 库、脚本库等等，其核心库也加入了多线程模块、进程间通信模块等，最大程度的赋予了 Qt 开发复杂的跨平台应用的能力。

由于各家编译程序差别很大。Qt 为了跨平台的兼容性，只能以“最低兼容规格”来设计。因此 Qt 必须建构出自己的容器组件。下列是 Qt 提供主要用于软件开发的模块。

QtCore: QtCore 模块是所有基于 Qt 的应用程序的基础，提供信号与槽的对象间通讯机制、IO、事件和对象处理、多线程。

QtGui: 包含了开发图形用户接口应用程序所需的功能。使用其支持的各个平台的原生图形 API。支援反锯齿、向量形变。支持 ARGB 顶层 widget。

QtMultimedia: 提供了用于多媒体内容处理的 QML 类型集和 C++类集。同时提供访问摄像头及音频功能的 API。包含的 Qt 音频引擎支持三维音频回放及管理。

QtNetwork: 提供了网络程序设计功能。支持通用协议，如 HTTP、FTP 和 DNS，包括对异步 HTTP 1.1 的支持。与较低层的 TCP/IP 和 UDP 协议，如 QTcpSocket、QTcpServer 和 QudpSocket。

QtOpenGL: 提供在应用程序中使用 OpenGL 和 OpenGL ES 加入 3D 图形。在 Windows 平台上亦支持 Direct3D。

QtOpenVG: 提供 OpenVG 绘图支持的一个插件。

QScript: 包含完全整合的 ECMA 标准脚本引擎。提供信号与槽机制简化对

象间通讯和 QtScript 侦错程序。

QtScriptTools: 额外的 Qt Script 组件。

QtSql: 将数据库整合至应用程序。支持所有主要的数据库驱动包括 ODBC、MySQL、PSQL、SQLite、ibase、Oracle、Sybase、DB2。

QtSvg: 支持 SVG 格式。


QtWebKit: 整合 WebKit, 提供了 HTML 浏览器引擎, 便于在原生应用程序中嵌入网络内容和服务。

QtXml: 提供了 XML 文档的读者和编写器、支持 SAX 和 DOM。

QtXmlPatterns1: 提供了 XQuery 和 XPath 引擎支持。

Phonon: 整合 Phonon, 支持跨平台应用程序播放音频和视讯内容。Qt5 开始不支持 Phonon。

1.2.3 Qt 的安装

首先, 下载  qt-opensource-windows-x86-mingw48_opengl-5.2.1.exe。双击便可进行安装。安装过程十分简单。安装后打开 Qt Creator, 如下图 2.9 所示:

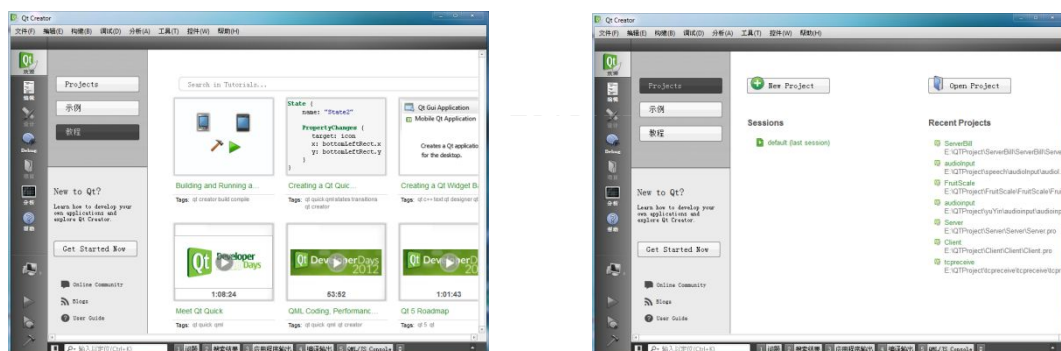


图 2.9 QT 主界面

然后, 设置 QT 的环境变量: 计算机属性->高级系统设置->环境变量, 在 Path 系统环境变量后面追加路径 QT 的安装路径:

E:\Qt\Qt5.2.1\5.2.1\mingw48_32\bin;E:\Qt\Qt5.2.1\Tools\mingw48_32\bin;E:\Qt\Qt5.2.1\Tools\mingw48_32\i686-w64-mingw32\lib;

2.3 客户/服务器架构

客户端-服务器 (Client/Server) 架构简称 C/S 架构, 是一种网络架构, 客户端通常为一个采用图形用户界面的程序; 服务器通常为一个业务逻辑处理的程序。网络上的计算机不是一个客户端, 就是一个服务器。服务器软件一般运行在强大的专用商业计算机上。另一方面, 客户端一般运行在普通个人电脑或者工作站上。服务端的特征: ①被动的角色 (从); ②等待来自客户端的要求; ③处理要求并

传回结果。客户端的特征：①主动的角色（主）；②发送要求；③等待直到收到回应。在实际的部署中，一般只有一个或几个服务器，而可以有很多个客户端，每个客户端实例都可以向服务器发出业务请求。服务器根据不同的客户端的请求进行处理。客户/服务器架构如图 2.10 所示。

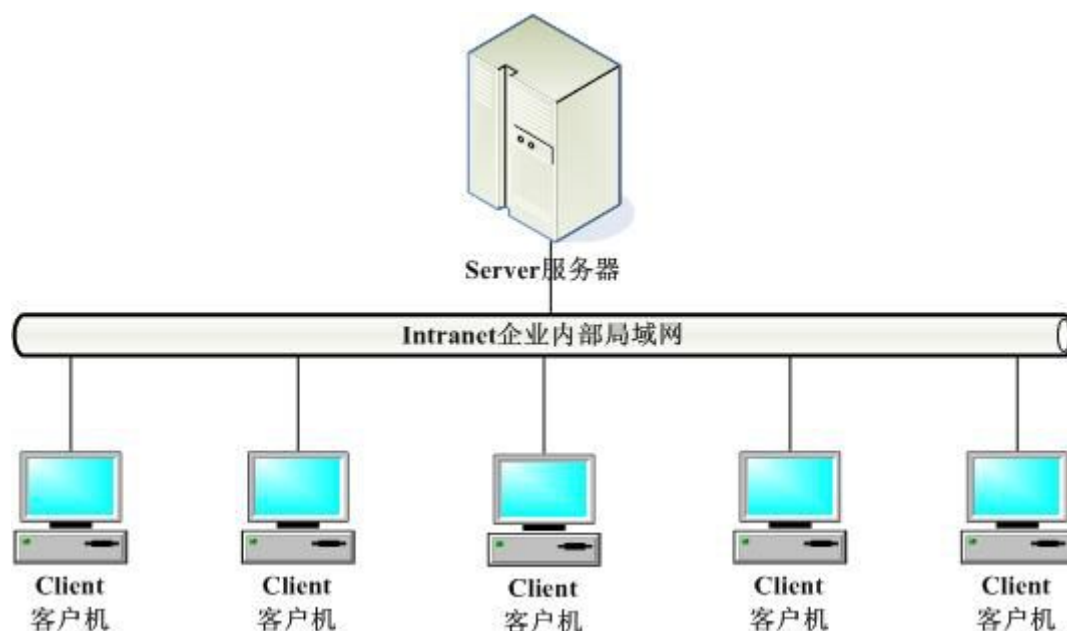


图 2.10 客户/服务器架构

C/S 架构一般分为两层，其客户端作为一层，服务器端作为一层。另外，站在服务器端的角度，有两种不同的形式，一种是直接将数据库作为服务器端，这种形式的服务器端将业务逻辑放在了客户端，服务器端只负责数据的存取；另一种是同客户端一起开发的业务逻辑处理程序，通过 Socket 编程通信，此种形式业务逻辑的处理则不在客户端。

相对于 B/S 架构，C/S 架构的客户端需要处理的事情更多，需要更多的工作量。因为客户端需要实现部分业务逻辑与界面展示。因此 C/S 架构中，客户端需要耗费很大的投入。但是 C/S 架构的优势也是明显的，C/S 架构的优势能很大程度上发挥个人电脑或者工作站的计算资源，在客户端可以把大多数业务处理后再发送给服务器。相应的服务端就能快速响应。总体上主要的优势有以下两点：应用服务器运行数据负荷较轻；数据的储存管理功能较为透明^[32]。

C/S 架构通过不同的途径应用于很多不同类型的应用程序，同时，B/S 架构也只是 C/S 架构的特殊情况，浏览器其实就是一个客户端。例如，当你在维基百科阅读文章时，你的电脑和网页浏览器就被视为一个客户端，同时，组成维基百科的电脑、数据库和应用程序就被视为服务器。当你的网页浏览器向维基百科请求一个指定的文章时，维基百科服务器从维基百科的数据库中找出所有该文章需要的信息，结合成一个网页，再发送回你的浏览器。

2.4 语音识别关键技术研究

本节详细探讨了语音识别技术，详细介绍了语音识别中的两个经典算法，包括动态时间规整和隐马尔可夫模型。

2.4.1 语音识别技术

语音识别是一门交叉学科。语音识别的研究涉及很多领域的内容，比如：信号处理、模式识别、发声机理与听觉机理、概率论与信息论、人工智能等。近二十年来，语音识别技术取得显著进步，开始从实验室走向市场。人们预计，在不久的将来，语音识别技术将应用于工业、通信、医疗、汽车电子、家庭服务、家电、消费电子产品等领域。很多业内人士认为语音识别是近年来信息技术领域重要的科技发展技术之一^[33]。

与机器进行视觉和听觉等高层次的交流，让机器明白看的懂和听的懂是人们长期的愿望和奋斗的目标。语音识别技术就是在机器听觉方面的研究成果。当下的语音识别技术主要注重让机器把语音信号变为对应的命令或文本。语音识别技术的过程主要有特征提取技术、模型训练及模式匹配技术三个方面，总体模型如图 2.11 所示。

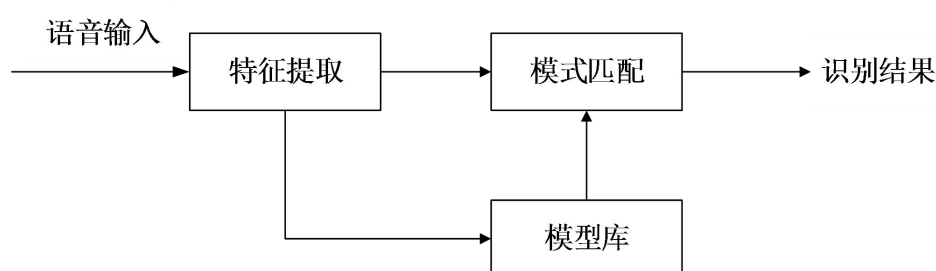


图 2.11 语音识别技术模型

语音识别过程是一个特征提取、模型训练和模式匹配有机结合的过程，语音数据的完整性、有效性、均衡性直接关系到模板训练结果是否有效，从而影响语音系统识别的效果。为了训练出一个好的语音模型，通常需要大量高质量的语音数据，特别是针对非特定人语音识别技术来说，这一点则更为关键。因此，首先要完成一个语音数据库建设，语音识别的研究工作才能有效的进行，数据库必须具有总体性，能够充分的反应总体的特性，能很好地反映实际情况。如果数据库没有如上特点，那么，使用该语音库训练的语音模型的识别效果将很难满足实际的需要。

语音识别过程中最重要的一步就是计算未知语音数据模式与语音模型库中的数据模式进行相似度测量，将相似度最高的作为匹配的结果。语音识别所广泛采用的主要的模式匹配方法有动态时间规整（Dynamic Time Warping, DTW）、隐马尔可夫模型（Hidden Markov Model, HMM）和人工神经网络（Artificial Neural

Networks, ANN) [34]。

(1) 动态时间规整: 把特征分析提取的一组随时间而变的特征矢量序列和事先通过学习后存在机器里的样本序列进行比较。由于人说话的速度有快有慢, 因此对语音序列进行动态时间规整(DTW)是样本匹配法成功的关键。采用动态规划技术(Dynamic Programming, DP)将一个复杂的全局最优化问题变为许多局部最优化问题, 一步一步地进行决策, 很好的解决了语音信号特征参数序列时长不等的难题。该算法小巧, 系统开销小, 识别速度快, 在孤立词语音识别中获得了良好性能。

(2) 隐马尔可夫模型: HMM 是一个概率模型, 它是由马尔可夫链演化而来, 用于描述随机过程统计特性。它由两个相互关联的随机过程共同描述信号的统计特性, 其中一个是隐蔽的(不可观测的)具有有限状态的马尔可夫链, 另一个是与马尔可夫链的每一状态相关联的观察矢量的随机过程(可观测的)。是目前已有的强有力的语音识别算法。对语音识别而言, HMM 的输出值常常是各帧的声学特性。为了减小模型的复杂度, HMM 模型有两个假设, 第一个内部状态转移只与前一个状态有关, 第二个输出值只与当前状态或当前状态转移有关。

(3) 人工神经网络: 人工神经网络是一种信息处理的数学模型, 结构类似于大脑神经突触联接。神经网络是一种运算模型, 由很多的节点相互联接组成的一个网状结构的信息处理系统, 该系统具有非线性和自适应性等特点。试图模拟大脑的神经网络结构和记忆信息、处理信息的方式, 对信息进行处理, 以获得人脑处理信息的优点。人工神经网络系统具有分布式特性且具有并行计算的能力, 在人工智能机理方面有很大的进步性, 传统人工智能是基于逻辑符号的, 对直觉和非结构信息的处理有严重的缺陷, 而人工神经网络系统中的人工智能机理具有自组织、自适应等特点, 所以对直接和非结构信息的处理具有很好的效果, 因此, 是当下研究和应用的热点。

HMM 非特定人员语音识别方法的优点是用户不需要对模型库进行训练, 用户可以直接使用, 有很好的稳定性, 语音识别准确率不会随着使用时间的增加而降低, 但也有其自身难克服的问题。其一, 中国是一个多民族国家, 民族很多, 而且各个地方方言很多, 所以不同方言的存在是语音识别一个难以克服的问题, 因此, 使用区域必然会受到极大的限制; 其二, 非特定人员语音识别需要提前采集大量的语音原始数据, 才能训练出充足完整的识别模型, 因此, HMM 非特定人员语音识别的前期投入将大大的增加; DP 特定人语音识别的优点就是方法不复杂, 对硬件资源需求不高; 另外一个优点是该方法训练也很容易, 并且不需要提前收集大量的语音样本数据, 因此, 能大大节约前期成本的投入, 而且当用户需要更换具体的语音命令时, 更新快速且容易。DP 特定人语音识别的缺点是它的稳定性很差, 由于其实特定人的, 所以对部分人的语音识别效果好, 可能另外部

分人语音识别效果却不好；训练完之后，短时间内语音识别效果好，但随着使用时间的增加而语音识别率可能会降低。本系统语音识别采用基于动态时间规整的匹配方法，并加以于隐马尔可夫统计模型（HMM）框架的非特定人语音识别优点，提出适合超市收银的语音识别算法。下面详细说明动态时间规整和隐马尔可夫统计模型。

2.4.2 动态时间规整

DTW 是把时间规整和距离测度计算结合起来的一种非线性规整技术。它是以动态规划为基础，找出两个向量的最短距离的技术。

假设测试和参考模板特征矢量序列分别表示为 T 和 R 表示，为了评估它们的相似性的大小，可以通过这两个矢量序列之间的距离来评估，并且假设距离越短则相似性越高，所以我们需要计算两个矢量序列的最短距离作为相似度的评估值。为了计算这一距离，应从 T 和 R 中各个对应帧矢量之间的距离算起。设 N 和 M 分别为 T 和 R 的帧个数， n 和 m 分别是 T 和 R 中任意挑选的帧号， $d[T(n), R(m)]$ 表示 $T(n)$ 和 $R(m)$ 这两个挑选的帧特征矢量的距离。具体距离函数取决于实际使用的距离计算方法^[35]。

若 $N=M$ ，则两个特征矢量序列已经对齐，距离可以很容易的计算出来，若 $N \neq M$ ，则两个特征矢量序列没有对齐，需要对特征矢量序列 T 和 R 进行对齐处理。对齐采用的是动态规划（Dynamic Programming）的思想。将测试模板的各个帧以时间先后编号为 $1, 2, \dots, N$ ，在直角坐标系中横轴上标出测试模板帧编号，将参考模板的各个帧以时间先后编号为 $1, 2, \dots, M$ ，在直角坐标系中纵轴上标出参考模板帧编号，通过这些表示帧号的整数坐标画出一些纵横线即可形成一个网格，网格中的每一个交点 (n, m) 代表着测试模板中的第 n 帧与训练模板中第 m 帧进行匹配或对齐。对网格中的交点进行选取形成一个规整函数 $m=\omega(n)$ 。规整函数必须满足一定的约束：

1. 必须是非递减的。即 $n_1 \leq n_2$ ，则 $\omega(n_1) \leq \omega(n_2)$ 。
2. 起点必须满足 $1=\omega(1)$ ，终点必须满足 $M=\omega(N)$ 。
3. n 需要取遍 $1, 2, \dots, N$ 所有的值， m 需要取遍 $1, 2, \dots, M$ 所有的值，即不能跳过某些帧的特征矢量。

如图 2.12 所示就是一条满足要求的对齐路径。

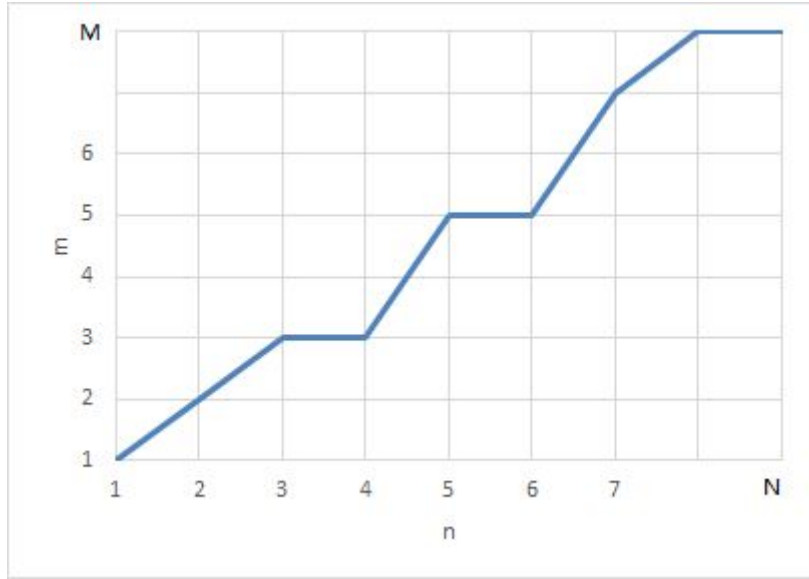


图 2.12 二维坐标表示

从上图可以看出，DTW 就是把时间规整和距离测度计算结合起来。测试语音参数共有 N 帧矢量，而参考模板共有 M 帧矢量， N 和 M 不等，寻找一个时间规整函数 $m=\omega(n)$ ，它将测试矢量的时间轴 n 非线性地映射到模板的时间轴 m 上，并使该函数 $\omega(n)$ 满足：

$$D = \min_{\omega(n)} \sum_{n=1}^N d[T(n), R(\omega(n))]$$

公式中， $d[T(n), R(\omega(n))]$ 第 n 帧测试矢量 $T(n)$ 和第 m 帧模板矢量 $R(\omega(n))$ 之间的欧氏距离测度， D 则是处于最优时间规整情况下两向量的距离。

2.4.3 隐马尔可夫模型

隐马尔可夫模型是一个双重随机过程，其中一个随机过程不可以观测，它是隐匿的，但是能通过另一个随机过程产生的可观测的符号序列来感知另一个不可直接观测的随机过程，即通过一个随机过程去感知另一个随机过程的存在及其特征^[36]。

一个 HMM 模型可用下列参数描述：

(1) N ：模型中状态的数目。一般来说，状态与状态之间是相互关联的，任意一个状态都可以由另外任意一个状态转化而来。状态集合记为 $S=\{S_1, S_2, \dots, S_N\}$ ， t 时刻的状态记为 q_t 。

(2) M ：每个状态可能观测到的符号数。每个状态可能会得到的观测符号的数量。观测符号集合记为 $V=\{V_1, V_2, \dots, V_M\}$ 。

(3) T ：观测符号序列的长度。模型产生的观测符号序列记为 $O=\{O_1, O_2, \dots, O_T\}$ 。 t 时刻的观测符号记为 O_t 。

(4) **A**: 状态转移概率矩阵。其元素 a_{ij} 意义为 t 时刻状态为 S_i , 而在 $t+1$ 时刻为状态 S_j 的转移概率, 即

$$A = (a_{ij})_{N \times N}, \quad a_{ij} = P(q_{t+1} = S_j | q_t = S_i), \quad 1 \leq i, j \leq N$$

(5) **B**: 状态 S_j 的观测符号概率矩阵。它是状态 S_j 的可能输出的观测符号概率构成的矩阵, t 时刻处于状态 S_j 时, 其元素 b_{jk} 是状态 S_j 输出观测符号 V_k 的概率。即

$$B = (b_{jk})_{N \times M}, \quad b_{jk} = P(O_t = V_k | q_t = S_j), \quad 1 \leq j \leq N, 1 \leq k \leq M$$

(6) **π** : 初始状态概率向量。 $t=1$ 时各个状态的概率。即

$$\pi = (\pi_1, \pi_2, \dots, \pi_N), \quad \pi_i = P(q_1 = S_i), \quad 1 \leq i \leq N$$

这样, 可以将一个完整的隐马尔可夫模型表示为:

$$\lambda = (N, M, \pi, A, B)$$

在隐马尔可夫模型中, 由于 N 和 M 比较好确定, 并非模型关注和研究的要点, 而 A, B 和 π 的取值对语音识别效果将产生直接影响, 是影响模型的重要因素, 所以隐马尔可夫模型可简化记为:

$$\lambda = (\pi, A, B)$$

由上面的各项参数, 隐马尔可夫模型产生观测符号序列 $O = \{O_1, O_2, \dots, O_T\}$ 的步骤如下:

- (1) 根据初始状态概率向量 π 决定一个初始状态 $q_1 = S_i$ 。
- (2) 设置 $t=1$ 。
- (3) 根据状态 S_i 的观测符号概率矩阵 b_{ik} 决定一个输出符号 $O_t = V_k$ 。
- (4) 根据状态 S_i 的状态转移概率矩阵 a_{ij} , 转移到一个新的状态 S_j , 即 $q_{t+1} = S_j$ 。
- (5) 令 $t=t+1$, 如果 $t < T$, 则回到步骤 (3), 否则过程结束。

以上的过程可以用于产生观测符号序列, 也可作为已知观测符号序列的隐马尔可夫模型。

HMM 模型的三个基本问题

在实际的应用场景中, 比如应用于语音识别等, 隐马尔可夫模型需要解决三个基本的问题:

(1) 问题 1: 评估问题

已知观测序列 $O = \{O_1, O_2, \dots, O_T\}$ 和模型 $\lambda = (\pi, A, B)$, 如何计算 $P(O|\lambda)$? 即已知模型和输出观察序列, 如何计算在模型 λ 下, 生成观测序列的条件概率。可以将其视为一个模型和特定观测输出序列的匹配度的评估, 因此可以用来在一系列候选输出观察序列中选取匹配度最好的输出观察序列。

(2) 问题 2: 解码问题

已知观测序列 $O = \{O_1, O_2, \dots, O_T\}$ 和模型 $\lambda = (\pi, A, B)$, 如何求得最优的状态序列 $Q = \{Q_1, Q_2, \dots, Q_T\}$ 。该问题即估计模型产生观测序列的各种可能的序

列，并且选择一条最佳的序列。在实际中，使用优化策略来尽量地解决这个问题。

（3）问题 3：学习问题

在模型中，如何调整模型参数 $\lambda=(\pi, A, B)$ ，使得一个固定观测序列 $O=\{O_1, O_2, \dots, O_T\}$ 的概率最大化，即使得条件概率 $P(O|\lambda)$ 取最大值。这个问题是三个问题中最难解决的一个，因为对于最大似然模型的分析，并没有一个很好的解决方法。该问题作用就是优化模型中参数，更好的描述已知的观测序列。

2.5 本章小结

本章介绍了智能手机终端 Android 操作系统，分析了 Android 系统的系统架构、平台的优势以及该平台的安装使用。介绍了 Qt 平台，分析了 Qt 平台的特点，说明了 Qt 的类库的组成模块，同时，说明了 Qt 平台的安装使用。然后详细说明了客户端与服务器端模式（C/S 模式），研究了语音识别的关键技术，详细介绍了动态时间规整和隐马尔可夫统计模型。为后续研究、设计和实现做好铺垫。

第 3 章 系统分析

本章将在前两章的铺垫下，对系统的可行性进行分析，以及系统的需求进行分析，为系统的后续工作打好基础，并且本章的内容是整个系统的重要内容，如果系统分析没有做好，将导致后续的工作走向错误的方向。

3.1 可行性分析

可行性分析主要包括技术上的可行性，经济上的可行性，以及环境的可行性。可行性分析是系统分析的重要环节，如下是各个方面的可行性的详细内容。

3.1.1 技术可行性

本系统采用 Qt 平台开发自助购物服务端，采用 Java 语言在 Android 平台上进行开发手机客服端，本地的数据库采用 xml 文档，通过在 Qt 平台中调用操作 xml 文档来进行数据的访问，将用户所需要进行交互的数据，通过 xml 数据库来达到与其他关系型数据库同样的效果；条形码的扫描使用现有的设备即可；由于 Java 语言是一种使用广泛的语言，所以 Android 平台的应用开发相对容易；网络方面，当下互联网技术以趋于成熟，网络的连接和 WiFi 热点的配置也不成问题；手机支付在技术上只要商家提前签约相应的银行和信用中心，并对接银行的标准接口，由银行负责提供并授权使用。同时系统架构采用 C/S 结构模式进行搭建，安全性好速度快，开发工具选用的是 Visual Studio 2008。另外上一章说明的相关技术也为系统的实现提供了技术支撑，因此，本系统在技术上是可行的。

3.1.2 经济可行性

为了提高购物效率、适应当前零售业快速扩张、用户要求的逐渐提高和零售业发展的新要求，零售业加快自动化、信息化已成为现代零售业的必然选择，自动化也成为零售业现代化的重要表现形式。信息和互联网技术在现代企业管理中的应用十分广泛，并且在提高管理效率、节省人力资源、减少人力成本、降低出错概率等方面获得显著效果。将信息技术应用到超市管理上，不仅是社会发展的趋势，也是企业做大做强的机遇。因此，传统零售业只有更新观念，加快实现企业信息化、自动化进程，充分利用当代信息和互联网技术才能在经济发展的潮流中有自己的一席之地。当前世界，一个企业是否完成信息化、自动化已经成为一个企业是否实现现代化的重要标准，国内企业在信息化、自动化方面一直落后于发达国家，而伴随着国内零售业的高速发展，为了适应快节奏的城市生活，超

市作为零售业的主体，信息化、自动化已势在必行。

系统采用的搭建方式为 C/S 结构框架，有多台自助服务端和多个客户端，而其中的客户端就是购物者手中的手机，系统通过本地网络进行连接并传输数据。系统构架清晰，规模适中，能很好满足零售业的收银需要，硬件方面不需要特别的投入，只需要在软件、网络以及研发费用上进行一次性的投入，同时，收银自动化可以解放收银员，大大减少企业的人力成本，该系统的使用和维护相当的方便、快捷。前期有一部分的投入，但是后期却有很大的好处。所以，本系统在经济上是完全可行的。

3.1.3 环境可行性

作为一个自动化的收银系统，它的构建和实现需要良好地环境。对于本系统需要考虑的环境主要有如下几个方面。

首先，购物者具备不具备使用该自动化的收银系统的能力。在这方面需要考虑到所有来购物的人群，而该智能收银系统完全的兼容和兼有旧系统，所以不会使用的人群可以使用人工收银系统。

其次，完全自动化对小偷、抢劫、搞破坏的人怎么办。在安全方面确实需要考虑。近来，国内人民的素质有很大的提高，是不容置疑的；社会公共安全国家也重视，公共安全措施也齐全，因此，如上问题有可信任的国家作为后盾。

最后，社会对收银自动化、信息化、智能化的态度和认同程度。现在国家在推进经济转型，正是大力发展第三产业的时候，因此，国家会给予极大的支持和帮助，社会的自动化、信息化、智能化也是发展趋势是人民的选择，因此，对于收银系统的自动化，社会和人民应该是欢迎的。

综上可知，本系统的构建和实现拥有良好的环境，在环境上是有优势的，是可行的。

3.2 系统需求分析

系统需求分析包括系统角色的分析，功能需求分析，以及性能需求分析等。如下是各个系统需求的具体分析。

3.2.1 角色分析

如果有一组人，他们的所需的权限是一样的，当对他们的权限进行管理的时候会很不便，因为你要对这一组中每个用户的权限都进行管理。有一个很好的解决办法就是使用角色。角色是一组权限的集合，将角色赋给一个用户，这个用户就拥有了这个角色中的所有权限。那么上面的问题就很好处理了，只要第一次将角色赋给这一组用户，接下来就只要针对角色进行管理就可以了。以上是角色的

一个重要用途。角色在一定程度上保证信息的安全，各类信息不会被随意访问和修改，理解角色就是一组权限的集合是很重要的点。下面对本系统中用户的角色进行说明。通过对系统的使用者进行调查，得到两类系统角色，分别是普通用户（购物者）、系统管理员（超市的管理人员）。

普通用户涉及的功能：普通用户的操作主要是条形码扫描获得商品的编码，检索商品信息以查看商品的单价及其他信息，需要 WiFi 热点进行网络的连接，在商品确定之后生成账单，使用手机客户端进行支付，同时，在操作的过程中可以使用语音进行操作等。

系统管理员涉及的功能：系统管理员功能包含整个系统的管理，操作在自助购物服务端进行。系统的启动、关闭等操作，管理整个系统的维护与设定。也需要条形码扫描、检索、修改、添加商品信息和语音操作等。系统具体用例图如图 3.1 所示。

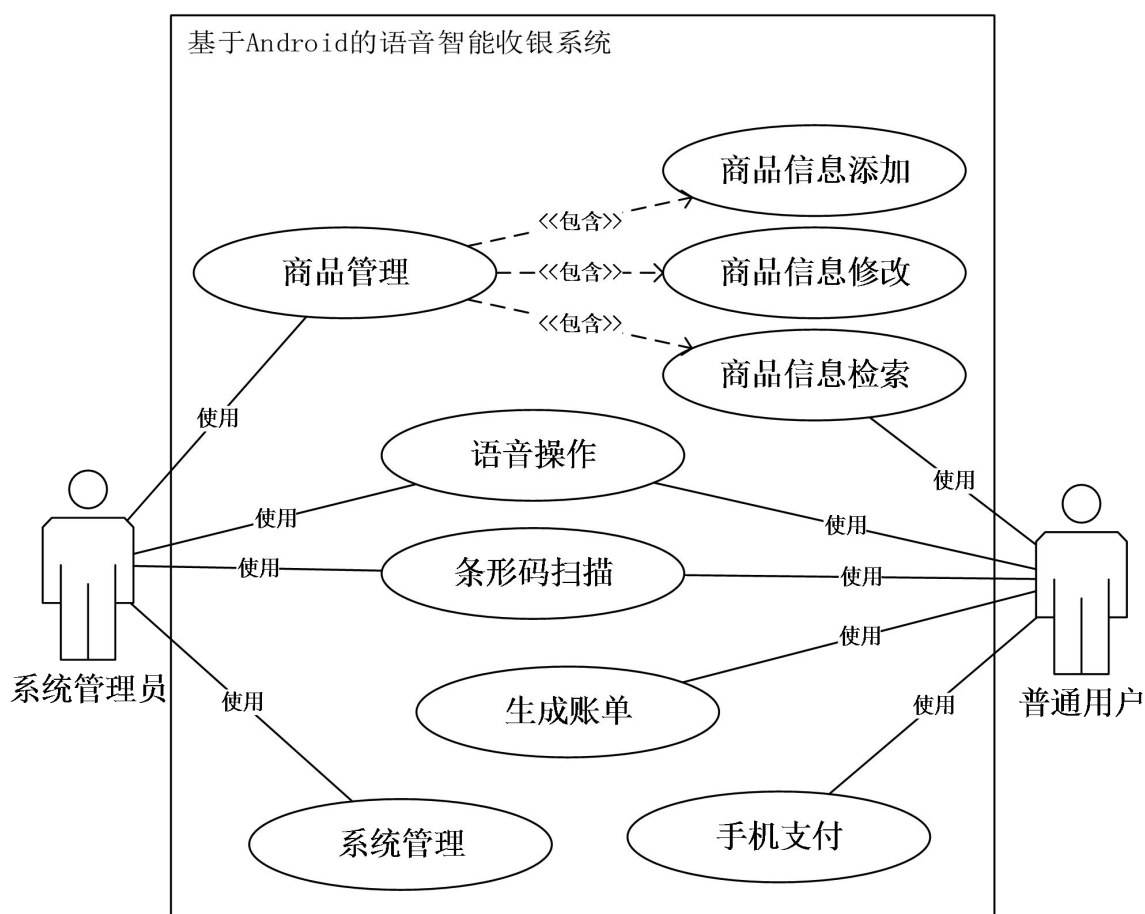


图 3.1 系统角色及功能的用例图

语音智能收银系统主要涉及两类用户，具体是系统管理员，普通用户，管理员通过自助服务端登陆系统，普通用户通过手机客户端登陆系统，进入系统后可以完成各自不同的功能操作，如此可以提高用户各自使用的效率，同时也能提高系统的安全性能。

3.2.2 功能需求

系统整体图如图 3.2 所示，当商家使用该系统时，通过扫描条形码生成相应的商品编号，在人工录入商品名称和单价等信息，保存进数据库中。当顾客挑选完商品后，扫描商品条形码获取商品编码，通过商品编码查询 XML 数据库，获取商品的名称、单价等信息。将多种商品加入到购物车，生成账单信息，便可将账单发送到手机，使用手机支付完成交易，再返回是否支付成功的消息。系统后台对命令进行处理。

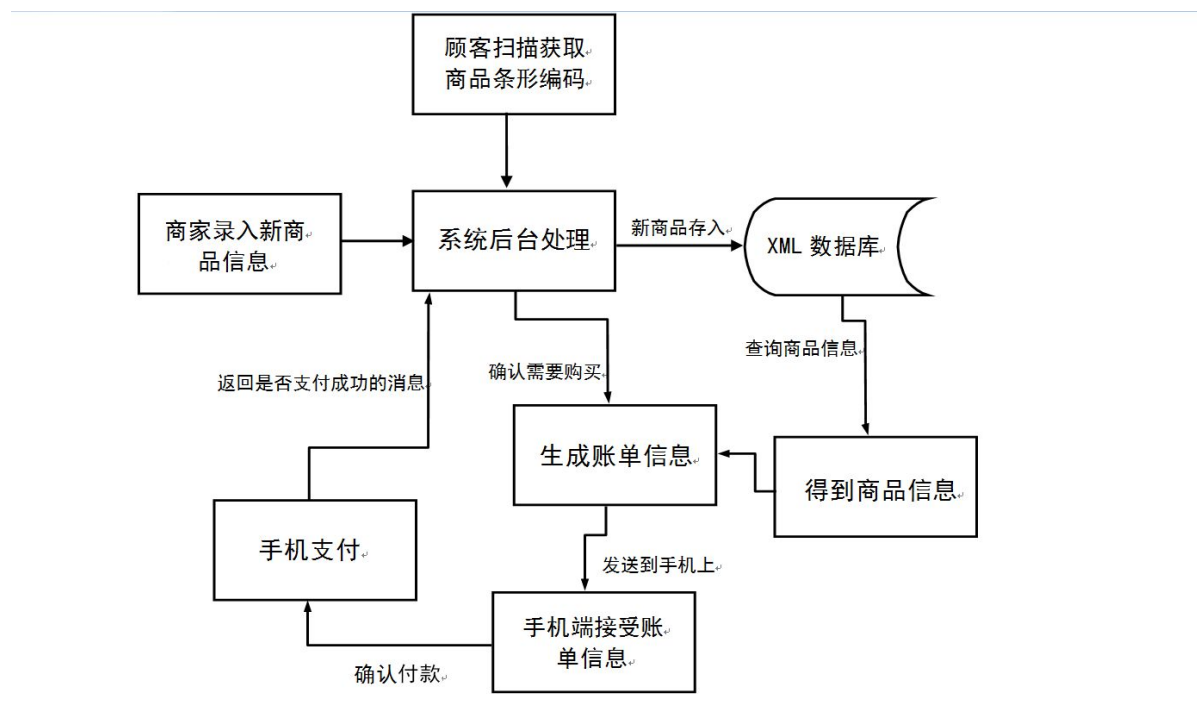


图 3.2 系统整体图

使用该系统的人员可分为两类：一类是商家；另一类是顾客。对不同类的使用者自然有不同的需求。商家需要对新商品进行名称，单价等信息的输入，而顾客需要完成商品扫描和结算。本系统需主要实现以下功能：

（1）系统要能够实现对商品条形码扫描，并将扫描的条形码转换成机器能识别的语言。

（2）商家扫描新购进的商品条形码，并将扫描结果保存为相应的商品编号，并输入商品的名称、单价等信息，将其添加进商品信息库。

（3）用户扫描输入商品条形码，搜索商品信息库，得出相应的商品名称和单价等信息。用户通过交互界面进行查看。

（4）系统要设置 WIFI 热点，并作为服务端接受来自手机客户端的连接。

（5）用户可以同时购买多种商品，并一次性付款。使用购物车，将要购买的商品都扫描入购物车，并生成账单信息发送到手机客户端。

（6）用户使用手机客户端连接到服务端，接受账单信息，手机支付实现。

(7) 系统能够识别简单的语音输入，并执行相应的操作，像：加入购物车，付款等。

(8) 系统输出流利的汉语语音，如：您已成功付款等。

3.2.3 性能需求

系统性能方面主要需求是系统响应速度要快，稳定性高和用户体验好等重要方面。系统应该做到高稳定性、快速响应、操作简单易用等优点。

(1) 系统响应及时且快速。系统强调的就是自动化和信息化，如果在效应速度上很慢，没有达到为购物者节约时间的目的，那该系统就没有存在的必要了。在实际使用中，客户对机器系统的容忍程度远没有对人工收银系统那么好。因此，系统需要及时响应用户的操作或返回相应的状态信息，不能出现类似死机的状态。

(2) 系统需要相当的抗干扰能力。当硬件故障或是系统断电等原因导致异常，系统需要保证数据的正确性。

(3) 用户体验要好。系统需要界面简单，操作方便，使得用户容易学会，乐于使用。

(4) 系统稳定性要高。用户一开始并不是很了解系统，所以操作可能是错误的，对于用户的错误操作不会轻易的陷入未知状态，在错误操作时有错误信息。

3.3 本章小结

本章在系统方案总体制定的基础上，对于基于 Android 的语音智能收银系统，在技术、经济和环境等方面进行了详细的可行性分析。然后分析系统的角色需要的功能，从整体上分析系统的功能需求和性能需求，完整的表明了系统的真实需求，调研了系统的总体业务流程，为本系统的之后的设计打下坚实的基础。在下一章，将对基于 Android 的语音智能收银系统进行详细的设计。

第 4 章 系统设计

系统设计是构建在系统分析的基础上的，根据系统的需求对系统进行设计。同时，系统设计也是系统实现和测试的基础，对系统的实现影响很大，如果系统设计的不完善，将会使得系统的实现困难重重、举步维艰。以下是系统的详细设计，包括系统构架设计以及系统的功能模块设计等。

4.1 系统架构设计

系统涉及到路由器、wifi 热点、数据服务器、常用的 pc、智能手机等硬件设施，它们之间的网络拓扑结构如图 4.1 所示。

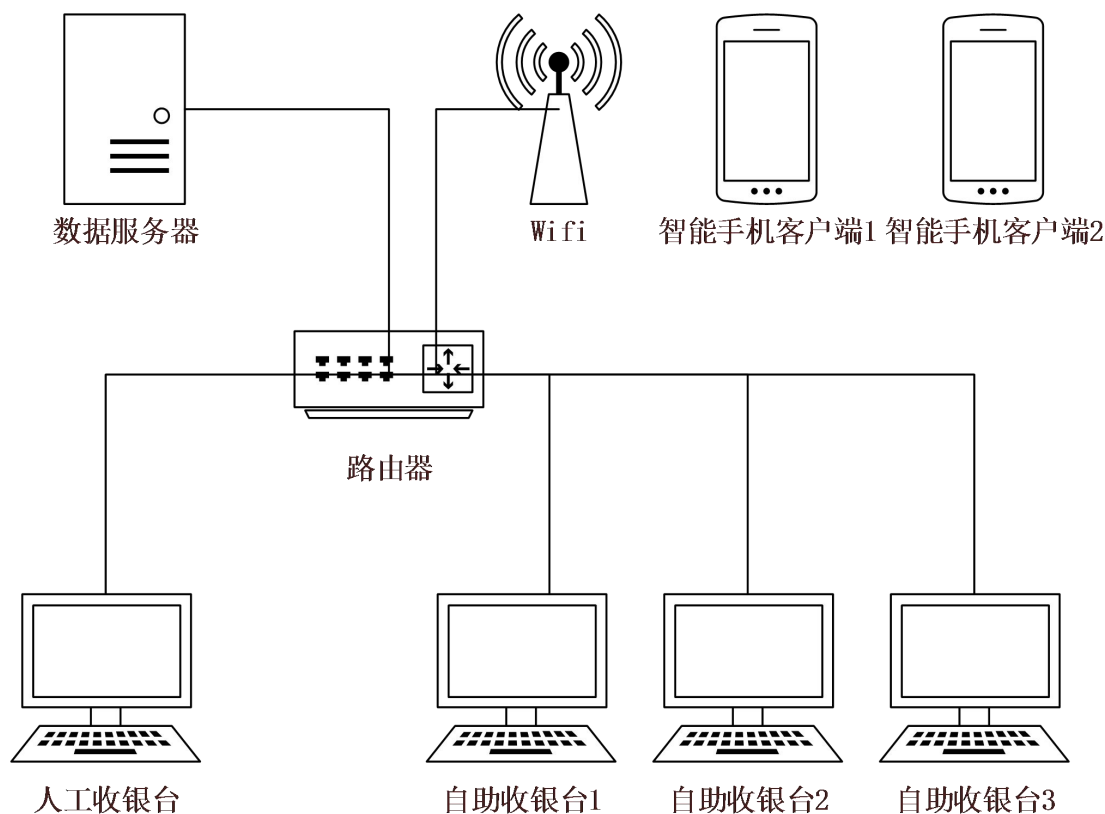


图 4.1 系统网络拓扑图

在软件体系架构的设计中，分层结构是最易用的和常见的，同时，也是最重要的一种结构。本系统采用的是三层的 C/S 架构，三层结构是传统的客户/服务器结构的发展，通过分层，可以限制子系统间的依赖关系，使系统以更松散的方式耦合，从而更易于维护。应用程序要解决三个层面的问题，分别是数据访问层、业务逻辑层、表示层。

表示层（UI）：通俗讲就是展现给用户的界面，即用户在使用一个系统的时候他的所见所得。在本系统中，表示层主要是手机客户端显示信息的页面和自助

服务端显示信息的页面。

业务逻辑层（BLL）：针对具体问题的操作，也可以说是对数据层的操作，对数据进行业务逻辑处理。在本系统中，业务逻辑层主要是语音控制部分和自助服务端业务逻辑部分。

数据访问层（DAL）：该层所做事务是直接操作数据库，针对数据的增添、删除、修改、检索等而设置的层。在本系统中，数据访问层主要是条形码扫描、xml 数据操作和支付操作。

由以上分析，系统的软件架构如图 4.2 所示。

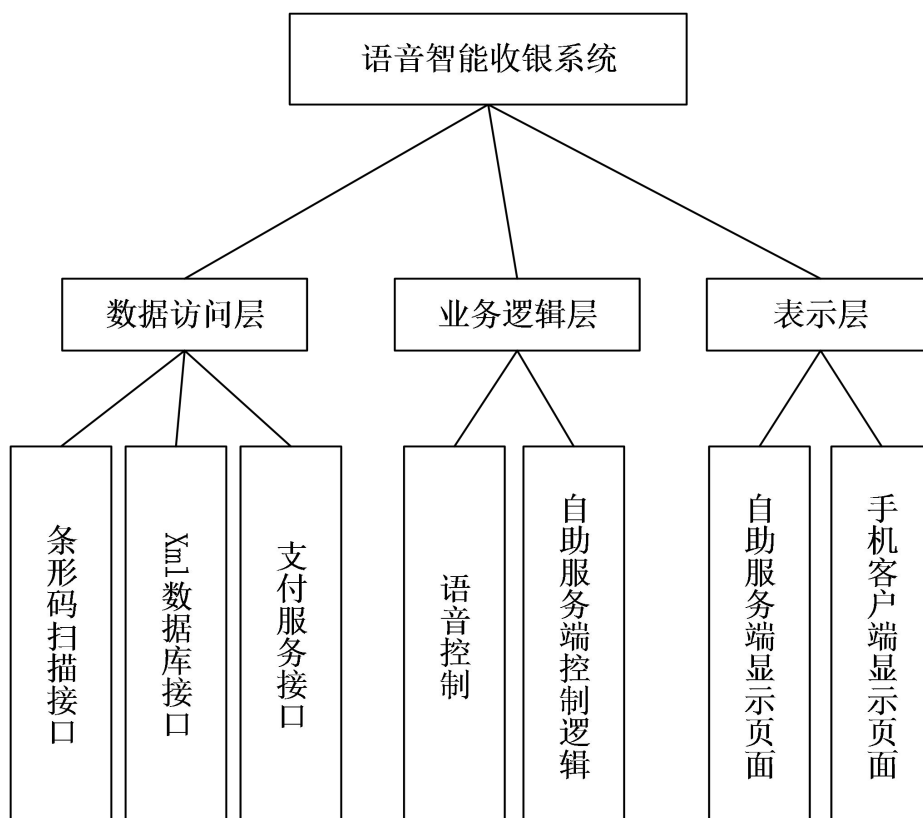


图 4.2 系统软件架构图

4.2 系统功能模块设计

系统的功能模块设计包括对系统的功能进行模块的划分，同时对模块进行高效的组织，形成层次结构。然后，对模块的流程进行相应的设计。

4.2.1 功能模块划分与组织结构

语音智能收银系统主要包括自助购物服务端和手机客户端两大模块，自助购物服务端主要的功能模块有条形码扫描、购物车生成、账单生成、语音操作；手机客户端主要的功能模块为语音操作、账单获取、货款支付。本系统总体的功能模块如图 4.3 所示。

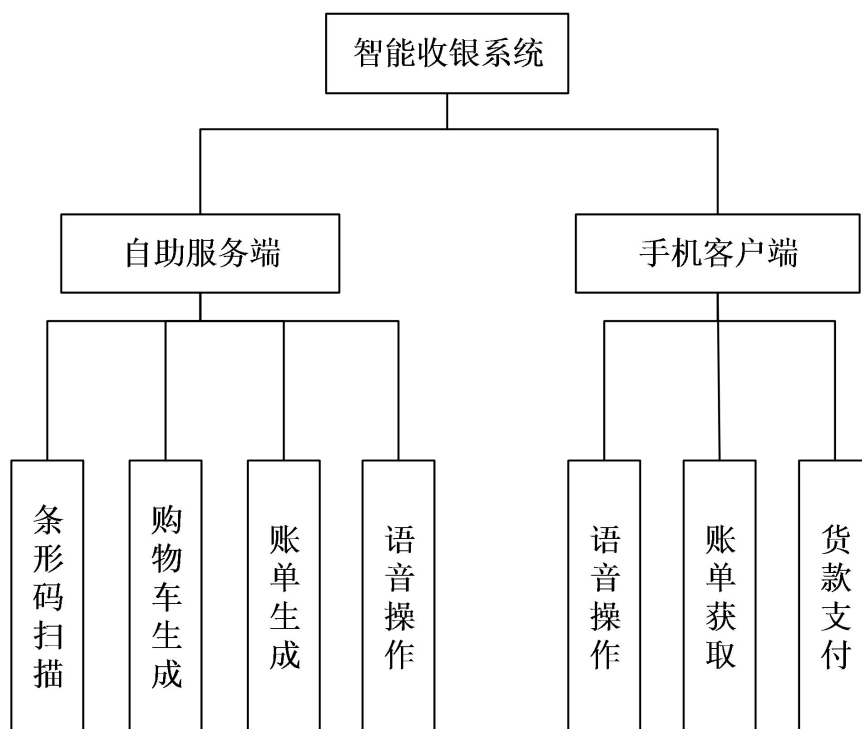


图 4.3 系统功能模块图

条形码扫描：用于获取商品的编号。

购物车生成：获取商品的名称、单价等信息，顾客判断是否购买确定是否加入购物车。

账单生成：将账单生成并存储到系统中，等待购物者使用手机获取，然后进行手机支付完成交易。

语音操作：当在自助购物服务端和手机客户端使用系统时，可以使用相应的语音对系统进行操作。

账单获取：获取来自自助购物服务端的账单信息，查看需要付款的款项是否正确，以准备下一步的支付操作。

货款支付：使用银行的支付接口进行支付，不管已支付成功或支付不成功，都将反馈该信息给自助购物服务端，然后由自助购物服务端通知手机客户端。

4.2.2 功能模块详细设计

1. 自助购物服务端

该部分是客户交互界面，完成客户商品扫描，得出具体的账单信息，并实现账单发送功能。

该部分的基本流程如图 4.4 所示，该部分实现的基本功能有：商品信息的录入，商品条形码扫描，显示扫描的商品的信息，加入购物车，查看购物车的商品信息，清空购物车，发送账单等。

卖家将商品条形码、种类、单价等信息录入形成商品信息库；当顾客使用时，

使用条形码扫描器扫描选购的商品，获取该商品的条形码，并在商品信息库中查询获取对应的商品名称、单价等信息，通过用户交互界面呈现给客户查看；客户如果不想购买商品，系统等待扫描下一件商品；如需购买，则将其加入购物车中；当扫描完选购的商品后，可进入购物车查看购物车信息，再此确定要买的商品，可以将不想购买的商品信息删除，留下的商品确认购买，则生成账单，并将账单数据发送给手机客户端。

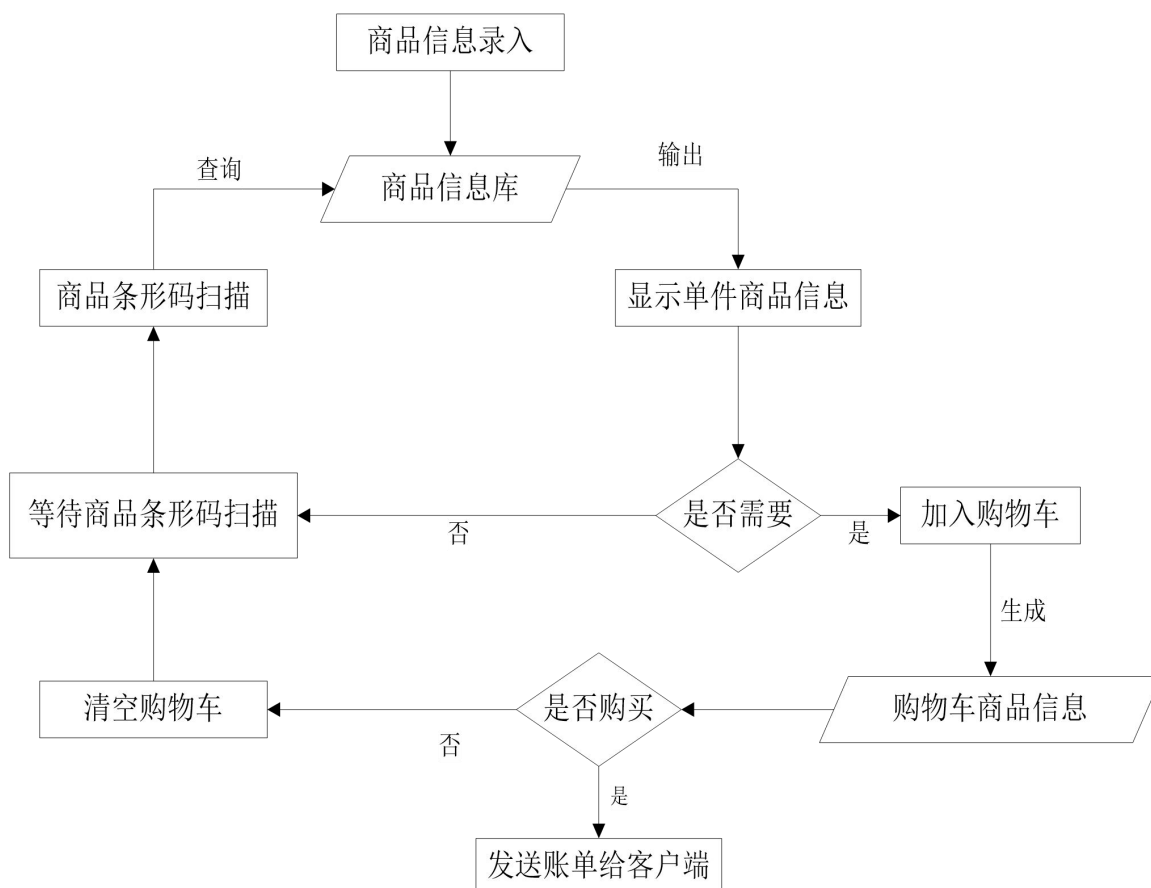


图 4.4 自助购物车工作流程图

2. 手机客户端

手机客户端提供用户手机支付功能，用手机支付的方式完成交易。

该部分的工作流程如图 4.5 所示，实现的基本功能有：查询以往账单信息，删除以往账单信息，保存账单信息，连接到自助购物服务端，接受来自自助购物服务端的账单信息，显示账单信息，确认付款，登录手机支付账户，支付，反馈支付成功的消息等。

客户打开手机应用后，可以查看、删除保存在手机中的以往的账单信息，如果需要支付，确认手机 WLAN 连接功能是否打开，打开后，将手机连接到服务端设置的 WIFI 无线网；如果需要购买，在服务端确认购买后，服务端将会发送账单信息，此时，手机客户端接受账单信息，解析后将账单通过交互界面显示给客户查看；确认付款，发送客户的账户、密码信息给手机支付运营商服务主机，进

行账户登录，再发送需要结算的金额和付款的商家信息给运营服务主机，支付相应的金额给超市账户；若支付成功，则反馈支付成功的信息给自助购物服务端；自助购物服务端再发送给手机支付成功的信息，可以作为客户诚信交易的凭证。

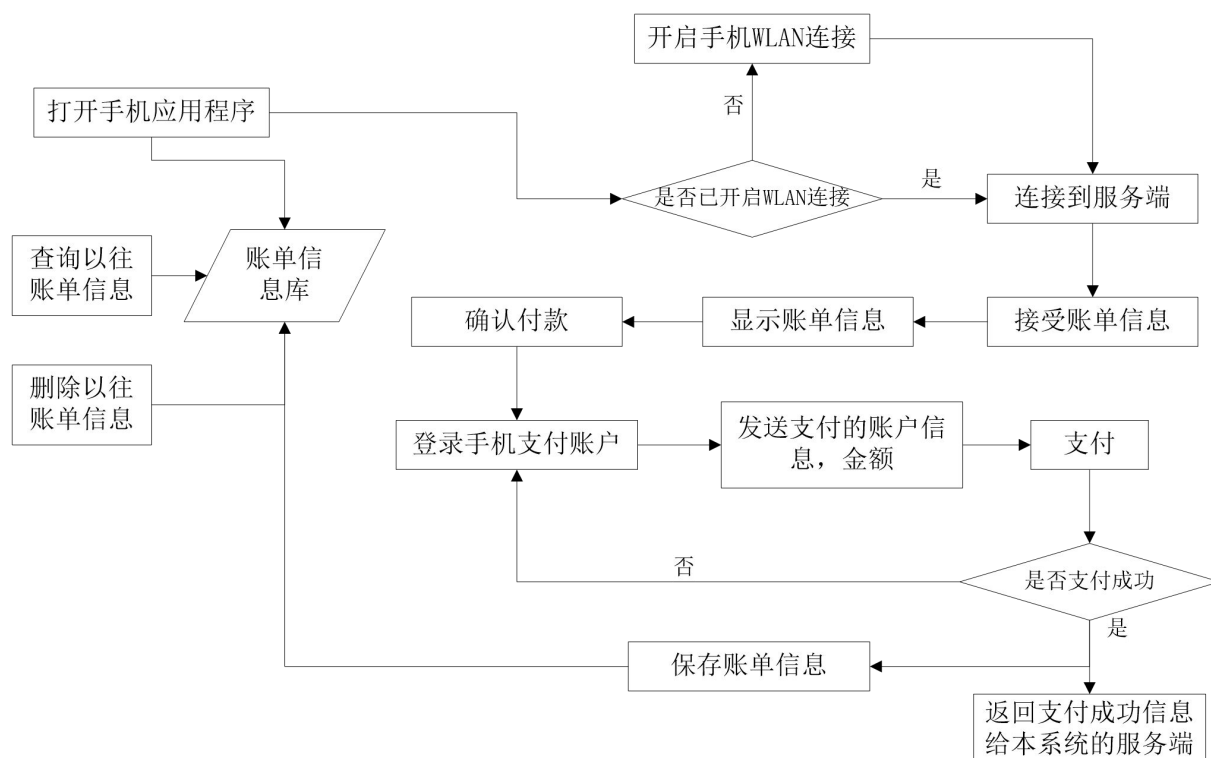


图 4.5 手机客户端工作流程图

4.3 本章小结

本章对于系统的架构进行了详尽的设计，包括系统的网络拓扑结构设计，以及软件的架构设计。具体的分析说明了系统中具体模块功能的划分和组织结构，并对各模块作用进行了说明。然后分别对手机客户端和自助服务端进行流程分析和设计。

第 5 章 系统实现与测试

本节对系统的实现进行了详细的说明，首先搭建了系统的开发环境，然后对自助购物服务端、手机客户端进行实现，最后对实现的各个模块进行了严格的测试，并对测试过程进行了详细的阐述。

5.1 开发环境与运行环境

软件开发环境(Software Development Environment, SDE)是指在基本硬件和数字软件的基础上，为支持系统软件和应用软件的工程化开发和维护而使用的一组软件，简称 SDE。它由软件工具和环境集成机制构成，前者用以支持软件开发的相关过程、活动和任务，后者为工具集成和软件的开发、维护及管理提供统一的支持。运行环境(Runtime Environment, RE)包括软件和硬件两方面。软件方面主要指操作系统及一些其他软件，还有第三方运行库等，硬件方面主要是计算机的配置，如 CPU、内存、显卡、硬盘等等。

5.1.1 开发环境

基于 Android 的语音智能收银系统的开发环境如下：

- (1) 开发架构：采用 C/S 架构
- (2) 开发语言：C++，Java
- (3) 数据库：XML 数据库
- (4) 开发工具：Visual Studio 2008，Eclipse
- (5) 开发时自助服务端操作系统：Windows7
- (6) 开发时手机客户端操作系统：Android

通过对以上各项的灵活运用和结合来完成系统的开发。

使用装有 windows7 操作系统的计算机，在其上装上 Visual Studio 2008 和 Eclipse，然后使用 C++语言在 Visual Studio 2008 中完成对 XML 数据库的操作和业务逻辑的处理，以及 Qt 界面的开发，完成自助服务端的开发。

使用 Java 语言在 Eclipse 中完成对自助服务端的连接和各项数据的获取，以及支付等操作，完成手机客服端的开发。

5.1.2 运行环境

本系统的开发结束后，产生一个手机客户端和一个自助服务端。运行需要有

相应的硬件设备和相应的软件设备，同时，需要网络支持，必须先安装相应的底层软件，pc 操作系统和手机操作系统等。详细的运行环境要求如下：

1. 手机客户端

智能手机：1 部以上；

网络：WiFi 热点；

操作系统：Android OS 2.3 以上；

CPU：0.68GHz 以上；

RAM 容量：512MB 以上。

2. 自助购物服务端

商用 PC 机或终端：1 台以上；

CPU：P4 3.0G 以上；

内存：2GB 内存以上；

硬盘空间：200G 以上；

网络：局域网；

数据库：XML 数据库；

数据库版本：无；

操作系统：windows 7；

5.2 自助购物服务端实现

5.2.1 条形码扫描实现

新建一个 Qt Quick UI 工程 ServerBill，如图 5.1 所示。

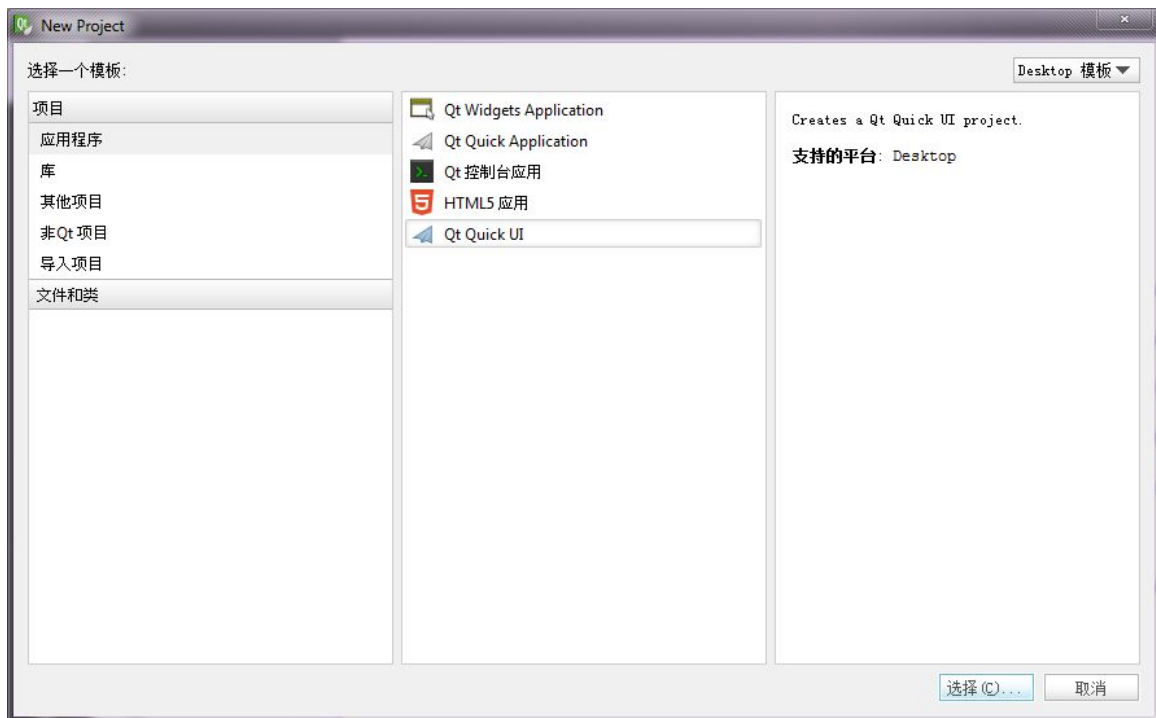


图 5.1 新建 Qt Quick UI 工程

用户交互的界面主要有三个：新商品存档界面、顾客商品扫描信息显示界面和购物车界面。界面设计如下所示：

（1）新商品存档界面

该界面在 `AddGoods.qml` 中实现，如下图 5.2 所示。该界面通过二维码扫描输入商品编号，手动输入商品名称与单价，通过“加入新商品”按钮，将其该新商品的信息存档。界面设计使用了 `qml` 的 `Rectangle`, `TextEdit`, `Text` 等控件，使用了 `Column`, `Row`, `gradient` 等布局。

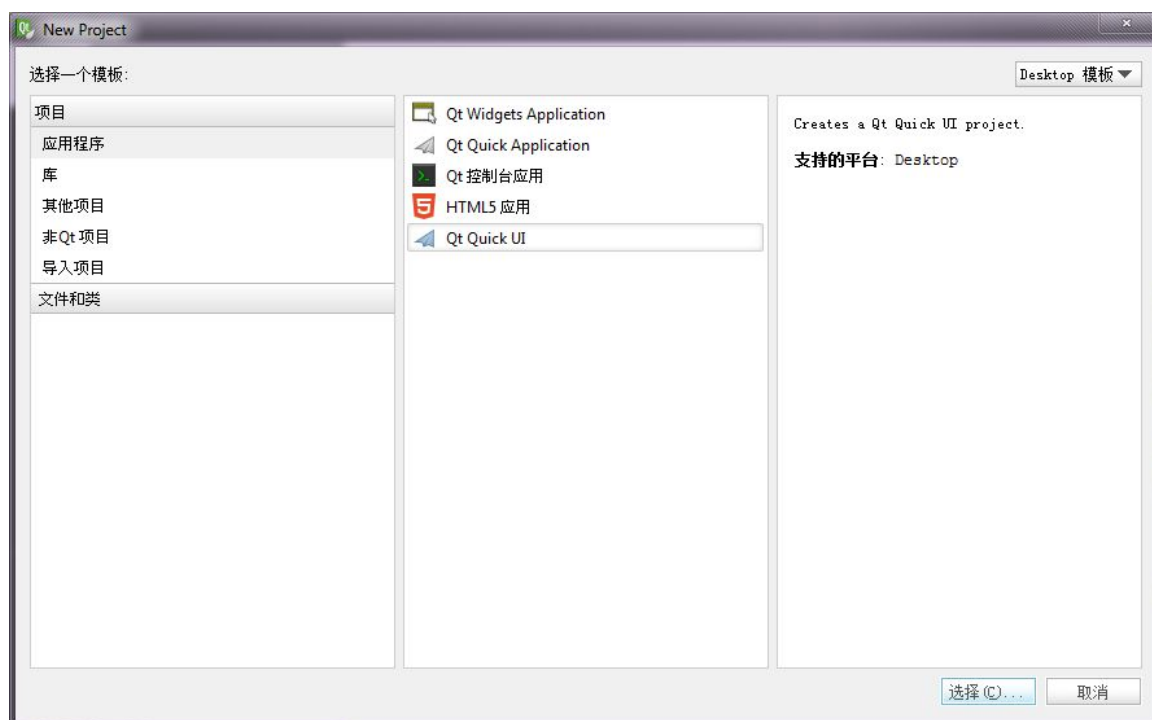


图 5.2 新商品存档交互界面

(2) 顾客商品扫描界面

顾客商品扫描界面在 Input.qml 中实现,用于当客户通过条形码扫描器扫描商品得到计算机可读编码后,查询商品信息库,查询对应商品信息,显示于该界面之上。该界面显示如图 5.3 所示。

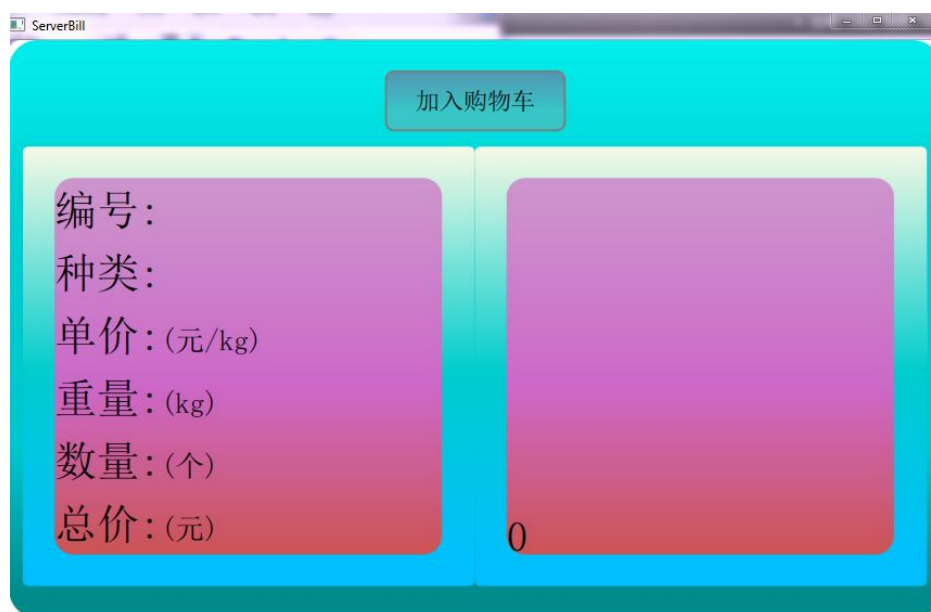


图 5.3 顾客商品扫描界面

该界面由 qml 的 TextEdit、Text、Rectangle 等控件, Column、Row 等布局,以及“加入购物车”按钮组成。当扫描一件商品时,则查询商品信息库,得到商品信息,并显示与该界面上,如果顾客需要,则通过“加入购物车”按钮触发信号,

并将其数据加入购物车。

5.2.2 购物车实现

该界面在 ShoppingCar 中实现，效果如图 5.4 所示。由 TextEdit, Rectangl, Text, ListView, ListModel 等组成，当有商品被加入购物车时，则将商品信息保存到该购物车中。“付款”按钮被点击时，生成账单信息。

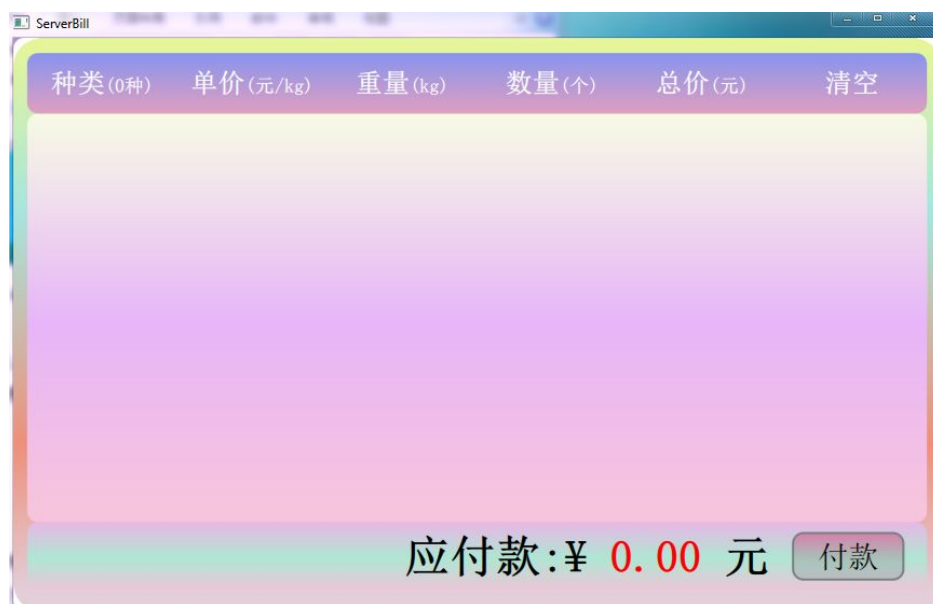


图 5.4 购物车界面

各个界面实现好后，需要对其进行业务逻辑的实现，通过注册类可以将类使用到界面上，进而将界面与后台的业务实现代码联系到一块，完成相应控件的业务逻辑实现：

该 qml 加载了前面说到的三个 qml 界面，使用 VisualItemModel 进行加载：

```
VisualItemModel {
    id: visualModel
    Input {
        id: idInput
        界面大小设置.....
        加入购物车信号处理.....
    }

    ShoppingCar {
        id: idshoppingCar
        界面大小设置.....
        付款信号处理.....
    }
}
```

```

AddGoods {
    id: idAddGoods
    界面大小设置.....
    加入新商品信号处理.....
}
    
```

用 ListView 进行显示，设置该 ListView 的 model 为 visualModel，QT 自动为 idInput、idshoppingCar 和 idAddGoods 设置 Index 分别为 0，1，2。

添加三个按钮“商品信息”，“进入购物车”和“加新商品”，当三个按钮单独被触发时，分别设置 ListView 的 currentIndex 为 0 或 1 或 2，则显示对应的界面。如图 5.5 所示。



图 5.5 main.qml

以上三个界面的按钮实现都是在此 qml 上。例如，在购物车界面中，如果点击“付款”按钮，则，发出 *payButtonClick()* 信号，在 main.qml 中如果有 onPayButtonClick 信号发生，则处理对应的事件，如下代码示：

```

onPayButtonClick: {
    获取 idshoppingCar 中 ListView 的每项，形成账单信息
    将账单信息发送
    
```



```
}
```

当有顾客扫描商品时，系统搜索保存商品信息的 XML 文档，算法及伪代码如下：

```
QString readAttributeFromGoodsInfoXml(QString number,
                                     QString attribute )
{
    打开改 XML 文档
    file.open(.....)
    生成 DOM 树
    QDomElement docElem = doc.documentElement();
    // 根据节点名称读取某个节点
    QDomNodeList n = docElem.elementsByTagName( number );
    QDomElement node = n.at(0).toElement();    // 转化为节点
    return node.elementsByTagName( attribute ).at(0).toElement().text();
}
```

而当商家保存商品信息进入商品信息库的算法及伪代码如下：

```
bool addNewNodeToGoodsInfoXml(QString number,
                               QString type,
                               QString perPrice)
{
    商品信息有效时 .....
    连接打开商品信息库文档
    生成 DOM 树
    doc.setContent( &file..... )
    断开连接
    // 获取根节点 应该为 GoodsInfo
    QDomElement root = doc.documentElement();
    // 首先判断是否存在此商品
    QDomNodeList n = doc.elementsByTagName( number )
    创建对应商品的节点
    建立商品节点的子节点
    root.appendChild(element); // 添加新商品节点到根节点
    将 DOM 树内容写入商品
    QTextStream out( &f );
```

```
doc.save( out, 3 );  
f.close();  
return true;  
}
```

5.2.2 账单生成发送实现

QT 中的 Socket 使用了类的封装机制，是用户的网络编程更加方便，容易理解，再加上 QT 的 signal-slot 机制，编写的程序更容易理解。QT 中一共提供了四个与套接字相关的类，它们分别是：

QServerSocket: 基础的 TCP 服务；

QSocket: TCP 连接缓存；

QSocketDevice: 独立于平台的底层套接字 API；

QSocketNotifier: 套接字回调支持；

本系统需要将账单发送到手机客户端，在手机上支付完成交易，这过程需要服务端与手机客户端的通信。本系统采用 Socket 编程实现该功能实现步骤依次如下：

(1) 设置 WIFI 热点

WIFI，是一种可以将电脑或手持设备以无线^[37-38]方式相互连接的技术，它的本质就是一个高频无线电信号。

无线网络在无线局域网的范围内是指一种无线相容性认证，它的本质上就是一种商业认证。在电脑使用的开始阶段，电脑是通过网线进行有线连接进入网络，而 WIFI 使得网络连接可以通过无线电波来进行。最常见的无线连接方式就是使用无线路由器，在无线路由器电波所包含的范围内，都可以采用 WIFI 连接，而且如果无线路由器连接了一条上网线路，那么连接到该路由器的终端设备都可以通过该网络进行上网等操作。

本系统是在带有无线网卡的 window7 系统上实现，在实验和测试阶段可以使用 DOS 命令为系统服务端设置 WIFI 热点，虚拟一个无线网络。具体步骤如下：

1. 以管理员身份运行命令提示符。
2. 在命令行上输入 netsh wlan set hostednetwork mode=allow ssid=Pumpkin key=11111111,该命令是启动 win7 系统自带的虚拟无线网卡，其中 mode=allow 表示允许启动该设备的虚拟无线网卡。ssid 为连接网络时扫描的 WIFI 网络名称，“11111111”是无线访问密码，最少需要 8 位。
3. 打开“网络和共享中心”，选择更改适配器设置，会发现有一个虚拟的网络连接，则表示虚拟无线设备已经成功启动。
4. 共享 Internet 网络。

5. 再回到命令提示符下，启动虚拟无线服务，输入 `netsh wlan start hostednetwork`。

(2) 实现 QT Socket 编程

QT 中 `QTcpServer` 用于侦听来自网络的连接信号。当有新连接时，`QTcpServer` 发出 `newConnection()` 信号，并连接到槽 `acceptConnection()` 函数：`connect(&tcpServer, SIGNAL(newConnection()), this, SLOT(acceptConnection()))`（QT 的信号槽机制见文献[10]）。

在 `acceptConnection` 槽函数中，获取连接套接字并监听各种需要处理的信号，算法及伪代码如下。

```
void acceptConnection()
{
    获取套接字连接
    //连接断开时的信号槽关联
    connect ( tcpServerSocket, SIGNAL( disconnected() ), this,
              SLOT( disconnectToShow() ) );
    将受到数据的信号连接到数据接受处理槽函数
    将错误信号处理的信号槽函数关联.....
}
```

通信中定义了数据接受和发送的槽函数，算法及伪代码如下：

数据接受：

```
{
    //计算可接受数据的字节数
    qint64 len = tcpServerSocket->bytesAvailable();
    //QByteArray，接受 len 长度的数据
    inData = tcpServerSocket->read( len );
    //设置编码
    QTextCodec *codecUtf8 = QTextCodec::codecForName( "UTF-8" );
    QString utf8Bill = codecUtf8->toUnicode( inData.mid( 2 ) );
    if(接受到的数据是“付款成功”)
    {
        发送交易完成的凭证给客户手机端
        发出付款成功的信号
        emit payCompleted();
    }
}
```

```
}

```

发出数据：

```
{
    QDataStream sendData( &outData, QIODevice::WriteOnly );
    sendData << msg.toUtf8(); //将数据转化为 utf8 编码格式
    发送数据
    tcpServerSocket->write( outData );
}
```

5.2.3 语音操作实现

人的交流是通过听和说来实现的，要实现人与机器的交流，就必须使机器具有听和说的能力。而语音合成使得机器有了说的能力^[39]，而语音识别使得机器具有听得能力。使计算机具有类似于人一样的交流能力，已成为当今时代信息产业的重要竞争市场。

语音识别是将语言中的文字信息提取出来，转换成相应的文本信息或控制指令，使机器拥有类似“耳朵”的功能。语音识别的主要方法是模式匹配法，分为训练阶段和识别阶段。其中在训练阶段要做的事：将需要识别的词语说一遍，并获取它们的特征矢量，然后作为模板存入模板库中。另外在识别阶段，便可以将输入的语音的特征矢量依次与模板库中的模板进行一一匹配，获得相似度最高的作为识别结果返回便可。

语音合成是利用电子的和机械的方法来产生语音的。其中的 TTS 技术，又称文语转换技术，它能够计算机产生的或外部输入的文字转换为汉语口语并说出。

作为一种具有声调的语言，汉语非常复杂。对于同样一个音节，在不同的环境下，其声调参数是各不相同的。要使用有限的存储单元存储基本汉语的基本语音单元，进而从有限的存储单元中合成出无限词汇，类似于组合，形成连续的语句，则必须在一定的声调规则下对音库单元的参数进行调整，以得到符合当前语言环境的语音库单元。

该系统在操作时的按钮功能，都可以用语音指令来替代。语音识别与语音合成技术在最近几年得到了广泛的关注，许多科研人员在这方面投入研究，各种语音产品相应也随之而出。语音是科技走向智能化的一个很大的进步。语音识别与合成相当于人的听说。如果语音识别与合成技术得到成熟与完善，那我们在电影中看到的与机器人对话的炫酷场景将成为现实。未来的电子产品也会逐渐趋向于语音智能化。目前，语音识别与语音合成技术虽然得到了广泛的发展，但是因其

受各种因素的影响（如外界噪音、语种不同，方言等），其识别准确率还有待提高。

在本系统使用了科大讯飞的在线语音识别技术来实现语音模块。该系统语音模块的工作步骤大致为：语音采集->登录科大讯飞服务端->上传语音语法->上传需要识别的文件->接受识别返回结果->根据识别结果执行相应的措施。

(1) 语音采集

使用 QT 的 QAudioInput 定义变量 m_audioInput, 首先采集输入的语音文件，以 .raw 格式保存，以下为伪代码实现。

```
{
    设置保存语音输入的音频文件为.raw
    初始化语音采集的速率、频道、大小、类型等因素
    开始采集.....
}
将.raw 格式转换为.wav 格式的音频文件：
{
    新建或打开.raw 和.wav 文件
    设置语音采集的一些参数
    .....
    将.raw 格式中的数据读入.wav 格式中.....
}
```

(2) 识别阶段

首先需要在科大讯飞端注册自己的账户，再登录上传语音语法和需要识别的 .wav 音频文件，实现伪代码如下：

```
{
    设置登录参数 "appid = 5343cd27, work_dir = . . . . .";
    选择待识别的语音文件.....
    设置登录账号与密码，并登录
    MSPLogin(用户名，密码，登录参数);
    获取语法文件为.abnf 格式文件
    上传待识别的音频文件和语法文件
    获取识别结果.....
    return 识别结果
}
```

(3) 识别结果处理

得到识别结果后，匹配模板库，如果是特定的指令语音，则将对对应执行

相应的操作，如：返回的结果是“加入购物车”，则将扫描的商品信息加入购物车之中等等，关键代码如下：

```
if ( q == 0 ) { //成功识别
    QString str = dowith->sendForTTS( re );
    //结果处理
    if( re->r[0] == "0"){
        return ;
    }
    if( ( re->r[0].indexOf( "付款" ) ) != -1 ){
        emit payForBills();//return "付款";
    }else if( ( re->r[0].indexOf( "删除" ) ) != -1 ){
        emit deleteOneGoods();
        //QString s=re->r[0].mid(2);
        //return s;
    }else if( ( re->r[0].indexOf( "加入购物车" ) ) != -1 ){
        emit addToShoppingCar();
        //return QString("购物车已准备好");
    }else if( ( re->r[0].indexOf( "进入购物车" ) ) != -1 ){
        emit goToShoppingCar();
        //return QString("购物车已准备好");
    } else if( ( re->r[0].indexOf( "清空" ) ) != -1 ){
        emit clearShoppingCar();
        //return QString("购物车已清空");
    }else{
        emit repeatAgain();
        //return QString("请您重复一遍刚才的命令");
    }
    speech_tts* tts = new speech_tts();
    tts->TTS_main( str );
}
```

5.3 手机客户端实现

根据腾讯数据，安卓手机在 2013 年里的全球销量达 7.8 亿，占全球智能手机 78.9% 的市场份额。Android 系统同时正在向平板电脑领域发展，而且，在许多可穿戴设备中也搭载了 Android 系统。在该系统中的手机客户端以 Android 手机为

开发平台。为了促进与自助购物服务端的通信，本系统采用 Socket 通信开发 Android 客户端的应用程序^[40-44]。

5.3.1 XML 数据说明

(一) bill_layout.xml

该文档用来显示购物车里的所有商品的信息。该文档的组成：

(1) XML 文档的标准开头

```
<?xml version="1.0" encoding="utf-8"?>
```

encoding="utf-8"说明该文档是以 utf-8 编码格式。

version="1.0"说明该文档应该按照 1.0 版本的 XML 规则进行解析

(2) 该 XML 采用线性布局，为竖直方向，在最外层线性布局中又包含有两个线性布局和一个 ListView 来显示接收到的商品账单的信息。

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
```

最外层布局，整个界面的整体布局 >

```
<LinearLayout android:id="@+id/headlayout" >
```

用来加载 head.xml，在 MainActivity.java 中使用 JAVA 代码实现加载

```
</LinearLayout>
```

```
<LinearLayout>
```

添加了四个 TextView 控件，用来分别显示名称、单价、数量、价格提示栏信息

```
</LinearLayout>
```

```
<ListView>
```

用来显示来自服务端的账单信息

```
</ListView>
```

```
</LinearLayout>
```

如下图 5.6 所示。



图 5.6 bill_layout.xml

(二) head.xml

该文档用来添加了显示总价格的 TextView 和付款的 Button。在 MainActivity.java 中被加载到 bill_layout.xml 中。如图 5.7 所示。

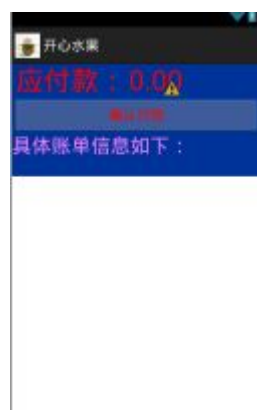


图 5.7 head.xml

5.3.2 WiFi 连接实现

因为要使用到 Wifi 连接，所以首先在 AndroidManifest.xml 配置文档里进行授权操作，添加：

```
<uses-permission android:name = "android.permission.INTERNET" />
<uses-permission android:name=
"android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission
android:name="android.permission.ACCESS_WIFI_STATE"/>
```

(1) 首先加载并显示 bill_layout.xml: setContentView(R.layout.bill_layout)。

(2) 使用 View 变量加载 head.xml:

```
View view = LayoutInflater.from( this ).inflate( R.layout.head, null )。
```

并将 head.xml 作为控件对象加载到 bill_layout.xml 中:

bill_layout 中要添加控件的对象 addView(要加载的控件在 head 中的对象)。

(3) 判断是否打开 WIFI 网络，如果没有，则提示打开如图 5.8 所示。

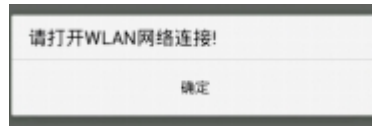


图 5.8 WIFI 网络没开提示

连接到服务端设置的 WIFI 网络中，再获取连接网络的网关：

```
wifiManager=(WifiManager)getSystemService( this.WIFI_SERVICE );
```

```
DhcpInfo dhcpInfo = wifiManager.getDhcpInfo();
```

```
host = Formatter.formatIpAddress( dhcpInfo.gateway );
```

(4) 在线程中连接到服务端，Android4.0 以后版本都需要开线程才能运行网络连接。

```
socket = new Socket( host, port );
```

5.3.3 账单获取实现

使用 SimpleAdapter 加载要显示到 ListView 中的值。

```
adapter = new SimpleAdapter( MainActivity.this,
```

需要显示的数据源为 Map 变量，

显示数据的布局文件，数据源中的 Key 值，

显示数据源中 Value 值的控件的 ID 号);

```
//listview 设置适配器
```

```
listview.setAdapter( adapter );
```

Socket 接受数据

接受数据使用：din = new DataInputStream(socket.getInputStream());再将数据以 utf 编码格式读取：content = din.readUTF() 。

Socket 发送数据

发送数据使用：dout = new DataOutputStream(socket.getOutputStream());将数据写入 dout 中，仍以 utf 的编码格式：dout.writeUTF(要发送的数据)。

获取账单的核心代码如下：

```
//建立连接
```

```
try {
```

```
    socket = new Socket( host, port );
```

```
    if ( socket.isConnected() ) {
```

```
        din = new DataInputStream( socket.getInputStream() );
```

```
        dout = new DataOutputStream( socket.getOutputStream() );
```

```
    }
```

```
} catch ( IOException e ) {  
    e.printStackTrace();  
}  
//显示接受连接消息  
try {  
    while ( true ) {  
        if ( socket.isConnected() ) {  
            if ( !socket.isInputShutdown() ) {  
                if ( ( content = din.readUTF() ) != null ) {  
                    //不能再 run 函数里对空间进行文字显示，通过 Handler 句柄来做  
                    mHandler.sendMessage( mHandler.obtainMessage() );  
                    Log.v( "Pumpkin", String.valueOf( 1 ) );  
                } else {  
                    Log.v( "Pumpkin", String.valueOf( 2 ) );  
                    Toast.makeText( MainActivity.this, "没有账单输入", 2 );  
                }  
            }  
        }  
        else Toast.makeText( MainActivity.this, "未连接", 2).show();  
    }  
} catch ( Exception e ) {  
    e.printStackTrace();  
}
```

5.3.3 支付实现

在获取账单信息后，需要付款。点击付款按钮，登录授权的银行账户，支付相应的金额，确定支付成功后，返回支付成功的消息给自助购物服务端。在接收到交易成功的特殊凭证后，可以带走购买的商品。实现后的界面如图 5.9 所示。



图 5.9 WIFI 网络没开提示

手机支付实现的核心代码如下：

```
public void onClick(View v) {
//str 为字符串变量，存储授权银行接口函数调用完成后返回的消息。
String str = 授权银行接口函数 (账号, 密码, 动态口令, 金额, 备注信息, ...);
try {
    dout = new DataOutputStream( socket.getOutputStream() );
    if ( socket.isConnected() ) {
        Toast.makeText( MainActivity.this, "网络不可用，请重试！ ", 2 );
        return;
    }
    dout.writeUTF( str );
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
}
```

5.4 系统测试

系统实现的基本功能：通过条形码扫描器扫描得出机器可读语言，匹配商品信息库，得出正在扫描的商品的信息并显示，顾客根据需要加入购物车。扫描完所有商品后进入购物车进行确认购买，将账单发送到手机上，在手机上确认付款，连接到相应的支付商家支付。

由于语音测试无法添加到论文中，就不多加赘述。在实际系统演示中，语音

部分是可以实现的。

5.4.1 测试环境

本系统硬件测试平台如图 5.10 所示。包括：路由器、WiFi 热点、智能手机客户端、数据库服务器，自助收银台（自助购物服务端）等。

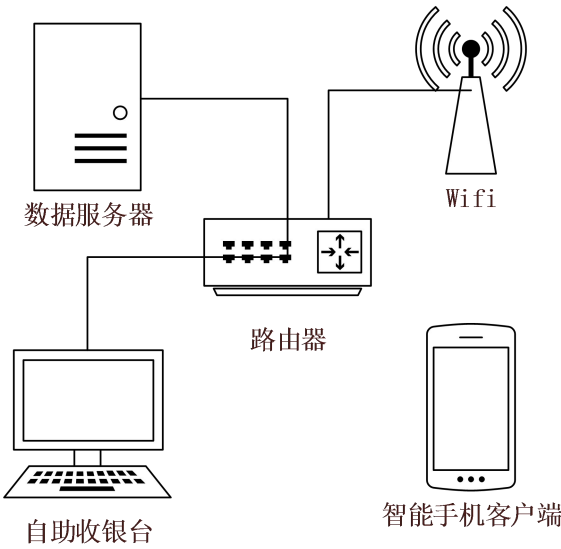


图 5.10 测试平台网络拓扑图

系统环境和配置情况如表 5.1 所示。

表 5.1 系统环境及配置表

		自助购物服务端	智能手机客户端
硬件配置	CPU	P4 3.0 GHz	Lenovo A390t 1.0GHz
	内存/ RAM 容量	2G	512M
	硬盘/ ROM 容量	250G	4G
	操作系统	Windows 7	Android 4.0.3
	网络环境	Internet 或 Intranet	支持 Internet 或 Intranet
	备注	户并发数超过 100 人时 建议使用双 CPU	Android 系统必须 2.3 以上

5.4.2 测试过程

1. 打开自助购物端和手机应用程序端。

QT 自助购物端如图 5.11 所示，此时服务端为侦听状态，如图 5.12 所示。

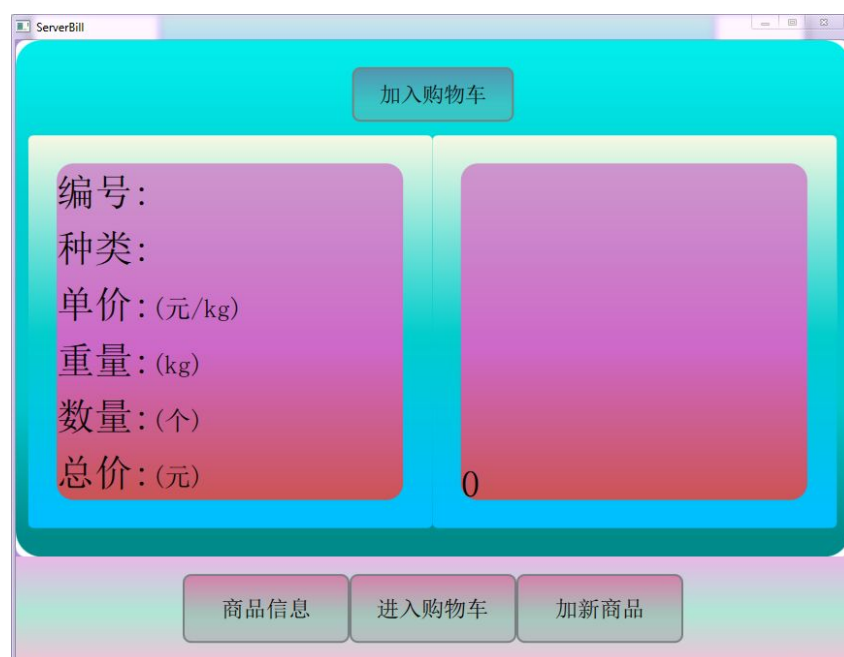


图 5.11 QT 自助购物端初始化界面

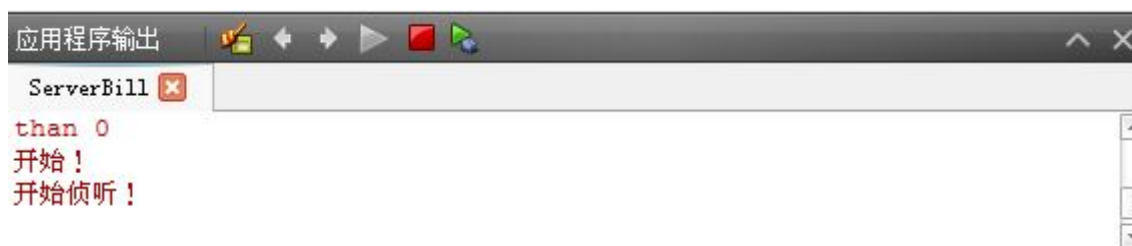


图 5.12 QT 端为侦听状态

2. 商家可通过商品信息输入界面添加新商品信息，新商品添加区如图 5.13 所示。



图 5.13 新商品添加

3. 扫描选购的商品，并加入购物车，可通过“加入购物车”语音命令实现，查看购物车，如图 5.14 所示。加入购物车的商品保存在界面的 ListView 中。当确认购买时，再从 ListView 中取出对应的信息，传递得到后台使用 C++编程处理。



图 5.14 购物车信息

4. 本系统测试使用的手机型号为 Lenovo A390t,Android4.0.3 版本的操作系统。打开手机端的应用程序，应用通过 WIFI 网络连接到自助购物端。如图 5.15 所示。



图 5.15 应用程序账单接受

5. 在 QT 自助购物端确认购买，则发送账单信息到手机应用上，手机接受后，解析账单，并显示到相应的位置。如图 5.16 所示。



图 5.16 账单信息

6. 确认付款，发送账户信息与密码，登录到相应的支付运营服务端，支付应付款项。返回支付成功的消息给自助购物端，清空自助购物端的购物车信息，等

待下一个客户。

5.4.3 测试结果

测试过程中，一切功能都正常通过。测试的功能有新商品录入功能，商品查询功能，加入购物车，清空购物车，确认购买，发送账单信息等功能。测试的功能，具体内容及其结果如表 5.2 所示。

表 5.2 测试的功能及结果

功能	测试内容	结果
新商品录入功能	录入商品的存储信息，保存进 XML 数据库	成功
商品查询功能	通过商品编号，查询商品的单价信息	成功
商品加入购物车	将扫描的商品加入购物车保存，等待与后续的商品一起结算	成功
购物车删除单个商品	删除购物车中不想购买的某个或某些商品	成功
购物车清空	清空购物车中的所有商品	成功
确认购买	单击购买按钮，将账单发送的手机上	成功
语音命令控制	通过输入加入购物车，清空或付款等命令实现相应的功能	成功
手机接收账单	手机端接账单信息	成功
手机返回支付信息	手机返回是否支付成功的信息	成功

5.5 本章小结

本章在上一章的分析和设计的基础上，对系统进行了实现和相应的测试工作。首先说明了系统的开发环境和运行环境，其次，对自助购物服务端条形码扫描、购物车、账单生成发送、语音功能进行实现和手机客户端 WiFi 连接、账单获取、支付等功能进行了实现。然后，对系统进行了测试，包括系统的测试环境，以及系统的测试过程和测试的结果。

结 论

本文详细分析了基于 Android 的语音智能收银系统的在 21 世纪现代化发展中的时代背景和重要的现实意义，以及能为社会行业带来巨大的价值，同时通过调查研究仔细了解自助购物在国内外的发展现状，明确了语音智能收银系统在当下中国的发展和结构，提出了 C/S 模式的基于 Android 的语音智能收银系统。该系统主要由两部分组成，自助购物端和手机客户端。在自助购物端，顾客通过条形码扫描器获取商品编号，由商品编号在 XML 数据库中检索商品的各项信息，确认商品的信息后，将其加入购物车中。顾客确认购物车中商品后，需要通过网络将账单信息发送到手机上，并接受来自手机端的付款是否成功的消息。自助购物端由顾客操作，交互界面友好。在自助购物端可以使用语音操作，顾客可通过输入语音命令操作购物过程，计算机再通过输出语音引导顾客进行购物过程。在手机客户端，手机通过网络连接到自助购物端，获取来自自助购物端的账单信息，并进行手机支付，返回支付是否成功的消息。购物者需开通自己银行卡的手机支付功能，商家需提前向各金融机构申请相应的服务接口。本文具体工作概括如下：

其一，阐述了系统的研究背景和现实意义。经过详细调查分析了现如今国内外语音智能收银系统的研究现状和发展趋势，并据此对系统的现实意义做出了估计。

其二，介绍了智能手机终端使用的 Android 操作系统。说明了 Android 系统的架构、平台的优势、平台的安装。介绍了 Qt 平台。说明了 Qt 平台的特点、组成模块以及平台的安装。然后详细说明了 C/S 架构，研究了语音识别的关键技术，即动态时间规整和隐马尔可夫统计模型。为后续研究、设计和实现做好铺垫。

其三，对基于 Android 的语音智能收银系统，在技术、经济和环境等方面进行可行性分析。然后从系统角色出发，从整体上分析系统的功能需求和性能需求，完整的明确了系统的真实需求，分析了系统的总体业务流程，为随后的设计打下基础。

其四，对系统的架构进行设计，包括系统的网络拓扑结构设计，软件的架构设计等。具体的分析了系统中各模块功能的划分和组织结构。然后对自助服务端和手机客户端进行流程分析及设计。

其五，对系统进行了实现和相应的用例测试。说明了系统的开发环境和运行环境，对自助购物服务端进行实现，包括条形码扫描、购物车、账单生成发送、语音功能；手机客户端的实现，包括 WiFi 连接、账单获取、支付等功能。然后，对系统进行了测试，包括系统的测试环境、系统的测试过程和测试的结果。

整体上，系统已经达到了我们设计预期功能。通过本语音智能收银系统，用户可以轻松方便的实现自助购物，同时提高了购物的效率。最大限度的增强了零售企业的市场竞争力，为社会创造巨大价值。

虽然，从整体上基于 Android 的语音智能收银系统满足了设计预期功能，但是系统还是有很多的地方需要进一步的研究和改进，以使得系统具有更强的功能和更好的使用性。如下两个方面是值得关注和研究改进的。一方面，语音识别的要求太高，要求用户说话很标准，当用户的语音带有口音时，识别难度很大，准确率达不到要求，以及语音识别的反应速度等问题。因此，语音识别算法的研究改进是很重要的，同时也是改进的难点所在。另一方面，当前的基于 Android 的语音智能收银系统只考虑一个超市的使用场景，但实际中超市可能是一个连锁超市，这个时候就需要对系统的使用范围进行扩展，从局域网扩展到广域网或者一个内部网络系统，同时保证系统内数据的完整性等。以上两方面是研究和改进的主要方面。

参考文献

- [1] Ferreira F, Dias F, Braz J, et al. Protege: A Mobile Health Application for the Elder-caregiver Monitoring Paradigm. *Procedia Technology*, 2013, 9: 1361-1371
- [2] 曹旭敏, 陈杰. 银行营业厅客户排队问题研究. *商场现代化*, 2012 (5): 59-60
- [3] 张瑞婷. 自主服务技术对顾客感知价值及文化影响研究: [西南财经大学硕士学位论文]. 成都: 西南财经大学, 2012, 15-25
- [4] 王功勋. 人机对话识别系统在移动领域的应用实践: [北京邮电大学硕士学位论文]. 北京: 北京邮电大学, 2012, 11-23
- [5] 任贇. 简单语音识别系统的设计实现. *中国新通信*, 2013, 15(16): 73-73
- [6] Xiao X, Droppo J, Acero A. Information retrieval methods for automatic speech recognition//Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on. IEEE, 2010: 5550-5553
- [7] 于淑华. 手机支付研究与探索. [北京邮电大学硕士学位论文], 北京: 北京邮电大学, 2010, 10-20
- [8] 赵新江. 手机支付, 与钱包说 Byebye. *理财: 市场版*, 2014 (3): 20-21
- [9] 陆文周. Qt5 开发及实例. 北京: 电子工业出版社, 2014, 20-30
- [10] 艾朱斯特. C++ Qt 设计模式. 第 2 版. 北京: 电子工业出版社, 2012, 40-50
- [11] 霍亚飞. Qt 应用编程系列丛书: Qt Creator 快速入门, 第 2 版. 北京: 北京航空航天大学出版社, 2014, 30-40
- [12] Marko Gargenta, Masumi Nakamura. *Learning Android: Develop Mobile Apps Using Java and Eclipse*. USA: O'Reilly Media, 2014, 1-3
- [13] 汪峙峰, 薛源. 安卓 Android 与苹果 ios 优劣与发展前景. *计算机光盘软件与应用*, 2011 (16): 34-34
- [14] 韩超, 梁泉. *Android 系统级深入开发: 移植与调试*. 北京: 电子工业出版社, 2011, 50-58
- [15] 耿祥义, 张跃平. *XML 基础教程*. 第 2 版. 北京: 清华大学出版社, 2012, 15-20
- [16] 李刚. *疯狂 XML 讲义*, 第 2 版. 北京: 电子工业出版社, 2011, 30-36
- [17] 刘雨潇, 杨雅娟. 基于 JDOM 的 XML 数据解析技术内涵研究. *现代电子技术*, 2010, 33(22): 40-41
- [18] Biskup J, Li L. On inference-proof view processing of XML documents. *Dependable and Secure Computing, IEEE Transactions on*, 2013, 10(2): 99-113
- [19] Puder A, Antebi O. Cross-compiling android applications to ios and windows

- phone 7. Mobile Networks and Applications, 2013, 18(1): 3-21
- [20] Ipsic I. Speech and language technologies. Rijeka, Croatia: InTech, 2011,3-10
- [21] Oonishi T, Iwano K, Furui S. A noise-robust speech recognition approach incorporating normalized speech/non-speech likelihood into hypothesis scores. Speech Communication, 2013, 55(2): 377-386
- [22] Lopez-Ludena V, San-Segundo R, Martin R, et al. Evaluating a speech communication system for deaf people. Latin America Transactions, IEEE (Revista IEEE America Latina), 2011, 9(4): 565-570
- [23] 江涛. 科大讯飞: 语音玩家的消费市场经. 商界: 评论, 2013 (5): 130-132
- [24] 宋伟, 金畅, 盛四辈. 我国智能语音行业专利战略研究——以科大讯飞为例. 科技进步与对策, 2011, 28(21): 107-111
- [25] 胡郁, 凌震华, 王仁华, 等. 基于声学统计建模的语音合成技术研究. 中文信息学报, 2012, 25(6): 127-136
- [26] 肖国彪. 基于 Android 平台的一体化运维管理系统的设计与实现: [东华大学学士学位论文]. 上海: 东华大学, 2014,5-10
- [27] 王云胜. 基于移动电子商务的快速消费品代理商管理系统设计与实现: [湖南大学硕士学位论文]. 长沙: 湖南大学, 2013,6-20
- [28] 明日科技. Java 从入门到精通, 第 3 版. 北京: 清华大学出版社, 2012,60-80
- [29] 梁勇, 李娜. Java 语言程序设计(基础篇). 北京: 机械工业出版社, 2011,20-100
- [30] Nokia Corporation and/or its subsidiary(-ies). Qt4.6 Whitepaper. <http://qt.nokia.com/files/pdf/qt-4.6-Whitepaper>, 2009
- [31] 李法春. Callback 机制在网络教学系统中的应用研究. 计算机与现代化, 2013 (6): 151-155
- [32] Mentis A, Katsaros P, Angelis L, et al. Quantification of interacting runtime qualities in software architectures: Insights from transaction processing in client-server architectures. Information and Software Technology, 2010, 52(12): 1331-1345
- [33] 邹琳. 基于网页的语音呼叫系统的研究与实现: [电子科技大学硕士学位论文]. 成都: 电子科技大学, 2014,11-24
- [34] 王稚慧. 基于 HMM 建模的语音识别算法的研究: [西安建筑科技大学硕士学位论文]. 西安: 西安建筑科技大学, 2005,30-40
- [35] 肖利君. 基于 DTW 模型的孤立词语音识别算法实现研究. [中南大学硕士学位论文], 长沙: 中南大学,2010,10-25
- [36] Rabiner L, Juang B H. An introduction to hidden Markov models. ASSP

- Magazine, IEEE, 1986, 3(1): 4-16
- [37] 陆华. 面向购物中心的信息系统设计与实现: [华东理工大学硕士学位论文]. 上海: 华东理工大学, 2013,15-20
- [38] Jha K R, Singh G. Terahertz planar antennas for future wireless communication: A technical review. *Infrared Physics & Technology*, 2013, 60: 71-80
- [39] Kitaoka N, Enami D, Nakagawa S. Effect of acoustic and linguistic contexts on human and machine speech recognition. *Computer Speech & Language*, 2014, 28(3): 769-787
- [40] 袁远. 基于 Android 平台端到端即时通信系统的分析与设计: [北京邮电大学硕士学位论文]. 北京: 北京邮电大学, 2012,20-30
- [41] 姚尚朗, 靳岩等. *Android 开发入门与实战*. 第 2 版. 北京: 人民邮电出版社, 2013,30-80
- [42] 纳德尔曼. *Android 应用 UI 设计模式*. 北京: 人民邮电出版社, 2013,50-80
- [43] 陈文, 郭依正等. *深入理解 Android 网络编程*. 北京: 机械工业出版社, 2013,60-100
- [44] 马志强. 基于 Android 平台即时通信系统的设计与实现: [北京交通大学博士学位论文]. 北京: 北京交通大学, 2009 ,15-30

致谢

论文即将付梓，心情却显得有些复杂五味杂陈。感慨这三年来一边工作一边攻读学位所走过的艰难曲折，艰辛付出换来的硕士论文，有些欣慰，也略有些感伤。未来的学习和成长是我人生中浓墨重彩的一部分，难忘校园里的一草一木，难忘学习中的点点滴滴，在即将结束学生生涯的此刻，略表自己的一些心声。

跟随导师向志强教授攻读硕士学位 3 年，向教授对我学习、工作各方面都给予了关怀和帮助。向教授渊博的知识、严谨的治学态度、科学的研究方法、敏锐的学术洞察力、勤奋工作的敬业精神、幽默的举止谈吐、豁达开朗的性格和高尚的人格魅力将使我终身受益。向教授一心教书育人，以学术研究为乐趣，无不让人心生敬仰。导师的境界我虽不能至，但心仍向往之。古人云“先生之风，山高水长”，这在向教授身上体现的淋漓尽致。

校外导师张华立台长是我的榜样和楷模，张台长在学术及实践经验上大公无私，将自己所掌握的知识慷慨讲解并倾囊相授，在做人做事上亦有许多闪光点。不管是在在学习上还是生活中都给予了我诸多照顾，对此我也将倍加珍惜，永记心里。

衷心感谢学院的各位任课老师期间教授我知识。各位老师对教学与科研的严谨态度对我产生了重要影响，未来在工作上这些宝贵的精神财富都将为我以后的工作学习提供了榜样。

感谢家人在我求学期间的默默付出，正是因为有他们的关心支持才有我现在的成绩。寸草春晖，永记心中。

感谢所有在我学习和成长过程中给予我指导和帮助的人们，是你们传递我知识与正能量，使我一步一步的往前走。

最后，感谢尊敬的各位评审和答辩专家的宝贵意见，让我得以进一步完善论文水平，提升论文质量。

周 昀

二〇一五年六月 于长沙