

**本科毕业设计（论文）**

**智能音箱语音控制系统的设计与实现**

**Design and Implementation of Smart** **Speaker**

**Voice Control System**

学 院： 软件学院

专 业： 软件工程

学生姓名： XXX

学 号： XXX

指导教师： XXX

北京交通大学

2017年5月

学士论文版权使用授权书

本学士论文作者完全了解北京交通大学有关保留、使用学士论文的规定。特授权北京交通大学可以将学士论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：

**中文摘要**

**摘要：**人类的语言交流是人类交互的主要方式，从键盘，鼠标，触屏控制，人机交互的本质始终没有改变，无法完成对话式的交流，而语音交互的出现打破了这一现状，它可以解放双手，甚至解放双眼，给人带来极大的快感，它也将是智能时代的主要交互方式。由于近年来深度学习的突破以及计算能力和数据积累，让语音识别得到了快速的发展，麦克风阵列技术兴起更是实现了远场自由语音交互的需求，国内的科大讯飞、声智科技，思必驰已经拥有成熟且经过市场验证的麦克风阵列技术，这种技术为智能音箱的核心技术，为智能提供了解决方案。而音响这样一个电子设备是家庭中除了手机，耳机，家电以外最常见的的设备了，仅国内它就是千亿级别的总产值，需求是刚性的。这样一款设备与语音交互结合产生的智能音箱必定拥有广阔的市场。

该项目来源于我在珠海市魅族科技有限公司北京分公司的实际项目——“gravity智能音箱”。该产品是一款基于远场语音交互的音箱。它通过语音识别，语意理解，结果处理，语音合成实现各种功能，不仅能对控制音箱音量，搜索歌曲，歌曲切换，还提供了很多日常功能，如百科，新闻，天气，时间，闹钟等。实习的工作是针对该产品的语音控制系统的设计与实现，完成语音交互与音箱实际功能的对接。

**关键词：**智能音箱、语音控制、语音交互

ABSTRACT

**ABSTRACT:** Human language communication is the main way of human interaction, from the keyboard, mouse, touch screen control, the nature of human-computer interaction has not changed, can not complete the dialogue of the exchange, and the emergence of voice interaction to break the status quo, it can liberate Hands, and even the liberation of the eyes, gives great pleasure, it will also be the main interactive era of intelligence. In recent years, due to the depth of learning breakthroughs and computing power and data accumulation, so that voice recognition has been rapid development, the rise of microphone array technology is to achieve a far-field free voice interaction needs, the domestic science and technology, sound technology, thinking Bixi already has a mature and market-proven microphone array technology, this technology for the smart speaker core technology, provides a solution for the intelligence. The sound of such an electronic device is the family in addition to mobile phones, headphones, home appliances other than the most common equipment, and only it is the domestic level of 100 billion output value, the demand is rigid. Such a combination of equipment and voice generated by the combination of smart speakers must have a broad market.

The project comes from my actual project in Zhuhai City Meizu Technology Co., Ltd. Beijing Branch - "gravity smart speaker". The product is a far-field voice-based speakers. It provides a variety of functions through voice recognition, semantic understanding, result processing, speech synthesis, not only to control the volume of speakers, search songs, song switching, but also provides a lot of daily functions, such as Wikipedia, news, weather, time, alarm clock and so on. Internship is the work of the product for the voice control system design and implementation, complete the voice interaction and the actual function of the docking dock.

**KEYWORDS**: Smart speaker, voice control，voice interaction

目 录

[**中文摘要**](#_Toc20785)

[ABSTRACT](#_Toc14904)

[目 录](#_Toc1073)

[1 引言](#_Toc19595)

[1.1论文背景与意义](#_Toc8287)

[1.2 国内外发展状态](#_Toc32553)

[1.3论文主要内容](#_Toc28970)

[1.4论文组织结构](#_Toc22648)

[2 相关技术概述与名词解释](#_Toc29002)

[2.1 相关技术概述](#_Toc26262)

[2.1.1 语音识别](#_Toc924)

[2.1.2 语音合成](#_Toc16694)

[2.1.3 自然语言处理](#_Toc26646)

[2.1.4 Android Activity](#_Toc15199)

[2.1.5 JSON数据](#_Toc29944)

[2.1.6 多线程](#_Toc19518)

[2.2 名词解释](#_Toc7912)

[3需求分析](#_Toc12136)

[3.1 系统简介](#_Toc28321)

[3.2 系统模块划分](#_Toc58)

[3.3 功能性需求分析](#_Toc14677)

[3.3.1系统用例分析](#_Toc28991)

[3.3.2 音乐类用例分析](#_Toc22781)

[3.3.3有声类用例分析](#_Toc1988)

[3.3.4天气类用例分析](#_Toc19483)

[3.3.5新闻类用例分析](#_Toc17663)

[3.3.6闹钟与提醒用例分析](#_Toc8395)

[3.3.7百科类用例分析](#_Toc12657)

[3.3.8聊天类用例分析](#_Toc27919)

[3.3 非功能性需求分析](#_Toc9332)

[4系统架构与设计](#_Toc26295)

[4.1 系统总体架构设计](#_Toc12558)

[4.2 系统接口设计](#_Toc32599)

[4.2.1 process 监听接口](#_Toc10879)

[4.2.2 UI交互接口](#_Toc19755)

[4.2.3音箱命令执行接口](#_Toc1577)

[5 模块设计与实现](#_Toc4057)

[5.1 特定场景模块](#_Toc19900)

[5.1.1 设计描述](#_Toc22633)

[5.1.2 流程图](#_Toc11220)

[5.1.3 类图](#_Toc22003)

[5.2 魅族NLP模块](#_Toc11875)

[5.2.1 新闻类](#_Toc5335)

[5.2.1.1 设计描述](#_Toc30117)

[5.2.1.2 流程图](#_Toc11373)

[5.2.1.3 类图](#_Toc31596)

[5.3 第三方NLP模块](#_Toc30617)

[5.3.1 音乐类](#_Toc15873)

[5.3.1.1 设计描述](#_Toc5804)

[5.3.1.2 流程图](#_Toc13047)

[5.3.1.3 类图](#_Toc11621)

[5.3.2 有声类](#_Toc18842)

[5.3.2.1 设计描述](#_Toc9768)

[5.3.2.2 流程图](#_Toc16874)

[5.3.2.3 类图](#_Toc19347)

[5.3.3 天气类](#_Toc1079)

[5.3.3.1 设计描述](#_Toc21536)

[5.3.3.2 流程图](#_Toc18140)

[5.3.3.3 类图](#_Toc6004)

[5.3.4 新闻类](#_Toc31412)

[5.3.4.1 设计描述](#_Toc9100)

[5.3.4.2 流程图](#_Toc31794)

[5.3.4.3 类图](#_Toc22501)

[5.3.5 闹钟与提醒类](#_Toc13897)

[5.3.5.1 设计描述](#_Toc8019)

[5.3.5.2 流程图](#_Toc3463)

[5.3.5.3 类图](#_Toc25892)

[5.3.6 百科类](#_Toc23823)

[5.3.6.1 设计描述](#_Toc2238)

[5.3.6.2 流程图](#_Toc23634)

[5.3.6.3 类图](#_Toc27362)

[5.3.7 聊天类](#_Toc15295)

[5.3.7.1 设计描述](#_Toc18988)

[5.3.7.2 流程图](#_Toc30658)

[5.3.7.3 类图](#_Toc4249)

[6总结](#_Toc7357)

[参考文献](#_Toc520)

[致谢](#_Toc19437)

[附录](#_Toc31775)

# 1 引言

本文阐述的是基于智能音箱的语音控制系统的设计是实现，其来源是我实习中的的实际项目——“gravity”智能音箱。语音控制系统是为了实现音箱功能跟语音的对接以及关于日常生活中的场景的语音交互，是该项目的核心系统之一。根据软件工程理论对本项目进行了设计与实现。本章将从论文背景与意义、国内外发展状态、论文主要内容、论文组织结构四个方面对项目进行简单介绍。

## 1.1论文背景与意义

人类的语言交流是人类交互的主要方式，从键盘，鼠标，触屏控制，人机交互的本质始终没有改变，无法完成对话式的交流，而语音交互的出现打破了这一现状，它可以解放双手，甚至解放双眼，给人带来极大的快感，它也将是智能时代的主要交互方式。由于近年来深度学习的突破以及计算能力和数据积累，让语音识别得到了快速的发展，麦克风阵列技术兴起更是实现了远场自由语音交互的需求，国内的科大讯飞、声智科技，思必驰已经拥有成熟且经过市场验证的麦克风阵列技术，这种技术为智能音箱的核心技术，为智能提供了解决方案。而音响这样一个电子设备是家庭中除了手机，耳机，家电以外最常见的的设备了，仅国内它就是千亿级别的总产值，需求是刚性的。这样一款设备与语音交互结合产生的智能音箱必定拥有广阔的市场。

该项目来源于我在珠海市魅族科技有限公司北京分公司的实际项目——“gravity智能音箱”。该产品是一款基于远场语音交互的音箱。它通过语音识别，语意理解，结果处理，语音合成实现各种功能，不仅能对控制音箱音量，搜索歌曲，歌曲切换，还提供了很多日常功能，如百科，新闻，天气，时间，闹钟等。实习的工作是针对该产品的语音控制系统的设计与实现，完成语音交互与音箱实际功能的对接。

智能音箱具有广阔的市场，可以创造巨大的财富，同时它也是未来布局智能家居的入口。语音控制系统是语音交互技术与音箱实际功能的对接，加强语音的交互体验，系统的稳定性直接影响到产品的用户体验以及生命周期的长短，它的业务流程未来甚至可以拓展到其他智能家居上。

## 1.2 国内外发展状态

2015年智能音箱出现，国外亚马逊Echo，国内有京东的叮咚音箱，阿里的小飞。智能音箱一开始就被认为是家庭互联网的入口之一，与智能家居进行融合也是其使命之一，各个巨头出于这方面的考虑都开始抢占这个领域。但是智能音箱还处于起步阶段，有很多困难还待突破。音箱特别是其中的中高端音箱，本来就强调品牌，而且是技术门槛较高的领域，而智能音箱将声学设计、无线技术、语音识别、远场拾音、自然语言处理等众多技术融合在一起，不仅使技术更为复杂，而且更加依赖音乐内容平台的支持，这些诸多因素都是创业者需要直面解决的难题。用户对智能音箱的要求也远非语音识别所能做到的那样简单，还需要深入的语音交互才行，而且这种交互还应该是日常生活中的场景。即使解决某个特定领域的语音交互就让众多科研人员心力交瘁了，更何况如此广泛的领域。

目前智能音箱的主要研究方向是好的语音交互体验，更丰富的内容交互，以及对其他智能硬件产品的控制。

## 1.3论文主要内容

结合语音识别，语音合成，自然语言处理，为了很好的实现音箱功能跟语音的对接以及关于日常生活中的场景的语音交互，语音控制系统的设计与实现是非常有必要性，同时也是非常重要的，这也是我在魅族实习的主要内容。

本论文将根据软件工程项目开发流程对语音控制系统从需求分析，架构设计，模块详细设计与实现等各个方面对系统进行论述。作者把语音控制系统根据语音识别后自然语言的处理方式分做三个模块：特定场景模块，魅族NLP处理模块，第三方NLP处理模块。然后分别从用户角度对其进行了用例分析，之后通过流程图，架构图对系统架构进行了整体性说明。然后从又设计描述，流程图，类图的角度对系统主要功能进行了详细阐述。最后对整个论文进行了总结。为了方便读者阅读本论文，以及对系统核心技术有所了解作者同时也进行了关键技术概述与名词解释。

## 1.4论文组织结构

本论文是以项目“智能音箱语音控制系统”的设计与实现作为研究对象，依据软件工程项目开发的流程进行论述，主要分为六个模块，引言、相关技术概述与名词解释、需求分析、系统架构、模块设计与实现、总结。

第一章：引言。分为四个小节，前两个小节对课题的研究背景与意义、国内外研究现状进行简述，表明了本课题的研究背景和需要解决的问题；后两个小节说明了论文主要内容和论文的组织结构。

第二章：相关技术概述与名词解释。分为关技术概述、名词解释两个小节，其中相关技术概述介绍了整个项目用到的关键技术；名词解释介绍了论文涉及到的专有名词。

第三章：需求分析。分作四个小节，第一个小节对系统作了简介，第二节是介绍系统的模块划分，把系统根据自然语言处理方式分作三大模块，第三节是对系统功能性需求进行论述，通过用例图分析了系统的主要功能，第四节是对系统的非功能性需求进行论述，从五个方面考虑了该需求。

第四章：系统架构。分作两小节，第一小节对系统整体架构进行说明，通了流程图，架构图进行了阐述。第二小节对系统主要接口进行了论述，讲解了模块与模块间是如何交互的。

第五章：模块设计与实现。分作三小节对三大模块进行了详细论述，每一个模块都从设计描述，流程图，类图进行了分析阐述。

第六章：总结，对论文撰写的总结和对所有支持、帮助我完成这个项目的人表示感谢，以及参考文献和英文文献翻译。

# 2 相关技术概述与名词解释

“gravity”智能音箱的软件端是基于Android平台实现的，语音控制系统的软件端的重要组成部分。在项目的设计实现过程中，用到的关键技术有：语音识别、语音合成、自然语言处理、Android activity、JSON数据、多线程。本章除了对相关技术进行概述，还做了名词解释。

## 2.1 相关技术概述

### 2.1.1 语音识别

语音识别技术是让机器通过识别和理解把语音信号转变为相应的文或命令，也就是让机器听懂人类的语音。目前主流的语音识别技术是基于统计模式识别的基本理论，这种语音识别系统由信号处理包括统计信号处理和[特征提取](http://baike.baidu.com/item/%E7%89%B9%E5%BE%81%E6%8F%90%E5%8F%96)、声学模型、发音字典、语言模型、解码器等几个基本[模块](http://baike.baidu.com/item/%E6%A8%A1%E5%9D%97)一起构成。信号处理和[特征提取](http://baike.baidu.com/item/%E7%89%B9%E5%BE%81%E6%8F%90%E5%8F%96)模块的主要任务是从输入信号中提取特征，供[声学模型](http://baike.baidu.com/item/%E5%A3%B0%E5%AD%A6%E6%A8%A1%E5%9E%8B)处理，同时，它一般也包括了一些信号处理技术，以尽可能降低环境噪声、信道、说话人等因素对特征造成的影响。声学模型，目前多采用基于隐马尔科夫模型的建模方法。发音字典包含系统所能处理的词汇集及其发音，实际提供了声学模型建模单元与语言模型建模单元间的映射。语言模型对系统所针对的语言进行建模，目前各种系统普遍采用的还是基于统计的N元文法及其变体。[解码器](http://baike.baidu.com/item/%E8%A7%A3%E7%A0%81%E5%99%A8)通常是在给定了根据语法、字典对马尔科夫模型进行连接后的搜索的网络后，在所有可能的搜索路径中选择一条或多条最优路径作为识别结果。

### 2.1.2 语音合成

语音合成指将文字信息转化为可听的声音信息，它涉及声学、语言学、[数字信号处理](http://baike.baidu.com/item/%E6%95%B0%E5%AD%97%E4%BF%A1%E5%8F%B7%E5%A4%84%E7%90%86)、计算机科学等多个学科技术，是[中文信息处理](http://baike.baidu.com/item/%E4%B8%AD%E6%96%87%E4%BF%A1%E6%81%AF%E5%A4%84%E7%90%86)领域的一项前沿技术。通过计算机语音合成则可以在任何时候将任意文本转换成具有高自然度的语音，从而让机器像人一样开口说话。

本项目使用的是百度语音合成技术。百度语音合成客户端Android版SDK是一种面向Android移动设备的语音合成解决方案，已支持SDK内部直接播放合成语音和从SDK获取语音数据，并支持男女声、语速、音调、音量、音频码率等设置。SpeechSynthesizer，主要用来控制合成进程：设置参数，开始，结束，取消等。getInstance()方法获取语音合成引擎实例。setContext()方法设置语音合成实例的上下文环。setSpeechSynthesizerListener()方法设置语音合成监听器，回调合成和播放进度以及错误信息。initTts()方法初始化合成引擎，可以指定使用online在线，或者mix离在线混合引擎. mix混合引擎会在online在线不能用的情况下自动使用offline离线引擎。batchSpeak()方法批量合成并播放文本文件，一次可传入一个list，这个list会按顺序合成播放。onError()方法出错时的回调函数。

### 2.1.3 自然语言处理

自然语言处理就是用计算机来处理、理解以及运用人类语言，它属于人工智能的一个分支，是一门语言学与计算机科学的交叉学科。简单自然语言处理主要是对文本进行分词、关键字提取等，复杂自然语言处理包括词义消歧、语法分析、语义理解、上下文关系等。自然语言在词法、句法及语义三个层次上充满了大量的歧义，自然语言处理的关键在于歧义消除。现在自然语言处理都要靠大规模语料库和统计机器学习方法，它涉及与语言处理相关的数据挖掘、机器学习、知识获取、知识工程、人工智能研究、语言学研究等。自然语言处理具有广泛的应用前景，特别是在信息时代，例如：机器翻译、手写体和印刷体字符识别、语音识别及文语转换、情感分类、信息检索、问答系统等。

### 2.1.4 Android Activity

Activity 是用户接口程序，原则上它会提供给用户一个交互式的接口功能。它是 android 应用程序的基本功能单元。Activity 本身是没有界面的。所以Activity 类创建了一个窗口，开发人员可以通过setContentView(View)接口把UI放到Activity 创建的窗口上，当Activity 指向全屏窗口时，也可以用其他方式实现：作为漂浮窗口，或者嵌入到其他的Activity 。Activity 是单独的，用于处理用户操作。几乎所有的Activity 都要和用户打交道，用户可以用来交互为了完成某项任务，例如拨号、拍照、发送email、看地图。创建一个新的Activity必须继承Activity类，从一个Activity跳转到另一个Activity，可以使用startActivity()方法。一个Activity 从启动到停止要经过各种状态的转变，它被通知这种状态转变通过Activity 的生命周期回调函数。有许多回调函数一个Activity 可能会收到，源于它自己的状态变化，无论系统创建它、停止它、恢复它、摧毁它，并且每个回调提供你完成适合这个状态的指定任务的条件。

### 2.1.5 JSON数据

JSON是一种轻量级的数据交换格式，采用完全独立于语言的文本格式，这些特性使JSON成为理想的数据交换语言，易于人阅读和编写，同时也易于机器解析和生成。JSON有两种表示结构，对象和数组。对象结构以“{”大括号开始，以“}”大括号结束。中间部分由0或多个以“,”分隔的“key(关键字)/value(值)”对构成，键值对之间运用 “,”（逗号）分隔。数组结构以“[”开始，“]”结束。中间由0或多个以“,”分隔的值列表组成。GSON是Google开发的Java API，用于转换Java对象和JSON对象，是本项目所使用的对象转换器。gson提供toJson() 方法可以将对象转换成JOSN字符串，想要把JSON字符串转换成Java对象可以使用其提供的fromJson()方法来实现。

### 2.1.6 多线程

线程是进程中独立运行的子任务，多线程就意味着多任务，可以在同一时间内运行多种不同的任务，极大的提高了CPU的利用率。本项目的编码语言为Java，在Java中，实现多线程编程主要的两种方式：一是继承Thread类，另一种是实现Runnable接口。多线程就涉及到线程安全与非线程安全，线程安全的情况下多线程对同一个对象实例中变量的并发访问不会产生脏读，而非线程安全就会产生脏读。对于非线程安全的情况下可以使用synchronized进行同步处理，也可以使用volatile解决变量在多个线程之间的可见性。对于并发量的控制可以使用semaphore,semaphore控制了当前的访问个数，提供同步机制，控制同时访问的个数。有时线程需要优先执行，这时就需要对线程进行优先级划分，设置线程的优先级可以使用setPriority()方法,使用sleep可以在毫秒级别内控制线程的休眠。

## 2.2 名词解释

NLP:自然语言处理--Natural Language Processing

ASR:自动语音识别--Automatic Speech Recognition

HTTP:超文本传输协议--HyperText Transfer Protocol

UI:用户界面--User Interface

# 3需求分析

语音控制系统作为智能音箱的核心组成部分，其功能的设计与实现对于音箱用户的粘性影响重大。需求分析通过对功能及系统行为建模分析，将用户需求精确化、完全化为系统的设计与实现定义了明确要求。本章将从系统简介、系统模块划分、功能性需求分析、非功能性需求分析四个方面进行论述。

## 3.1 系统简介

语音控制系统是为了控制语音交互流程，完成对音箱功能的对接以及提供用户有好的一些关于日常生活中场景的语音交互。语音控制系统根据语音识别后自然语言的处理方式分做三个模块：特定场景模块，魅族NLP处理模块，第三方NLP处理模块。特定场景模块：该模块主要处理一些音箱基本功能，如上一曲，下一曲，音量调整等和一些增加用户友好，体现音箱更加人性个功能，如闭嘴，我回来了。魅族NLP处理模块：该模块主要是通过关键字的内容识别，上下文分析，然后提供更好的服务，这里面主要是关于新闻的内容，这是魅族自己编辑、筛选过的内容。第三方NLP处理模块：该模板包含百科，诗歌，闹钟，有声娱乐等各种日常生活中的交互场景。除了以上提到的主要功能外，当然其中还贯穿着语音唤醒，语音识别，语音打断的全程开关的控制，以及日志输出，错误处理等。

## 3.2 系统模块划分

语音控制系统根据自然语言处理方式分做三个模块：特定场景模块，魅族NLP处理模块，第三方NLP处理模块。特定场景模块包括：歌曲切换、音量调节、收藏、播放暂停、回家出门。魅族NLP模块包括：新闻。第三方NLP模块包括：百科、歌曲、闹钟、天气、新闻、聊天。如图3-1所示。



图3-1 系统组织结构图

## 3.3 功能性需求分析

下面对介绍语音系统功能性需求分析，主要系统层面以及音乐类节目、有声类节目、天气、新闻、闹钟与提醒、百科类、聊天类等七大功能方面对系统进行分析。

### 3.3.1系统用例分析

下面从用户角度对语音控制系统的进行分析，使用用例图对系统功能进行展示，根据系统模块划分分为特定场景模块、魅族NLP模块、第三方NLP模块三个模块，用例分析如下所示。

首先是特定场景模块，该模块包括对歌曲切换、音量调节、收藏、播放暂停、出门回家等的语音控制，用例图如图3-2所示。

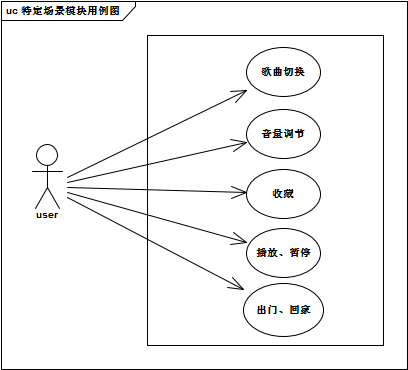


图3-2 特定场景模块用例图

接下来是魅族NLP模块，该模块是针对包含新闻文本进行NLP处理以调用魅族自己编辑过的新闻，用例图如图3-3所示。



图3-3 魅族NLP模块用例图

接下来是第三方NLP模块，该模块包括百科，歌曲、闹钟，天气，新闻，聊天类，用例图如图3-4所示。

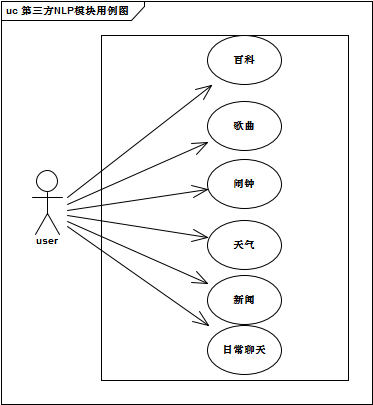


图3-4 第三方NLP模块用例图

### 3.3.2 音乐类用例分析

音乐类节目的主要功能包括：按格式点播、按歌曲点播，按组合点播、歌手名+歌曲名点播、混合点播、分类点播、歌词点播、随机点播。用例图如图3-5所示。

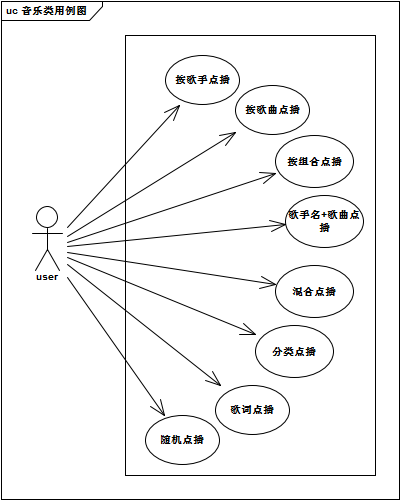


图3-5 音乐类用例图

点播格式如表3-1所示

| 功能名称 | 结果返回方式 | 例句 |
| --- | --- | --- |
| 按歌手点播 | 返回按该歌手名查询得到的结果中，热度排行前20的歌曲 | 我想听周杰伦的歌 |
| 按歌曲点播 | 返回按该歌曲名查询得到的结果中，热度排行前20的歌曲 | 我想听稻香 |
| 按组合点播 | 返回以合唱者名称综合查询得到的结果中，热度排行前20的歌曲 | 我想听周杰伦和温岚合唱的歌 |
| 歌手名+歌曲点播 | 返回按意图搜索的结果里，热度最高的1首歌 | 我想听周杰伦的稻香 |
| 混合点播 | 返回按混合条件分别搜索后，能多重命中的歌曲 | 我想听周杰伦忧伤风格的歌 |
| 分类点歌 | 返回按关键词查询到的结果中，热度排行前20的歌曲 | 我想听咖啡厅音乐 |
| 歌词点歌 | 先按歌词查询对应歌曲，再按歌曲名称查询曲库并返回排行前20的歌曲 | 歌词里的“我竟悲伤得不能自己”的歌 |
| 随机点播 | 从热门歌曲曲库中，随机返回20首歌 | 随便来首歌 |

表3-1 音乐类点播格式表

### 3.3.3有声类用例分析

有声类节目的主要功能包括：按节目类型点播、按作家点播、按节目名称点播、按作家+节目名称点播。用例图如图3-6所示。

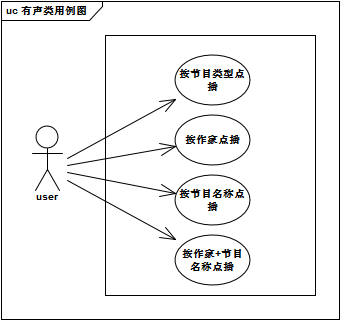


图3-6 有声类用例图

点播格式如表3-2所示

| 功能名称 | 结果返回方式 | 例句 |
| --- | --- | --- |
| 按节目类型点播 | 返回按节目类型查询得到的结果中，热度排行前20的节目 | 来段相声 |
| 按作家点播 | 返回按作家名称查询得到的结果中，热度排行前20的节目 | 我想听郭德纲的相声 |
| 按节目名称点播 | 返回按节目名称查询得到的结果中，热度排行前20的节目 | 我想听报菜名 |
| 按作家+节目名称点播 | 返回按作家名称和节目名称综合查询的结果中，能匹配的20歌节目 | 我想听郭德纲的报菜名 |

表3-2 有声类点播格式表

### 3.3.4天气类用例分析

天气类的主要功能包括：按城市查询天气、按日期查询天气、按天气查询天气、按日期+城市查询、按天气+日期查询天气、按日期+城市+天气查询天气、询问后连续查询。用例如图3-7所示。

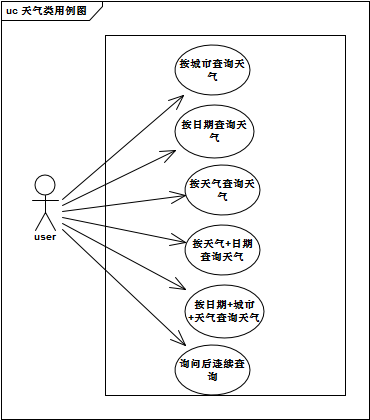


图3-7 天气类用例图

表达方法如表3-3所示

| 功能名称 | 例句 | 返回内容 |
| --- | --- | --- |
| 按城市查询天气 | 深圳天气 | 深圳明天xx℃-xx℃，x风x级，雾霾指数xx |
| 按日期查询天气 | 明天天气 | 深圳明天xx℃-xx℃，x风x级，雾霾指数xx |
| 按天气查询天气 | 下雪了吗？ | 没有 |
| 按日期+城市查询 | 明天上海天气 | 上海明天xx℃-xx℃，x风x级，雾霾指数xx |
| 按天气+日期查询天气 | 明天下雪吗 | 明天不下雪 |
| 按日期+城市+天气查询天气 | 明天深圳会下雨吗？ | 明天深圳不会下雨  明天深圳会下雨 |
| 询问后连续查询 | 那北京呢？ | 按上一条查询方式 |

表3-3 天气类表达方式表

### 3.3.5新闻类用例分析

新闻类的主要功能包括：按新闻类查询、按事物查询、按时间查询。用例图如图3-8所示。

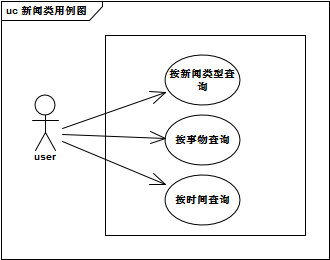


图3-8 新闻类用例图

表达方法如表3-4所示

| 功能名称 | 结果返回方式 | 例句 |
| --- | --- | --- |
| 按新闻类型查询 | 播报该类型新闻 | 我想听国内新闻 |
| 按事物查询 | 播报根据该事物查询到的新闻 | 王宝强最近有什么新闻 |
| 按时间查询 | 播报两日内热门新闻 | 最近有什么新鲜事儿？ |

表3-4 新闻类表达方式表

### 3.3.6闹钟与提醒用例分析

闹钟与提醒的主要功能包括：按日期设定闹钟、按时间设定闹钟、按循环方式定闹钟、删除已设闹钟、更改已设定闹钟参数、设定提醒、更改提醒、删除已设定提醒。用例图如图3-9所示。

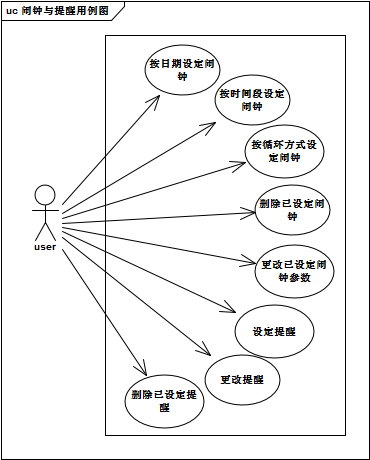


图3-9 闹钟与提醒用例图

表达方式如表3-5所示

| 功能名称 | 例句 | 应返回的参数 | 本项需处理的不同行为 |
| --- | --- | --- | --- |
| 按日期设定闹钟 | 定明天早上8点的闹钟 | 新建、明天、早上8点 |  |
| 按时间段设定闹钟 | 定半小时后的闹钟 | 新建、今天、半小时后的时间点 |  |
| 按循环方式定闹钟 | 定工作日早上8点的闹钟 | 新建、工作日、早上8点 | 工作日、每天、每周X（每周X可有多个） |
| 删除已设定闹钟 | 取消工作日早上8点的闹钟  删除我的所有闹钟 | 删除、工作日、早上8点  删除、所有 | 删除 |
| 更改已设定闹钟参数 | 把工作日早上8点的闹钟后延半小时 | 更改、工作日早上8点的闹钟、延后半小时的时间点 | 延后、提前 |
| 设定提醒 | 十分钟后提醒我洗澡 | 十分钟、洗澡 | 时间、行为 |
| 更改提醒 | 把提醒后延10分钟 | 后延、十分钟 | 时间、修改行为 |
| 删除已设定提醒 | 不需要提醒我去洗澡了 | 删除、洗澡、提醒 |  |

表3-5 闹钟与提醒表达方式表

### 3.3.7百科类用例分析

百科类为按名称查询返回的结果，用例图如图3-10所示。

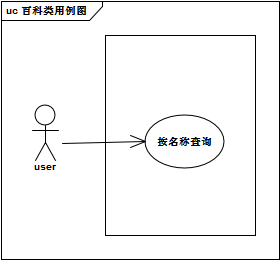


图3-10 百科类用例图

表达方式如表3-6所示

| 功能名称 | 结果返回方式 | 例句 |
| --- | --- | --- |
| 按名称查询 | 返回按关键词查询到的百度百科结果 | 谁是王菲 |

表3-6 百科类表达方式表

### 3.3.8聊天类用例分析

聊天类为随意说一句话，当所有分类都处理不了的时候，返回聊天结果。用例图如图3-11所示。

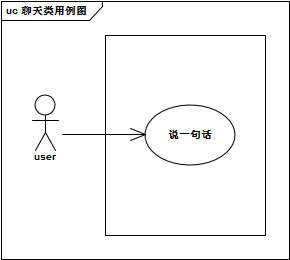


图3-11 聊天类用例图

## 3.3 非功能性需求分析

系统除了考虑功能性需求外，还应该考虑非功能性需求。当非功能性需求被很好的满足，系统的功能将更能够被用户接受，同时也将更有利于系统未来的升级优化、维护，这往往能觉得整个项目的成败。下面从五个方面考虑非功能性需求。

系统实用性:系统除了有针对音箱控制的功能外，还有可以对日常工作可以进行辅助的功能。比如通过语音设置闹钟，询问天气，听新闻，百科等。同时还可以进行日常语音交互。在进行广泛的语音交互的过程中，语音控制系统要能够控制每一次语音交互的完整性，以及能够打断语音交互，和命令式退出语音交互。

系统可靠性:在语音控制系统运行过程中不应该发生系统故障，在语音交互过程中造成系统崩溃时，应能够进行系统恢复，并且不影响其他语音交互功能。

系统效率:在进行语音交互的时候，状态切换都是控制在秒的级别。唤醒后三秒内无输入给出提示，五秒内无输入再次给出提示，如果十秒内一直没有语音进行交互则退出语音交互；当一次语音交互结束后，如果五秒内没有再次进行语音交互则退出语音交互。

系统可拓展性:当语音识别，自然语言处理的技术进一步提升后，语音交互将能够实现更多的功能，为了接入更多的语音交互功能，系统的可拓展性是必须的。同时伴随着智能家居，物联网的进一步发展，在以后针对家居与家居的相连，公司进行功能调整的时候，都需要系统有良好的可拓展性。

用户友好性:语音交互是人与机器的交互，相比于人与人进行语言交流，机器如果不能及时反馈，用户体验将会非常不好。所以当系统出现错误，或者无法识别问题以进行语音交互的时候，系统都设置了特殊情况发生下的特定语音，以及特定交互UI。同时根据测试结果和用户体验设置体验良好的语音交互流程。

# 4系统架构与设计

在上一章的基础上，读者已经对系统的功能比较了解了。本章将结合所有功能对语音控制系统进行系统架构与设计，下面从系统总体架构设计、系统接口设计两个方面进行论述。系统总体架构设计主要分析构成系统的模块及其关系，系统接口设计主要介绍模块与模块间是如何交互的。

## 4.1 系统总体架构设计

系统的整体流程是先由用户录入语音，然后由思必驰进行语音处理，把语音转文字并做初步理解，识别出里面的部分业务，如：语音无法理解，识别错误等。然后根据处理完后的文本，拦截特定场景下的业务，处理后让音箱去执行相关命令。如果超出这个范围，这把文本交个NLP服务方，这么NLP服务方进行处理分为魅族NLP服务方和第三方NLP服务方，然后识别出其中的音乐，百科，有声，天气，聊天机器人，及其他业务，处理后让音箱去执行相关命令，至此完成一轮语音对话，如果用户要进行持续语音对话，就再次进行语音输入，完成第二轮对话，这样进行多轮循环。流程图如图4-1所示。



图4-1 系统流程图

系统的实现由语音输入模块（唤醒、语音）、文本处理模块（特定场景模块、魅族NLP模块、第三方NLP模块）、业务逻辑处理模块（process）、交互处理模块（Android UI，语音合成）、命令下发模块五个模块组成。用户先进行语音唤醒，成功后音箱进入语音识别状态，当用户输入语音后进行语音识别，把识别后的文字进行文本处理，文本处理分作三块：特定场景模块、魅族NLP模块、第三方NLP模块。由这三块依次处理。处理完后把文本下发给process进行业务处理，处理完后再根据业务显示Android UI交互页面，同时给出提示音，或者语音合成播报动态内容。然后在根据具体业务把命令下发让音箱执行相应功能。具体架构设计如图4-2所示。

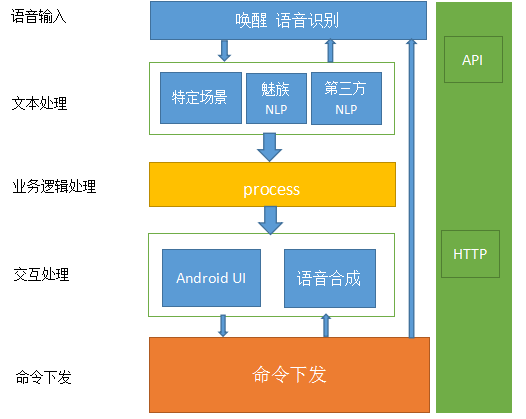


图4-2 系统架构图

语音输入模块：该模块唤醒，语音识别两个子模块。唤醒指通过个某个特定的关键字让音箱从录音状态切换到语音识别状态。语音识别指在用户录入语音后，把语音转文本，然后把得到的结果JSON数据和状态码（0：成功；70904：等待语音录入超时；70914：识别错误；70910：网络超时）交给下个模块处理，同时在处理过程中关闭语音唤醒和语音识别。

文本处理模块：该模块从上一个模块传递过来的数据中提取出用户输入input部分数据，依次让特定场景，魅族NLP，第三方NLP进行处理，只要有一个模块能够处理就停止传递，并返回处理后的状态值。

业务逻辑处理模块：对上一个模块传递下来的实际业务进行处理，如音乐，要处理资源无效、资源有效、播完完成、播放打断等业务逻辑。在业务逻辑处理的同时，交互处理模块，命令下发模块也开始执行。

交互处理模块：根据不同的业务传递过来的类型值，显示不同的UI交互及语言提示。

命令下发模块：更加业务逻辑模块处理后的具体业务下发相应的命令，根据不同情况，在命令下发后可能调用交互处理模块，如收藏，音量调节，会在命令操作成功后在进行交互处理；在命令下发后可能是对唤醒，语音识别的开启、关闭，如音乐播放，会把语音识别关闭，语音唤醒开启。

## 4.2 系统接口设计

这里主要介绍控制process 监听接口、UI交互和音箱命令执行的接口，process监听接口是针对process的处理状态控制唤醒，以及语音识别的开关，控制UI交互的接口是针对用户语音交互过程中UI状态以及提示音的切换，音箱命令执行的接口是针对音箱功能的控制。

### 4.2.1 process 监听接口

接口设计如图4-2所示



图4-3 process 接口设计图

方法参数的值如表4-1所示

|  |  |
| --- | --- |
| Code 值 | 描述 |
| -1 | 关闭交互界面，语音识别 |
| 0 | 关闭语音唤醒，语音识别继续 |
| 2 | 结束语音控制 |
| 3 | 切换音箱状态到工作者状态 |
| 4 | 切换音箱状态播放新闻 |

表4-1 process接口方法参数表

### 4.2.2 UI交互接口

接口设计如图4-2所示



图4-2 UI交互接口设计图

方法参数值如表4-2所示

|  |  |
| --- | --- |
| Param值 | 描述 |
| ResponseStatus.RESPONSE\_STATUS\_WAKE\_UP | 唤醒 |
| ResponseStatus.RESPONSE\_STATUS\_WAKE\_UP\_AGAIN | 再次唤醒 |
| ResponseStatus.RESPONSE\_STATUS\_END\_COMMUNICATE | 结束对话 |
| ResponseStatus.RESPONSE\_STATUS\_WAKEUP\_CANNOT\_UNDERSTAND | 无法理解 |
| ResponseStatus.RESPONSE\_STATUS\_ERROR\_HANDLING | 错误处理 |
| ResponseStatus.RESPONSE\_STATUS\_PLAYING | 播放 |
| ResponseStatus.RESPONSE\_STATUS\_HUMAN\_SERVE | 人口客服 |
| ResponseStatus.RESPONSE\_STATUS\_TIMING | 提醒 |
| ResponseStatus.RESPONSE\_STATUS\_ARM\_CLOCK | 闹钟 |
| ResponseStatus.RESPONSE\_STATUS\_COMMUNICATE\_END\_WAITING\_INPUT | 对话结束等待再次输入 |
| ResponseStatus.RESPONSE\_STATUS\_BREAK, | 打断 |
| ResponseStatus.RESPONSE\_STATUS\_ASR\_INPUT | 识别结果 |
| ResponseStatus.RESPONSE\_STATUS\_TEST, | 测试 |
| ResponseStatus.RESPONSE\_STATUS\_COLLECTION | 收藏 |
| ResponseStatus.RESPONSE\_STATUS\_SHUTDOWN | 关闭 |
| ResponseStatus.RESPONSE\_STATUS\_NETWORK\_TIMEOUT | 网络超时 |
| ResponseStatus.RESPONSE\_STATUS\_CENTER\_CONTROL\_UNPROCESSED | 中控 |
| ResponseStatus.RESPONSE\_STATUS\_CAN\_NOT\_HANDLE\_COMMAND | 命令无法处理 |
| ResponseStatus.RESPONSE\_STATUS\_NETWORD\_USELESS | 网络不可用 |

表4-2 UI交互接口方法参数表

### 4.2.3音箱命令执行接口

接口设计如图4-3所示



图4-3 音箱命令执行接口设计图

音箱控制部分协议如表4-3所示：

|  |  |  |  |
| --- | --- | --- | --- |
| action | 说明 | 请求方式 | 携带的参数 |
| Info | 获取音箱的设备信息 | Get | 无 |
| Status | 获取音箱的播放状态 | Get | 无 |
| SetPlayMode | 设置播放模式（乱序，随机，顺序） | Post | {  playMode:Int  } |
| Pause | 请求音箱暂停播放 | Get | 无 |
| Resume | 请求音箱继续播放 | Get | 无 |
| Prev | 请求音箱播放下一首歌曲 | Get | 无 |
| SetVolume | 设置音箱音量 | Post | {  currentVolume:Int  } |
| GetVolume | 获取音箱音量 | Get | 无 |
| Seek | 跳转到当前播放的曲目的某个时刻 | Post | {  position:Int  } |
| SetAlarm | 设置闹钟 | Post | {  "alarmList":  [  {  "time":String "mode":Int "dayList":[int] "playList":PlayList  "volume":Int "enable":Boolean  },  ]  } |
| GetAlarm | 获取音箱端所有闹钟列表 | Get | 无 |
| DeleteCurrentSong | 取消歌曲收藏 | Post | {  track:Track  } |
| Suspend | 定时关机 | Post | {  time:int  } |

表4-3 音箱控制协议表

# 5 模块设计与实现

本章对每个模块的详细设计进行分析，每一个详细设计都由设计描述、流程图、类图三部分组成，从不同的角度对详细设计进行分析。设计描述，通过功能分析简单描述该实现方式；流程图，从用户角度分析功能的实现流程；类图，从类与类之间的调用关系及接口设计方面分析具体实现方式。

## 5.1 特定场景模块

### 5.1.1 设计描述

特定场景模块主要用于对音箱功能的控制，以及增加用户友好性，该模块包括：歌曲切换、音量调节、收藏、播放暂停、回家出门等技能。特定场景在关键字在系统启动的时候先加载进来，当用户说一句话，语音识别后，对转换成的文本后进行进行关键字过滤。在有的场景下文本就等于关键字时触发特定场景，如暂停关键字：静音、暂停、停止播放、暂停播放、闭嘴，结束对话关键字：再见、拜拜、没事儿了、没有了、没了、你退下吧。在有的场景下包含特定场景关键字时触发特定场景，如频道切换关键字：切、换、播放，同时还必须包含“频道”字符。当特定场景触发后则把显示相应的交互UI和提示音，并且把命令下发给音箱，执行相应的音箱功能。关键字设置如表5-1所示。

|  |  |
| --- | --- |
| 类型 | 内容 |
| Pause | 静音，暂停，停止播放，暂停播放，闭嘴 |
| Scenarios\_LeaveHome | 上班去了，上班去啦，上班了，出门了，出门啦，我上班去了，我上班去啦，我上班了，我上班啦，我出去了，我出去啦 |
| Scenarios\_GetHome | 回家了，回家啦，到家了，回来了，我回家了，我回家啦，我到家了，我到家啦，我回来了，我下班啦，下班了，下班啦 |
| Music\_Play | 我要听，我想听，播放，给我播放，为我播放，搜索 |
| Close | 关上，关闭，关机，休眠 |
| Next | 下一首，下一集，下一回，下首歌，切歌，换首歌听，再来一首，换首歌，换一个，换一首 |
| Prev | 上一首，上一曲，上一个，上一集，上一回 |
| Next\_Ch | 下个频道，下一频道，下一个频道 |
| Prev\_Ch | 上一频道，上个频道，上一个频道 |
| Volume\_Up | 声音大点，大点声，大声点，声音大一点，大声一点 |
| Volume\_Down | 声音小点，声音小一点，小一点，小声点，小声一点 |
| End\_Dialog | 再见，拜拜，没事儿了，没有了，没了，你退下吧 |
| Collection\_Song | 收藏，收藏这首歌，我喜欢这首歌，我收藏这首歌，喜欢这首歌 |
| Collection\_Song\_Cancel | 取消收藏，取消收藏这首歌，不收藏这首歌了 |
| Collection\_Album | 收藏专辑，收藏这个专辑，帮我收藏这个专辑 |
| Collection\_Album\_Cancel | 取消收藏专辑，取消收藏这个专辑，不收藏这个专辑 |

表5-1 特定场景关键字表

### 5.1.2 流程图



图5-1 特定场景流程图

### 5.1.3 类图



图5-2 特定场景类图

SpeechWakeupEngineOnWakeupListenerImpl类：监听语音识别，并取得结果，然后对结果进行前期处理，以找出当前用户说的一句话属于哪个模块。

KeyWordsFetcher类：用来加载特定场景的关键字

KeyWords类：关键字抽象类，这个类定义了判断用户语音转文本后是否等于特定场景关键字的equalsKeyWord()方法，以及是否包含特定场景关键字的containsKeyWord()方法。该类定义的抽象方法doHandle()用于具体的特定场景继承KeyWords类后自己去实现关于特定场景的一些业务，如命令下发、显示UI交互等。

ActionManager类：用于控制一些action使用，如语音合成播放文本、语音识别的开启关闭、语音唤醒的开启关闭、UI交互等。

ResponseState接口：状态响应接口，该接口定义了状态切换switchState()方法。具体的业务由实现该接口的状态去实现。

ResponseActivity类：用于显示UI的Android activity类。

ResponseAudioPlayer类：用于播放提示音。

## 5.2 魅族NLP模块

### 5.2.1 新闻类

#### 5.2.1.1 设计描述

用户输入的语音转文本后，判断是否属于特定场景，如果不属于再判断是否包含“新闻”关键字，如果是则对该条文本进行魅族自己定义的NLP处理，否则交由其他NLP处理。魅族NLP处理主要是通过模板方法去掉新闻查询表达方式里的动词，如：听、播、来一段等，同时去掉结尾的“的新闻|地新闻”字符，然后在分组提取出里面的关键字，得到NLP结果，NLP结果包含频道、关键字、收索关键字以及原问题，然后根据结果去查询魅族编辑过的新闻内容。新闻内容包括：军事，财经，体育，娱乐，八卦，科技，国外，国内，社会，热点，热门，魅族新闻。如果查询到新闻，把查询的所有新闻下发给音箱进行语音播报并显示交互UI。

#### 5.2.1.2 流程图



图5-3 魅族NLP——新闻类流程图

#### 5.2.1.3 类图



图5-4 魅族NLP——新闻类类图

SpeechAsrEngineResultListenerImpl类：其中的onResult()方法用来处理语音识别后的结果，distributeResult()方法用于分发结果。processCloudResult()方法获取相应模块的处理结果。

MZNLPQuery类：用于处理魅族NLP模块。

NLPQuery类：用于获取NLP处理结果，结果以MZResult对象的形式返回。

NewsAnnouncer类：用于处理NLP结果，把每个News对象中的内容，标题，URL提取出来，放进Content中，然后把list<data>下发给音箱进行语音播报，并改变UI状态。

SpeechManager类：用于控制UI状态个改变，如关闭、启动、维持不变。

## 5.3 第三方NLP模块

### 5.3.1 音乐类

#### 5.3.1.1 设计描述

NLP处理方根据句义识别用户是否有点播意图，意图是否可直接执行：如果用户录入的内容，视为命中了定义的音乐类点播格式，且程序在检索后发现有结果，则视为用户的这条意图可以直接执行；如果用户录入的内容，不能够准确地命中音乐类点播格式，或命中了点播格式，但搜索结果无可用内容，则需要做下一层判断。

若意图可直接执行，则由NLP服务方返回相应结果到魅族，魅族音箱播放器按结果中的歌曲ID拉取播放地址信息进行播放。

若意图不能执行，则对用户意图做分词，拆分出能够命中相关音乐类词库的关键词，有相应关键词，则以对话方式引导用户接受推荐，用户返回肯定意图，则返回推荐结果到音箱；用户不接受推荐，则结束对话，由音箱询问用户是否有新要求。

关键词：关键词指的是对于音乐类需求典型关键词的词库，如：歌手名词库、歌曲名词库、风格名称词库等。

#### 5.3.1.2 流程图



图5-5 音乐类流程图

#### 5.3.1.3 类图



图5-6 音乐类类图

RuyiAi类：用于获取第三方NLP结果，sendMessage()方法用于把用户语音识别后的文本发送给第三方，然后把得到JSON字符串序列化为RuyiIntentsResult对象。

HttpHelper类：通过HTTP获取资源。其中sendGet()方法用于get方式的请求，sendPost()方法用于post方式的请求。

RuyiIntentsResult类：第三方NLP返回的JSON结果的序列化对象。分作四层，从RuyiIntentsResult到Result到Intent到Output。

RuyiMusicResult类：用于封装从RuyiIntentsResult类的对象中提取出关于音乐的信息。

ProcessListener接口：用于当第NLP结果处理完后回调对语音唤醒、语音识别的控制。

### 5.3.2 有声类

#### 5.3.2.1 设计描述

NLP处理方根据句义识别用户是否有点播意图。意图是否可直接执行：如果用户录入的内容，视为命中了定义的有声类点播格式，且程序在检索后发现有结果，则视为用户的这条意图可以直接执行；如果用户录入的内容，不能够准确地命中有声类点播格式，或命中了点播格式，但搜索结果无可用内容，则需要做下一层判断

若意图可直接执行：则由NLP服务方返回相应结果到魅族，魅族音箱播放器按结果中的有声节目ID拉取播放地址信息进行播放。

若意图不能执行，则对用户意图做分词，拆分出能够命中相关有声类词库的关键词，有相应关键词，则以对话方式引导用户接受推荐，用户返回肯定意图，则返回推荐结果到音箱；用户不接受推荐，则结束对话，由音箱询问用户是否有新要求。

关键词：关键词指的是对于有声类需求典型关键词的词库，如：声优名词库、有声节目名词库、风格名称词库等

#### 5.3.2.2 流程图



图5-7有声类流程图

#### 5.3.2.3 类图



图5-8 有声类类图

RuyiAudioResult类：用于封装从RuyiIntentsResult类的对象中提取出关于有声的信息。

PlayInfo类：用于封装对RuyiAudioResult类对象内容处理后的数据的封装。

RuyiAi类：该类中processAudio()方法用于处理有声类业务，如命令下发、UI交互，资源不可用404处理等。

### 5.3.3 天气类

#### 5.3.3.1 设计描述

默认的表达方式可以理解为日期+城市+天气，如果用户的表达里没有城市、日期，则默认为今天、当前城市。

天气的表达里，“天气”视为全面的天气预报，“晴风雨雪雾霾”等视为天气特征的表达。对于问句里问详细气象的意图，只返回天气结果里那一段的结果。举例如：

明天下雪吗？—>明天不下雪

明天雾霾—>明天北京雾霾指数为219，建议您务必戴上口罩

明天天气—>明天北京天气晴，北风2-3级，气温-3—8℃，雾霾指数为23，请注意保暖

特别要求:

当用户按天气查询天气时，若查询的天气可能引起气温变化，则在得到否定答复时，只回答是与否；得到肯定答复时，回答是与否，并同时返回查询的天气、气温的报告，以及人性化提示。举例如下：

Q：北京明天下雪吗？

A：不会下雪的

A：会下雪哦！北京明天小雪转大雪，-3℃~ -10℃，建议您多穿衣服，尽量少出门

返回结果的要求:

对于天气的返回结果，要求尽量人性化，做一些温暖的提示，举例如下：

Q：北京明天下雪吗？

A：会下雪哦！北京明天小雨转大雪，-3℃~ -10℃，建议您多穿衣服，尽量少出门

Q：北京明天多少度？

A：北京明天14摄氏度，微风，有轻微雾霾，建议您穿上外套，戴上口罩

对于返回结果的文案，日期描述的结果，按用户说法返回。举例说明：

问：北京明天下雨吗？

答：北京明天不下雨

问：12月21号（日）上海下雨吗？

答：12月21号（日）上海会下（大）雨，请记得带伞

#### 5.3.3.2 流程图



图5-9 天气类流程图

#### 5.3.3.3 类图



图5-10 天气类类图

RuyiAi类：把第三方NLP返回的结果提取、处理后封装成CompoundData对象，然后下发给ActionManager类。

CompoundData类：包含用于动态语音合成的数据compoundStr和回调类ProcessListener。

ActionManager类：用于调用语音合成引擎及交互UI。

ActionHandler类：用于辅助ActionManager类异步调用语音合成引擎。

TTSEngine接口：语音合成接口，定义了开始合成startCompound()方法，合成compound方法，是否在合成isCompouding()方法，停止合成stopCompoud()方法，合成完成completeCompound()方法以及回调接口TTSCompoundListener、TTSInitListener，具体是逻辑由实现类自行实现。

BaiduTTSEngine类：具体的语音动态合成引擎，实现TTSEngine接口并提供回调方式。

### 5.3.4 新闻类

#### 5.3.4.1 设计描述

固定类型的新闻，如科技新闻、国内新闻、娱乐新闻等，需要返回【人声阅读】的音频文件，发送给音箱直接可以播放，以保证视听感受。

按事物查询的新闻，需要走网络查询的接口，查找到对应的新闻，并需要保持新闻长度在200字以内。对于源数据少于200字的，直接播放；对于源数据多于200字的，需要截取前200字，若抓取的新闻源有概要类信息，则直接播放概要。

按时间查询的新闻，需要返回【人声阅读】的新闻里，2日内的热门新闻。

#### 5.3.4.2 流程图



图5-11 第三方NLP——新闻类流程图

#### 5.3.4.3 类图



图5-12 第三方NLP——新闻类类图

RuyiAi类：该类的processIntentsResult()方法对第三方NLP返回结果进行前期处理，判断是有声类新闻还是文本新闻，如果是有声类新闻则按有声类处理，否则按文本新闻处理。processNews()方法用来处理文本新闻，从第三方NLP返回的JSON数据中提出标题、描述，然后传递给NewsAnnouncer类处理，播放新闻和显示UI交互。

Content类：用来封装从第三方NLP结果中提取处理的数据。

ProcessListener接口：用来在新闻处理完回调对语音唤醒，语音识别的控制。

### 5.3.5 闹钟与提醒类

#### 5.3.5.1 设计描述

按照用户不同的闹钟需求，返回不同的参数，由音箱自行在闹钟上做不同行为。

主要有新建、删除、更改三类功能。NLP服务方需要覆盖用户不同意图的说法，比如新建闹钟的说法有“定、设定、设置、添加、设”。

未确定循环方式的闹钟，默认循环方式为只响1次

允许用户更改已设定闹钟参数，包含循环方式、时间点、日期。

在定义产品时，魅族倾向于认为闹钟功能是【有规律地面向未来做日程规划】，即认为闹钟功能一般是未长期的未来服务，并且有一定循环规律；魅族倾向于认为提醒功能是【即时性地面向短期未来做规划】，即认为提醒功能面向的是短暂的未来。虽然不排除用户会有制定长期可循环提醒、短期闹钟的case，但多数情况下只为核心定义考虑。

闹钟与提醒，不需要用户复核任务是否完成。在音箱产品上体现时，当用户打断了闹钟或提醒事件，就视为服务已经到位——用户已经感知到音箱的提醒了。因此，对于闹钟和提醒功能的管理只需要魅族记录、更新状态，不需要向云端汇报。

按照用户输入的提醒内容，拆分出时间点、用户意图，返回时间参数、具体时间、答复用户的文字。

#### 5.3.5.2 流程图



图 5-13 闹钟与提醒类流程图

#### 5.3.5.3 类图



图5-14 闹钟与提醒类类图

RuyiAi类：其中processTime（）方法对第三方NLP结果进行前期处理，然后传递给TimeResultProcessor类进行处理。

TimeResultProcessor类：对传递进来的数据进行进一步处理然，如果属于闹钟则传递给ClockProcessor类进行处理，如果属于提醒类则传递给RemindProcessor类进行处理。

ClockProcessor类：对闹钟进行处理，提取出闹钟的具体时间，同时把所有已设置的闹钟从后台取出来进行操作，完成后设置闹钟并显示UI交互和提示音。

RemindProcessor类：对提醒进行处理，提取从提醒的具体时间然后下发给音箱，同时显示交互的UI和提示音。

### 5.3.6 百科类

#### 5.3.6.1 设计描述

用户查询百科，如：故宫，王菲。语音识别正确后，第三方NLP进行处理，判断是否收到百科结果，收到则进行动态语音合成播报并显示UI交互，否则进行百度百科查询，如果查询到内容，动态进行语音合并播报显示UI交互。如果查询不到内容，给出提示音，关闭语音唤醒，语音识别继续。

#### 5.3.6.2 流程图



图5-15 百科类流程图

#### 5.3.6.3 类图



图5-16 百科类类图

RuyiAi类：其中processCommon()方法为根据第三方NLP结果进行处理，如果处理不了，则通过getBaikeAnswer()方法进行百科查询。

BaikeDataCrawler类：用于通过HTTP查询百科内容，并在查询完后告诉监听器处理结果。

BaikeDataCrawlerListener接口：用于在BaikeDataCrawler类处理完后回调结果处理逻辑。如动态语音合成播报结果、显示交互UI。

### 5.3.7 聊天类

#### 5.3.7.1 设计描述

用户进行语音交互，判断是否为聊天类型。如果是，则根据第三方返回内容，提取机器聊天结果进行语音动态合成并显示UI交互，交互完成后回调控制语音识别的开启与关闭。否则交由其他类型进行处理。

#### 5.3.7.2 流程图



图5-17 聊天类流程图

#### 5.3.7.3 类图



5-18 聊天类类图

RuyiAi类：根据第三方NLP处理结果，如果用户语音识别后的文本类型属于聊天（chat\_kids）类型。则从第三方NLP返回的JSON数据中提取出语音合成文本。

CompoundData类：用于封装根据用户语音机器返回的聊天结果。

ActionManager类：对聊天结果调用语音合成引擎动态合成，并在合成结束后调用结果处理逻辑。

# 6总结

本项目来源于本人在魅族科技有限公司北京分公司的实际项目——“gravity智能音箱”。我在实习过程中主要负责语音控制系统的设计与实现。通过背景分析发现智能音箱很有发展前景，也是未来智能家居的入口，在音箱上的进行语音交互是智能音箱的标志性特色，也是核心竞争力。语音控制系统的设计与实现就是这样的需求下诞生的，旨在提供良好的语音交互效果，完成通过语音交互与音箱功能的对接及日常对话。

本文从背景分析，关键技术概述与名词解释，需求分析，系统架构，模块设计与实现等五个方面对该系统进行阐释。

关于背景分析，主要论述了项目的背景及意义、国内外发展状况、论文主要内容。通过背景分析读者能了解到智能音箱的发展前景以及对论文有一个整体性的了解，便于理解论文的内容。

关于关键技术概述与名词解释，简单介绍项目涉及到的关键技术与专业名词。关键技术：语音识别，语音合成，自然语言处理，Android activity，JSON数据，多线程与回调。专业名词：NLP，ASR。通过对关键技的论述与名词解释让读者了解项目的难度以及方便阅读文论。

关于需求分析，对项目做了简介，模块划分，以及主要功能的用例图分析。通过需求分析，读者能了解到语音控制系统的主要功能以及系统模块间的组织关系，从而对项目有个全面的了解。

关于系统架构，说明了系统的整体架构及主要的系统接口。通过架构分析，整体性介绍了系统工作流程，模块的作用以及模块与模块之间的连接关系。通过系统接口论述，让读者理解模块与模块之间具体是如何对接以及相互协作的，语音控制系统是如何设计的。

关于模块设计与实现，通过设计描述，流程图，类图对模块设计与实现进行了详细论述，让读者对系统的实现细节有了更好的理解，了解到语音控制系统是如何实现的。

通过项目开发与论文撰写，我学到了很多东西同时也发现自己还有很多不足需要改进提升。项目开发过程中涉及到了一些前沿技术，比如语音识别，自然语言处理等，极大的开阔了我的眼界，也让我对技术更加热爱与崇拜。开发过程中少不了与其他同事沟通，协作开发，让我明白了及时沟通的重要性，还有好的沟通技巧能很好的提高沟通效率。我也需要不停的学习进步，一点点慢慢的改正自己不足之处。

# 参考文献

【1】 《疯狂Android讲义》 李刚 电子工业出版社 2011-6

【2】 《Java多线程编程核心技术》 高洪岩 机械工业出版社 2015-6

【3】 《Head First 设计模式（中文版）》 弗里曼 中国电力出版社  2007-9

【4】 面向智能家居的嵌入式语音控制系统的研究[D] 张伟 郑州大学2015(05)

【5】 智能家居语音控制系统的设计[J] 付蔚 唐鹏光 李倩 自动化仪表. 2014(01)

【6】 [科大讯飞:移动互联网进入语音时代](http://kns.cnki.net/kcms/detail/detail.aspx?filename=JSTX201042034&dbcode=CJFQ&dbname=CJFD2010&v=)[J]. 高弋坤.  通信世界. 2010(42)

【7】 [智能家居系统——语音识别](http://kns.cnki.net/kcms/detail/detail.aspx?filename=GDJC201008074&dbcode=CJFQ&dbname=CJFD2010&v=)[J]. 夏霆.  广东建材. 2010(08)

【8】 [智能家居语音控制系统的设计与实现](http://kns.cnki.net/kns/detail/detail.aspx?QueryID=19&CurRec=18&recid=&FileName=1014137821.nh&DbName=CMFD201401&DbCode=CMFD&yx=&pr=)[D] [陈哲](http://kns.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CMFD&sfield=%e4%bd%9c%e8%80%85&skey=%e9%99%88%e5%93%b2&scode=06558688;) 电子科技大学 2013(09)

【9】 《统计自然语言处理（第2版）》[宗成庆](https://book.douban.com/search/%E5%AE%97%E6%88%90%E5%BA%86) 清华大学出版

# 致谢

大学四年结束了，一路走来，磕磕绊绊，有伤心难过，也有兴奋快乐，在这个过程中很多人给了我帮助与指导，真心感谢每一个出现在我生活中的人，每一个帮助我的人。

在这里首先感谢我的导师，不论是论文撰写，还是工作就业都给了我极大的帮助。没有导师的指导，论文写作过程中遇到的很多问题，我都将很难解决。找过导师好多次，每一次导师都是极其耐心的帮我分析每一个问题，梳理我想不清的问题，正是有导师热心的帮助我才能顺利完成论文。在就业选择的时候也咨询了老师意，得到了很宝贵的建议，让我对就业是事也更加清晰明确了。

然后要感谢我实习单位的同事和导师。实习的时候跟着导师学到了很多东西，在导师严格要求和鼓励下，在技术方面我得到了很大的提升。在项目开发过程中也得到了很多同事的帮助，每次有项目难点要攻克，我解决不了的时候，他们都及时的给了我帮助。在他们身上学习到了认真负责的工作态度，和对专业素质不断提升的进取心。

在学校呆了四年，怎么能不感谢培养我的学校、学院以及同学呢。正是有他们的帮助，我的大学四年才过能非常的丰富多彩。不管是学习，还是生活中他们在我这里都担当了很重要的角色，我会记得他们的好。

最后再次感谢所有人，祝大家工作顺心，生活愉快，身体健康，感谢大家出现在我的生命里。

# 附录A:外文翻译原文

Intelligent Speech Control System for Human-Robot Interaction

A speech control system is developed that can better understand human’s speech commands (including recognizing and measuring), and translate them into control inputs. The overall system diagram is shown in Fig. 1. Human visual system observes control errors, and inputs observed errors into the brain. After that, the brain will control the human speech system to express control intentions as inputs of the speech control system, which includes three main parts. The first part is a speech recognition system, which executes the qualitative analysis, that is, recognizes objective contents of speech; the second part is a speech measurement system which conducts the quantitative analysis, that is, measures subjective contents of speech; the third part is a control system which translates fused results of the speech system into control inputs. The speech recognition and measurement system are collectively called as the speech system. These three parts will be explained in detail in subsequent sections.The speech control system can detect control intentions sent by brain, expressed by human speech system and translate speech signals into control inputs (see Fig. 2).Considering the robot as an actuator, and the brain as a controller, which provides an initial control input based on vision guidance of the relative position of the robot and the target. After that, in the process of approaching the target, the brain will adaptively adjust speech inputs so that the robot could reach the target fast and smoothly. The goal of the system control system is to accurately detect speech inputs and translate them into control inputs.

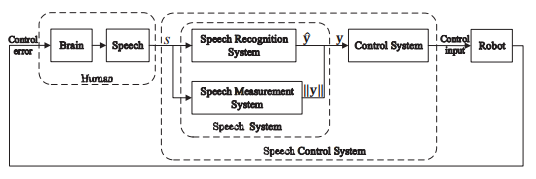


Fig. 1: Speech control system diagram

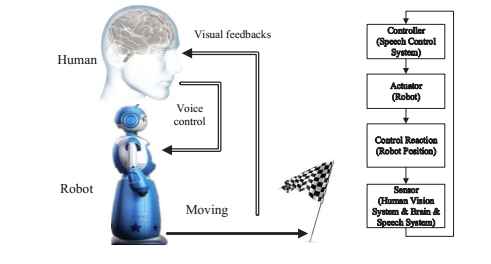
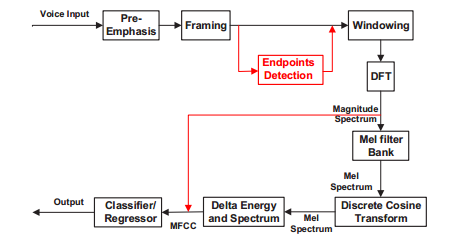


Fig. 2: Speech control system working principle

SPEECH RECOGNITION SYSTEM  
 Assume u = [u1, ..., un] to represent the control input of the robot, where n is the dimensionality of the control input, and ui denotes the ith dimension of u. Every two isolated speech commands are utilized to be mapped into one dimension of the control input as Y = {y1 +, y1 -, ..., yn +, yn -}, where yi + and yi - are corresponding to positive and negative directions of ui. Therefore, to control a system with n dimensions of the control input, there will be totally 2n classes of speech commands that needed to be recognized.

*A. Qualitative Feature Extraction* In the isolated command recognition system, features extracted must eliminate the influence from environment and human subjective factors, including emotion and health conditions, etc. Only objective contents should be reserved. The extraction of the best parametric representation of acoustic  
signals is an important task to produce a better recognition performance. The accuracy of this phase is crucial for the next phase since it directly decide the sign of control inputs. Here, MFCC is applied to represent the acoustic input . The overall process of the MFCC is shown in Fig. 3 marked  
in black.

  
Fig. 3: Comparison of MFCC and proposed spectrum-based features

*B. Speech Recognition by Template Matching* In this section, a map function f1 : s → y ˆ is going to be built, where s is an audio signal, and y ˆ ∈ Rn is an unit vector. Denote yi as the ith dimension of y ˆ, and if s ∈ yi +, then yi = 1; if s ∈ yi -, then yi = -1; if i = i, yi = 0. Furthermore, define u0 as a positive value representing a control value translated from speech inputs, if yi(j) = 1 (yi at time j), then ui = u0; if yi(j) = -1, then ui = -u0. It is worth noting that only one yi = ±1, i = 1, .., n at the moment of there is a speech input. In the next section, we will discuss how to obtain \_x0005\_||y\_x0005\_||, which is the measurement of y ˆ.

DTW algorithm is based on Dynamic Programming techniques . This algorithm aims at measuring similarity between two time series which may vary in time or speed. In this paper, we apply DTW to calculate the distance vectors of MFCC series for measuring the similarity between audio signals, and Euclidean distance to calculate distance vectors of audio feature series. After distances between the current speech feature series and each speech feature series from training templates are obtained, the target of the one in training templates with the maximum similarity will be assigned to the testing sample.

# SPEECH MEASUREMENT SYSTEM

There are two broad types of information in speech. The semantic part of the speech carries objective information insofar that the utterances are made according to the rules of pronunciation of the language. Subjective information, on the other hand, refers to the implicit messages such as the emotional state of the speaker or the control intention, which will be studied in this section.  
*A. Quantitative Feature Extraction* A set of features that can precisely represent control intentions are extremely significant to the accuracy of quantified outcomes. Two possible candidate features are studied as below.  
1) *Energy-based feature* Volume indicates the speech intensity, which could be represented as the amplitude of signal in each frame. It is the most direct way to express control intentions. The essential steps extracting energy-based features are listed as below.  
**Step 1:** Framing  
 The process of segmenting the speech samples obtained from analog to digital conversion (ADC) into a small frame with the length within the range of 20 to 40 msec. The voice signal is divided into frames of N samples. Adjacent frames are being separated by M (M < N). In this paper, we choose M = 128 and N = 256.  
**Step 2:** Endpoints detection  
 Endpoint detection of speech signal is a step directly affecting the accuracy of quantitative outcome. Here, a dual-threshold speech endpoint detection algorithm with the use of short-term energy and short-term average zero-crossing rate is applied .  
**Step 3:** Average active energy  
 The average energy between two endpoints will be calculated as quantified results of the intensity of control intentions to be further mapped as control inputs for the robot.

Several drawbacks of this method exist: (i) not all people express their control intensions through changing volume; (ii) the sound source must be fixed; (iii) it is sensitive to background noise. However, the advantages are (i) the model training process is not needed; (ii) although changing volume may not be a natural way to express control intentions, it is the most direct and easily controlled way under the condition that users know this rule.  
2) *Spectrum-based feature***Step 1:** Framing (the same as above)  
**Step 2:** Endpoints detection (the same as above)  
**Step 3:** Hamming windowing  
 The Hamming window equation is given as Y (n) = X(n)× W(n), where Y (n) is the output signal, X(n) is the input signal and W(n) is the window defined as W(n) = 0.54 - 0.46cos(2πn/ (N - 1)) , 0 ≤ n ≤ N - 1.

**Step 4:** Fast Fourier Transform

Y (ω) = FFT[H(n) ∗ X(n)] = H(ω) ∗ X(ω) (1)

where X(ω), H(ω) and Y (ω) are the Fourier Transform of X(n), H(n) and Y (n) respectively.

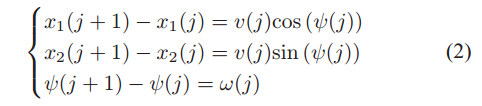
The overall progress of extracting spectrum-based features is shown in Fig. 3 marked in red. Compared with commonly used spectral features, e.g., LPCC, MFCC and LFPC, etc., powers of the spectrum are directly taken as the input of subsequent classifier or regressor, rather than being mapped using a set of given filters and taken DCT to obtain powers sequence in time domain. The motivation of this improvement is that diverse users may emphasis different frequencies to express their control intentions, and the subsequent regressor is able to learn this pattern.

The pros and cons of spectrum-based features are nearly opposite to the ones of the energy-based feature. For its superiority, (i) it is robust to background noise; (ii) being able to learning a natural pattern by which people express control intensions; (iii) the sound source may be mobile. For shortages, (i) the training process is required, that is, a large amount of training data should be provided; (ii) the training target is manmade given, so the calibration error would be another error source for control; (iii) although this method can learn a natural control pattern, the pattern may be adapted to the speaker’s emotion, environment and time, and it is difficult to be expressed deliberately.  
*B. Speech Measurer* In this section, a map function f2 : s →||y|| is going to be built. That is to decide after a set of features are extracted that may represent the intensity of the user’s control intentions, how to map them into a control magnitude. For the energybased feature which is a scalar, only a constant coefficient is needed to linearly map the extracted speech feature into a control input. For spectrum-based features which is a feature vector, a step of regression is required.

Therefore, for spectrum-based features, a regressor named Random Forest is applied here. It is a type of ensemble classification that uses decision tree as the base classifier. RF is chosen as the regressor for several main reasons: (i) high speed (ii) high accuracy (iii) capability to evaluate the importance of each feature variable (iv) being able to handle a feature vector with high dimension, which means the feature selection is not required. Especially, the high computation speed is extremely crucial to guarantee the realtime performance of the speech control system.

# CONTROL SYSTEM

Several control schemes will be described in this section such that the quantitative outcomes of the speech system can be translated into continuous control inputs.

*A. System Dynamics* Consider a linear system dynamics as  
   
where x1 and x2 refers to position coordinates, and ψ refers to the motion direction. Denote the control input as u(j) = v(j) ω(j)\_x0007\_T.

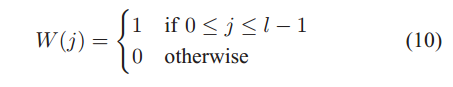
In this paper, as the environment is assumed to be unknown to the robot, control errors cannot be measured directly by the robot, but can be observed by the human. That means the robot can only sense the environment by outputs of the audio sensor. This assumption may be too incredible in realistic; however, it is to motivate the development of a speech-based HRI system used as the auxiliary correction system in real applications.  
*B. Controller Design* A common way to realize this mission is that every time there is a speech command, robot turns a fixed angle, which can be represented as  
 ω(j) = c (3)  
where c is a positive constant. The problem is that a small angle c may render a slow control speed; on the contrary, if c is too large, it may be very difficult to reach the desired direction exactly.

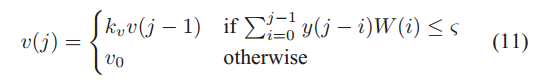
Alternatively, to enable the robot better understand human’s commands, control laws are going to be designed for the speech control system under conditions that control errors can only be perceived by the human, while the robot can indirectly measure control errors through speech signals delivered by the human. Denote  
 f : s → **y** (4)  
as fused outputs of the speech system, including the speech recognition system (f1 : s → y ˆ) and the speech measurement system (f2 : s → \_x0005\_y). Totally consider environmental noise and system noise as ξ, and simply write y\_x0005\_ as y ˆ. The estimation progress of the speech measurement system can be marked as y ˆ = f2(s) + ξ. In this section, the noise is ignored such that ξ = 0. The progress function can be rewritten as y ˆ = f2(s). The human brain map function from observed control errors to control intensions is de¿ned as f3 : ψe → y, which is unknown actually. The main purpose of this section is to ¿nd an approximation of f3, and y ˆ is the estimation of y. We assume y ˆ = y here, which means the outcome of speech system is assumed to be equal to the human control intention. Meanwhile, we assume the linear uncoupling relationship between ψe and y, which simplifies map function y = f3(ψe(j)) into y(j) = a∗ψe(j) at time j, where a∗ is an unknown positive constant. We denote the inverse of a∗ as g∗, that is g∗ = a∗-1. Similarly, g∗ is an unknown positive constant. Then we have ψe(j) = g∗y(j).

Human Adaptive Control System .The control law ω(j) is chosen to be

ω(j) = kyy(j) (5)

where k y > 0 is a constant. In this method, a constant coefficient k y is given to map the measurement result to control direction error, then human brain will gradually figure out this coefficient after several attempts. Therefore, we call this method as “Human Adaptive Control”.

*C. Acceleration strategy* A small value of velocity v(j) may be beneficial to the direction adjustment, while a large value of v(j) may be beneficial to reducing task time cost. To balance two factors, a relative small value of v0 is set in the beginning and a window function W(j) with a length of l is introduced that  


then the acceleration strategy is designed as  
   
where kv > 1 is a constant set for acceleration, and ς is a very small positive value. The reason why we do not set it as zero is to avoid the inÀuence of noise.

# 附录B:外文翻译译文

人机交互智能语音控制系统

语音控制系统用来更好地理解人类语音指令（包括识别和测量），并将其转换为控制输入。整体系统图如图1所示。人类视觉系统观察到控制错误，并输入观察到的错误进入大脑，之后，大脑会控制人类语音系统表达控制意图作为语音控制系统的输入。语音控制系统包括三个主要部分，第一部分是一个语音识别系统，它执行定性分析，即识别语音目的内容;第二部分是语音测量系统，进行定量分析，即测量语音主观内容;第三部分是控制系统把语音系统的融合结果转换成控制输入。语音识别和测量系统统一起称作语音系统，以下部分将详细说明这三部分。语音控制系统可以检测大脑发出由人类语音系统表达的控制意图，并将语音信号转换为控制输入（见图2）。考虑到机器人作为一个执行器，大脑作为控制器基于视觉引导的相对位置的机器人和目标提供初始控制输入，之后，在接近目标的过程中，大脑会自适应的调整语音输入，使机器人可以快速，顺利地理解目标。语音控制系统的目的是为了准确检测语音输入并将其转换为控制输入。

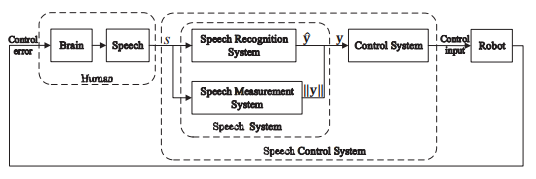


图1 语音控制系统图

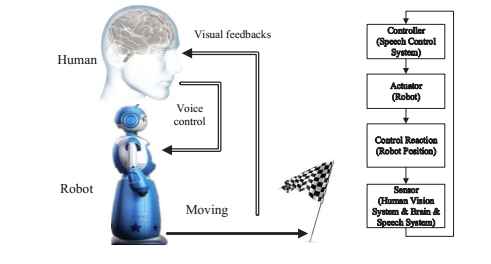


图2 语音控制系统工作原理

**语音识别系统**

假设 u = [u1, ..., un] 表示机器人的控制输入，其中n是控制输入的维度，ui表示u的第i个维度。每两个孤立的语音命令被用来映射成一个控制输入的维度为Y = {y1 +, y1 -, ..., yn +, yn -} ，其中yi +和y1 -对应于ui正和负方向。 因此，为了用控制输入的n个维度控制系统，这将产生2n个需要识别的语音指令类。

1. 定性特征提取

在孤立的命令识别系统中，提取的特征必须消除环境和人类主观因素（包括情绪和健康状况）等的影响，只能保留客观内容。 提取声信号的最佳参数表示是产生更好的识别性能的重要任务。这一阶段的准确性对于下一阶段至关重要，因为它直接决定了控制输入的符号。 这里，MFCC被应用于表示声输入[5]。 MFCC的整体过程如图3中标记为黑色的部分。

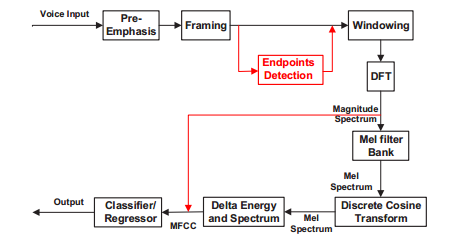


图3 MFCC和提出的基于频谱的特征的比较

B.模板匹配的语音识别

在本节中，将构建映射函数f1：s→y^，其中s是音频信号，y^∈Rn是单位向量。将yi表示为y的第i维，如果s∈yi +，则yi = 1;如果s∈yi-，则yi = -1;如果i= i，yi=0。此外，将u0定义为表示从语音输入转换的控制值的正值，如果yi（j）= 1（在时间ji），则ui = u0；如果yi（j）= -1，则ui = -u0。值得注意的是，在语音输入的时刻只有一个yi =±1，i = 1，...，n。在下一节中，我们将讨论如何获取||y||，这是y的度量。

DTW算法基于动态编程技术。 该算法旨在测量可能在时间或速度上变化的两个时间序列之间的相似性。 在本文中，我们应用DTW计算MFCC系列的距离向量，用于测量音频信号之间的相似度，以及欧几里得距离来计算音频特征序列的距离向量。 在获得训练模板的当前语音特征序列和每个语音特征序列之间的距离之后，将具有最大相似性的训练模板中的目标对象分配给测试样本。

**语音测量系统**

语音中有两种广泛的信息类型。 讲话的语义部分包含客观信息，只要按照语言发音规则进行发音。 另一方面，主观信息指的是隐含的消息，例如说话者的情绪状态或控制意图，这将在本节中进行研究。

1. 定量特征提取

可以精确表示控制意图的一组特征对于量化结果的准确性非常重要。 两个可能的候选特征如下研究。

1. 基于能量的特征

音量表示语音强度，可以表示为每帧中信号的幅度。 这是表达控制意图的最直接的方式。 提取能量特征的基本步骤如下。

步骤1：框架

将从模数转换获得的语音样本分割成长度在20到40毫秒范围内的小帧的过程，语音信号被划分为N个样本的帧，相邻帧被M（M < N）。 在本文中，我们选择M = 128和N = 256。

步骤2：端点检测

语音信号的端点检测是直接影响定量结果准确性的一个步骤。 这里，应用了使用短期能量和短期平均过零率的双阈值语音端点检测算法。

步骤3：平均活跃能量

两个端点之间的平均能量将被计算为控制意图强度的量化结果，以进一步映射为机器人的控制输入。

这种方法的存在几个缺点：（i）并不是所有人通过改变体量来表达自己的控制意图; （ii）声源必须固定; （iii）它对背景噪音敏感。 然而，优点是（i）不需要模型训练过程; （ii）虽然变化的数量可能不是表达控制意图的自然方式，但在用户知道这一规则的条件下，它是最直接和容易控制的方式。

1. 基于频谱的功能

步骤1：框架（与上述相同）

步骤2：端点检测（与上述相同）

步骤3：汉明窗

汉明窗方程给出为Y（n）= X（n）×W（n），其中Y（n）是输出信号，X（n）是输入信号，W（n）是窗口定义为 W（n）= 0.54-0.46cos（2πn/（N-1）），0≤n≤N-1。

步骤4：快速傅里叶变换

Y（ω）= FFT [H（n）\* X（n）] = H（ω）\* X（ω）

 其中X（ω），H（ω）和Y（ω）分别是X（n），H（n）和Y（n）的傅立叶变换。

提取基于频谱的特征的总体进展如图3标记为红色部分。与通常使用的光谱特征（例如LPCC，MFCC和LFPC等）相比，光谱的功率被直接作为后续分类器或回归器的输入，而不是使用一组给定滤波器进行映射，并且采用DCT获得功率时域序列。这种改进的动机是，不同的用户可以强调不同的频率来表达他们的控制意图，并且随后的回归者能够学习这种模式。

基于频谱特征的优缺点与能量特征几乎相反。优点：（i）它对背景噪声是鲁棒的; （ii）能够学习人们表达控制意图的自然模式; （iii）声源可以是移动的。缺点：（i）需要培训过程，即应提供大量的培训数据; （ii）训练目标是人为给予的，因此校准误差将是另一个控制误差源; （iii）虽然这种方法可以学习一种自然的控制模式，但是这种模式可能会适应演讲者的情感，环境和时间，很难被刻意表达。

B.语音测量

在这个部分，将要构建地图函数f2：s→||y||。用来在提取一组可能代表用户控制意图强度的功能之后进行决定，如何将它们映射到控制量级。对于作为标量的能量基特征，仅需要常数系数将所提取的语音特征线性地映射到控制输入中。对于作为特征向量的基于频谱的特征，需要回归步骤。

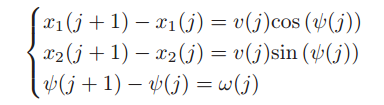
因此，对于基于频谱的特征，这里应用了一个名为Random Forest的回归函数。 它是一种使用决策树作为基本分类器的集体分类。 选择RF作为回归因子的几个主要原因：（i）高速度（ii）高精度（iii）评估每个特征变量（iv）能够处理高维特征向量的重要性的能力，这意味着 不需要功能选择。特别是，高计算速度对于保证语音控制系统的实时性能至关重要。

**控制系统**

本节将介绍几种控制方案，使语音系统的定量结果能够转化为持续的控制输入。

1. 系统动力学

考虑线性系统动力学：



其中x1和x2表示位置坐标，ψ表示运动方向。 将控制输入表示为u（j）= v（j）ω（j）\_x0007\_T。

在本文中，由于环境被假设为机器人未知，机器人无法直接测量控制误差， 但人可以观察到。这意味着机器人只能通过音频传感器的输出来感测环境。 这个假设在现实中可能太不可思议了 然而，它是激励在实际应用中用作辅助校正系统的基于语音的HRI系统的开发。

B.控制器设计

实现这一任务的一个常见方法是，每次有演讲指令，机器人转动一个固定的角度，可以表示为：

ω(j) = c

其中c是正常数。 问题是小角度c可能导致较慢的控制速度; 相反，如果c太大，则可能非常难以准确地达到所需的方向。

或者，为了使机器人更好地了解人的命令，在控制错误的条件下，控制规则将被设计用于语音控制系统，只能由人类，而机器人可以通过人类传递的语音信号间接测量控制误差。 表示：

f : s → **y**

作为语音系统的融合输出，包括语音识别系统（f1：s→y^）和语音测量系统（f2：s→||y||）。

将环境噪声和系统噪声综合考虑为ξ，并将||y||写为y^。语音测量系统的估计进度可以被标记为

y ^= f2（s）+ξ。 在本节中，噪声被忽略，ξ= 0。进度函数可以重写为y^ = f2（s）。从观察到的控制误差到控制意图的人脑图功能被定义为f3：ψe→y，这实际上是未知的。本节的主要目的是对f3进行近似，y^是y的估计。 我们假设y^ = y在这里，这意味着言语系统的结果被假定为等于人的控制意图。

同时，我们假设ψe和y之间的线性解耦关系，其将时间j的映射函数y = f3（ψe（j））简化为y（j）= a \*ψe（j） 其中a \*是未知的正常数。 我们将a \*的倒数表示为g \*，即g \* = a \* -1。 类似地，g \*是未知的正常数。 那么我们有ψe（j）= g \* y（j）。

人类自适应控制系统。控制规则ω（j）被选择为：

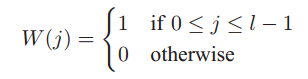
ω(j) = kyy(j)

其中k y> 0是常数。 在这种方法中，给出恒定系数k y，以将测量结果映射到控制方向误差，那么人脑会经过几次尝试后逐渐找出这个系数。 因此，我们称之为“人为自适应控制”。

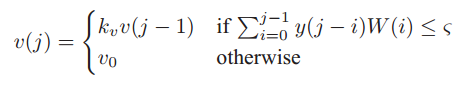
C.加速策略

速度v（j）的小值可能有利于方向调整，而较大的v（j）值可能有益于减少任务时间成本。

为了平衡两个因素，在开始时设置相对较小的v0值，并且引入长度为l的窗函数W（j），



那么加速策略设计为:



其中kv> 1是加速度的常数集，ζ是非常小的正值。 我们不将其设置为零的原因是为了避免噪音的不适。