

穴位动态识别 APP 研发^{*}

耿笑冉 刘秀峰 黄嘉健

(广州中医药大学医学信息工程学院 广州 510006)

〔摘要〕 从主要功能、实现方法等方面介绍基于语音识别技术的穴位动态识别 APP 设计, 测试 APP 性能, 结果表明该 APP 能够有效通过语音识别穴位信息, 有助于普及穴位养生保健知识, 帮助医学生记忆穴位信息, 有很好的实用价值。

〔关键词〕 穴位; 语音识别; PocketSphinx

〔中图分类号〕 R-056 〔文献标识码〕 A 〔DOI〕 10.3969/j.issn.1673-6036.2020.09.015

Study and Development of Acupuncture Point Dynamic Recognition APP GENG Xiaoran, LIU Xiufeng, HUANG Jiajian, School of Medical Information Engineering, Guangzhou University of Chinese Medicine, Guangzhou 510006, China

〔Abstract〕 The paper introduces the design of acupuncture point dynamic recognition APP based on speech recognition technology from the aspects of main functions, implementation methods, etc., and tests the performance of the APP. The results show that the APP can effectively recognize the acupuncture point information through speech recognition, which is helpful to popularize the knowledge of acupuncture point health care and help medical students memorize the acupuncture point information, and has good practical value.

〔Keywords〕 acupuncture point; speech recognition; PocketSphinx

1 引言

针灸是我国特有的治疗方法, 是一种“内病外治”的医术, 通过针灸治疗可起到疏通经络、调和阴阳、扶正祛邪的作用, 是我国中医疗法的瑰宝。采用针灸疗法治疗疾病的过程中, 要求行医者精准找到穴位。随着中医现代化和定量医学的不断推进, 穴位需要有客观和量化的指标进行描述, 其准

确探测对于针灸学发展、分析与辨认相关信号都具有重要作用, 因此人体穴位识别尤为重要。穴位不但可以治疗疾病, 也可用来日常保健^[1]。然而对于非专业人群来说, 精确找到穴位比较困难, 对穴位作用了解不够充分, 很大程度上影响保健效果。对于针灸专业的大学生来说, 穴位信息记忆十分重要, 仅课上时间学习远远不够, 课下仍需要辅助工具来复习。语言识别技术作为一种新兴的智能信息技术引起广泛关注和高度重视^[2], 它可以让机器“听懂”人的语言, 人可以用语音来控制设备, 极大地体现出人机交互的便利性。本研究使用语音识别技术, 研究开发一款穴位动态识别的移动应用, 将计算机技术与中医药学融合, 对降低穴位寻找难度, 普及穴位养生保健知识及帮助医学生记忆穴位信息有很好的实用价值, 具有广阔应用前景。

〔收稿日期〕 2020-02-18

〔作者简介〕 耿笑冉, 本科生, 发表论文 5 篇; 通讯作者: 刘秀峰, 硕士, 教授。

〔基金项目〕 广州中医药大学 2019 年校级课题“基于 AI 人机交互语音识别的针灸穴位动态识别系统”(项目编号: 201910572230)。

2 需求分析

2.1 概述

主要实现以用户输入或语音识别展示穴位具体信息，如功效、主治等，通过与目标用户交流讨论，确定登录注册、手动输入检索穴位、语音识别穴位、收藏和笔记等功能，总体功能模块，见图 1。

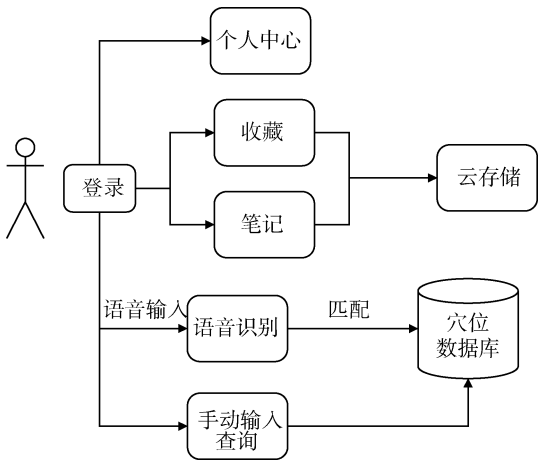


图 1 总体功能模块

2.2 登录注册

每个 APP 必备功能，主要为掌握用户信息。用户需要输入手机号、年龄进行注册，注册完成即可登录，在一定时间内 APP 提供自动登录。

2.3 手动输入检索穴位

界面顶部提供手动查询按钮，用户可输入穴位名称进行查询，输入穴位名称时输入框下方会出现提示，通过智能联想功能使用户快速找到要查询的穴位。保存查询结果并在下次输入时显示，提供历史查询信息。

2.4 语音识别查询穴位

界面底部中央设置语音查询按钮，通过语音识别算法能够更准确地识别用户查询信息；用户输入语音时会有文字提示当前识别的结构，可通过上划屏幕取消语音操作。

2.5 收藏和笔记功能

主要针对大学生，查漏补缺，记忆穴位信息。用户可对不熟悉或者重要穴位进行收藏，有助于快速查看；对穴位重点难点做笔记，反复观看。为使信息同步保存，此两项功能登录才能使用。

3 系统设计

3.1 总体模块

整个系统由注册登录、搜索、收藏、笔记版块组成，APP 具有语音指令搜索和界面输入指令搜索功能，按照功能模块的从属关系，自顶向下展示系统模块层次。对每个模块的功能拆分，见图 2。

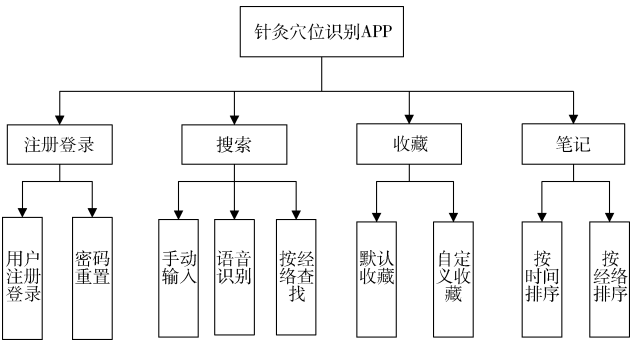


图 2 系统模块层次

3.2 语音识别处理

3.2.1 在线与离线语音识别对比 语音识别处理本质上是一种模式识别匹配的过程，是机器通过识别和理解过程将语音信号转变成相应的文本或命令的技术^[3]。语音听写包括在线和离线引擎两种。在线引擎需要使用网络，速度稍慢，产生一定流量，但有更好的识别效果。离线引擎不需要使用网络，识别速度更快，但需要使用离线的软件开发工具包（Software Development Kit, SDK）^[4]，两者区别，见表 1。虽然对于识别多种语料，离线较在线引擎识别率稍低，但对于只需要特定字典、实现特定功能的需求来说，离线识别往往更有优势，小范围的识别率很高，且识

别速度快，无需联网等。本研究中 APP 只需要识别 穴位或经络名称，因此使用离线识别方式。

表 1 在线与离线语音识别区别

语音识别种类		优点	不足	使用范围	语音识别引擎
在线		识别率高	识别速度较离线的低，需连接网络	大部分商用的 APP，需要识别各种语料	商用引擎
离线		识别速度快，不需要连接网络	识别率较在线稍低，只能识别特定的语料	特定功能的 APP，只需要满足特定功能需要	开源框架

3.2.2 工具 目前有 3 个基于 HMM 和 N - Gram 模型的语音识别工具：CMU Sphinx、Kaldi 和 HTK。其中 Kaldi 和 HTK 更适合部署在服务器端，部分商用语音识别模型在 Kaldi 和 HTK 的基础上改进而来，而 CMU 提供的多种框架属于较小型语音识别框架，经过不断改进，提供比较适合于安卓系统的开源框架 PocketSphinx。由于 APP 基于安卓平台研发，因此使用开源框架 PocketSphinx 作为语音识别的训练初始模型。PocketSphinx 是小型嵌入式语音识别引擎，在 Sphinx - 2 基础上修改、优化而来，是第 1 个开源面向嵌入式中等词汇量连续语音识别项目，对于中小词汇量有很高的识别率和实时性^[5]，适合手持和嵌入式设备。

4 主要功能实现

4.1 穴位库查询

主要分为两种方式：用户在界面左端选择经络，找到该经络上的穴位；用户在界面顶端输入框输入经络或穴位，点击穴位即可显示穴位详情，主要实现代码如下：

```
public void onClick (View v) {switch (v.getId ()) {
case R.id.xuewei_ right_ small_ title;
String inputMessage = input.getText (). toString ();
Intent intent = new Intent (getActivity (), XueweiDe-
scriptionActivi. class);
QueryDataBase queryDataBase = new QueryDataBase (input);
queryDataBase.getMessage ();
String input = new ArrayList < > ();
input.add (queryDataBase.getName ());
intent.putExtra (" input", inputMessage);
startActivity (intent);
break;
}
```

```
startActivity (intent);
break;
}
```

这段代码主要是用户在输入框输入信息，点击查询按钮，查询穴位数据库信息，返回查询结果并存到容器中，跳至信息展示界面。

4.2 语音识别

4.2.1 概述 用户通过语音说出查询的经络或具体穴位名称，经过语音识别模型的处理转换成文本，进而查询到穴位具体信息，不论是手动输入或者语音识别，都会对人体模型上进行相应标记，从数据库中查询到对应的经络穴位信息，反馈到信息详情界面。

4.2.2 获取语言模型 将识别的命令行编写成命令集，新建 txt 文件，编码采用 utf - 8 格式，每个命令占一行，将文件上传，在线生成语言模型。

4.2.3 获取字典模型 利用 CMU sphinx 自带的中文字典，将语言模型的字根据字典添加对应的中文发音，字典模型的作用是使语音识别器记住中文发音。

4.2.4 获取声学模型 若采用 sphinxtrain 训练声学模型会使识别系统对特定用户识别率高，而对其他用户识别率降低，因此利用 pocketsphinx 自带的中文声学模型，即隐马尔可夫模型（Hidden Markov Model，HMM）。

4.2.5 将开源框架导入项目 将框架中 aars 和 models 导入项目，快速集成相关接口。为能够识别中文，新建一个存放中文语音模型的文件夹，再放入前面训练的普通话声学模型、语言模型和穴位名称处理过的文件。

4.2.6 调用获取识别器 初始化语音识别引擎指定必要的语言和字典模型所在路径,并设置存放的位置参数,开启语音收集监听器。初始化语音识别引擎代码如下:

```
private void setupRecognizer (File assetsDir) throws IOException {
    recognizer = SpeechRecognizerSetup.defaultSetup ()
        .setAcousticModel (new File (assetsDir, " ptm - zh"))
        .setDictionary (new File (assetsDir, " xuewei. dic"))
        .setRawLogDir (assetsDir)
        .getRecognizer ();
    recognizer.addListener (this);
}
```

实现开启语音识别功能,通过长按按钮触发长按事件,调用已导入项目中的语音识别引擎,识别结果显示在界面。开启语音识别功能代码如下:

```
public AudioRecordButton (final Context context, AttributeSet attrs) {
    setOnLongClickListener (new OnLongClickListener () {
        @Override
        public boolean onLongClick (View v) {
            //开启语音识别引擎
            PocketSphinxUtil.get (context).start ();
            isReady = true;
            dialogManager.showDialog ();
            isSpeaking = true;
        }
    });
}
```

4.3 收藏

为用户提供收藏穴位信息的功能,分为默认和自定义收藏两类,默认收藏是将用户收藏的信息按经络和时间顺序进行存储,使用户快速找到收藏信息;自定义收藏是根据用户需求,建立文件夹,将穴位信息放在文件夹中。采用 ExpandableListView 实现收藏信息的按类摆放,利用 CardView 实现卡片式的子项,通过 Swipeistview 对每个子项左滑显示删除按钮。部分代码如下:

```
Collection collection = new Collection ();
collection.setXueweiName (input);
collection.save (new SaveListener <String> () {
    public void done (String s, BmobException e) {
        if (e == null) {
```

```
Toast.makeText (XueweiDescriptionActivi.this, " 收藏成功", Toast.LENGTH_ SHORT).show ();
    }
    else {
        Toast.makeText (XueweiDescriptionActivi.this, " 收藏失败", Toast.LENGTH_ SHORT).show ();
    }
}
```

4.4 笔记

利用 ListView 实现对笔记的显示,左滑显示删除按钮,当用户写完笔记点击保存后会显示多条笔记,点击会跳转到笔记详细页面。部分代码如下:

```
public View getView (int position, View convertView, ViewGroup parent) {
    if (convertView == null) {
        convertView = LayoutInflater.from (context).inflate (R.layout.expandablelistview_ children_ item, parent, false);
    }
```

5 系统测试

5.1 运行环境^[6]

基于以上设计,实现基于安卓语音识别的针灸穴位动态识别 APP。为验证该系统实用性,搭建系统运行环境并进行测试,测试设备为小米 5sp,处理器为 4 核 2.35GHZ,操作系统是 Android 8.0.0。

5.2 语音识别功能

测试采用手太阴肺经和足太阴脾经穴位名称为关键词,每个关键词分别测试 3 次,为便于整理与统计,关键词测试顺序先分经络,再按一定顺序排列,共收集 93 个关键词语音样本,5 人分别对其进行测试,结果正确率分别是 90.6%、90.2%、89.7%、90.8%、91%。表明系统较为准确地捕获到用户语音输入信息,能够快速离线识别,实时显示结果,具有很好的实时性。

(下转第 85 页)

4.2.3 建立基于区块链技术的虚拟阅读空间

定期将图书馆新购进和热门的专业知识书籍信息通过空间按照专业和年级进行分类、主动推送,使不同阶段的学生能在课堂外快捷获取专业知识和技能。

4.3 5G + VR 医学网络信息资源库

4.3.1 统一资源库 利用区块链技术分布式存储的优势,整合图书馆、各院系购买和自建各类型信息资源库,构建统一身份认证、资源分类和集群服务模式医学网络信息资源库。

4.3.2 实时在线访问 充分衔接和融合现代信息技术中 5G、虚拟现实 (Virtual Reality, VR) 和区块链技术,突破传统图书馆资源受互联网协议 (Internet Protocol, IP) 限制、虚拟专用网络 (Virtual Private Network, VPN) 访问体验不佳、不能移动化、受地域限制等不足,通过区块链多中心和链链式数据结构将过去“点对点”的网络资源变成虚拟化、移动化的实时在线访问。

4.3.3 数据保护 充分发挥区块链非对称加密算法的优势,对校本特色数字资源库进行信息安全保护,特别是对教学和科研过程中的典型医学案例资料,例如课件、病理诊断影像资料、实验数据、诊疗方案等进行有效保护^[7]。

5 结语

随着高校图书馆服务方式和内容转型,对于区块链技术在医学院校智慧图书馆建设方面的应用研究尚处于探索阶段。要进一步研究和挖掘出更多、更好的以区块链技术为核心的创新服务,需在技术应用、服务方式和手段上继续深入和加强。

参考文献

- 1 周耀. 区块链技术在智慧图书馆中的应用研究 [J]. 现代情报, 2019, 39 (4): 94 - 102.
- 2 袁勇, 王飞跃. 区块链技术发展现状与展望 [J]. 自动化学报, 2016, 42 (4): 481 - 494.
- 3 黄鹂. 智慧图书馆视角下医学图书馆转型思考 [J]. 医学信息学杂志, 2019, 40 (4): 78 - 80, 85.
- 4 陈玲, 任淑敏. 从服务视角看医学智慧图书馆建设 [J]. 医学信息学杂志, 2017, 38 (1): 79 - 82.
- 5 兰建华. 结合信息共享与区块链技术的图书馆服务创新与升级 [J]. 图书馆学刊, 2018, 40 (9): 121 - 124.
- 6 陈小平. 区块链技术在图书馆智慧服务中的应用研究 [J]. 现代情报, 2018, 38 (11): 66 - 71.
- 7 曾子明, 秦思琪. 去中心化的智慧图书馆移动视觉搜索管理体系 [J]. 情报科学, 2018, 36 (1): 11 - 15, 60.

(上接第 77 页)

6 结语

本文详细介绍一款基于安卓平台和语音识别技术的穴位动态识别 APP。测试结果表明该系统可较为准确地识别出用户语音信息。APP 运用人体模型并具备收藏与笔记功能,生动形象,信息具体,帮助学生记忆穴位有关信息。

参考文献

- 1 常梦龙. 基于特征点定位算法的面部穴位定位方法的研究 [D]. 北京: 北京工业大学, 2017.

- 2 唐贵尧, 万鑫. 语音识别技术的研究及基本实现 [J]. 电子技术与软件工程, 2015 (15): 128.
- 3 王爱芸. 语音识别技术在智能家居中的应用 [J]. 软件, 2015, 36 (7): 104 - 107.
- 4 科大讯飞. 科大讯飞 MSC 开发指南: 引擎类型 [EB/OL]. [2018 - 06 - 19]. https://doc.xfyun.cn/msc_android/%E5%BC%95%E6%93%8E%E7%B1%BB%E5%9E%8B.html.
- 5 崔金芳. 基于 OMAP5912 的嵌入式语音识别引擎的研究 [D]. 太原: 太原理工大学, 2009.
- 6 张天祥, 郝凯东, 徐志良. 基于 Android 系统下的夜跑 APP [J]. 电子测试, 2019 (2): 89 - 90, 88.