

基于树莓派的智能“魔镜”设计

潘志倩

(中国传媒大学 互联网信息研究院, 北京 100024)

摘要: 随着物联网技术的发展, 智能家居领域快速崛起, 拥有着美好的发展前景。智能家居产业下智能“魔镜”系统的设计与实现, 旨在丰富优化居家生活品质, 提升居家环境品味。智能“魔镜”以基于 Linux 环境的树莓派为硬件平台, 采用了物联网三层架构设计, 通过感知层的硬件设备进行信息获取采集, 网络层的 HTTP/2 调用云端 API 接口等通信技术进行数据传输和交换, 并由应用层的云计算技术对采集到的信息和数据进行分析处理并反馈。该智能“魔镜”系统经设计与实现后, 运行稳定, 能较好实现主界面信息展示、语音交互功能、实时监控功能等基本功能。

关键词: 树莓派; 物联网; 智能家居; 语音交互; 实时监控

中图分类号: TN91

文献标识码: A

DOI: 10.16157/j.issn.0258-7998.200183

中文引用格式: 潘志倩. 基于树莓派的智能“魔镜”设计[J]. 电子技术应用, 2021, 47(2): 45-48.

英文引用格式: Pan Zhiqian. The design of intelligent Magic Mirror based on Raspberry Pi[J]. Application of Electronic Technique, 2021, 47(2): 45-48.

The design of intelligent Magic Mirror based on Raspberry Pi

Pan Zhiqian

(Internet Information Research Institute, Communication University of China, Beijing 100024, China)

Abstract: With the development of the Internet of Things technology, the field of smart home has risen rapidly, and has bright prospects for development. The design and implementation of the intelligent Magic Mirror system in the smart home industry is designed to enrich and optimize the quality of life at home and enhance the taste of the home environment. The intelligent Magic Mirror uses the Raspberry Pi based on the Linux environment as the hardware platform, and adopts the three-layer architecture design of the Internet of Things. It uses the hardware devices at the perception layer to obtain information and collect, and the HTTP/2 at the network layer calls cloud API interfaces and other communication technologies to implement data transmission and exchange, and the collected information and data are analyzed and fed back by the cloud computing technology at the application layer. The intelligent Magic Mirror system is designed and implemented to operate stably, and can well realize basic functions such as the main interface information display, voice interaction function, real-time monitoring function, and so on.

Key words: Raspberry Pi; IoT; smart home; voice interaction; real-time monitoring

0 引言

基于物联网的智能家居是以家庭为平台, 利用自动控制技术、网络通信技术、智能感应技术、总线技术等, 将家庭生活中的通信设备、电器、家具等设施集中规划的智能化系统, 能为住户提供更便捷、更智能、更安全的居住环境^[1]。现以当下先进的微型计算机树莓派为基础展开课题研究, 在传统家居镜子的功能之上加入了智能化技术, 摆脱传统被动模式, 使之成为具有能动性、可交互的现代化智能工具, 为智能家居研究领域乃至其他领域提出新的产品架构和技术方法, 改善现代生活并创造高效、舒适的新家居情景和生活服务。

1 “魔镜”系统总体框架

智能“魔镜”系统主要由基于 Linux 环境的树莓派^[2]、本地客户端、因特网服务器端三部分构成。如图 1 所示,

硬件设计部分由树莓派主控板、摄像头、液晶显示器屏幕、单向透视玻璃、麦克风、音箱等设备改造和组装而成。树莓派作为控制处理中心, 向运行系统内置的信息流显示客户端、智能语音识别客户端、智能语音交互客户端和实时图像调取客户端等多个平台和程序, 向因特网各个服务平台调用和获取关键性中心数据, 使该智能“魔镜”系统具有选择性丰富的生活信息服务功能, 提供效率较高的安全保障。

智能“魔镜”系统采用物联网三层结构模型^[3], 主要包括感知层、网络层和应用层, 技术架构分别为全面感知、连接传输和智能计算。

2 系统功能论述

本文所设计的智能“魔镜”系统如图 2 所示, 包括镜面信息服务流展示平台、智能语音交互服务、实时监控

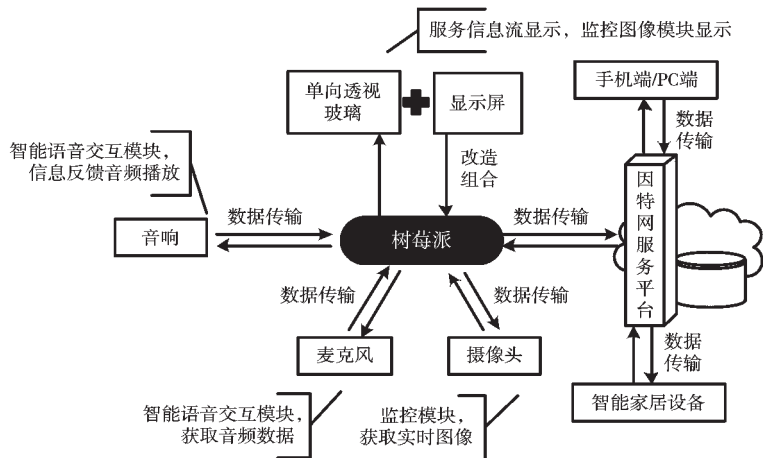


图1 智能“魔镜”系统总框架设计图

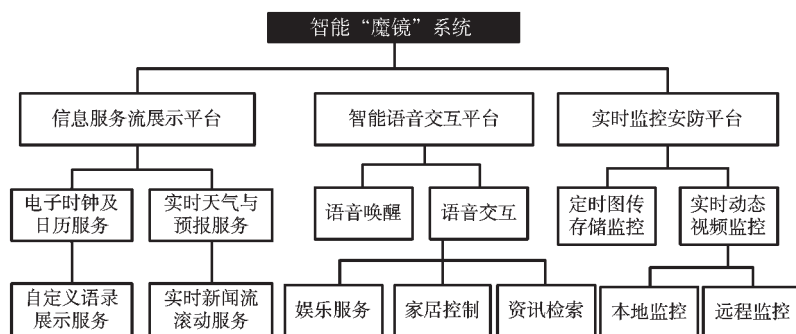


图2 智能“魔镜”系统功能架构

安全防范3个智能家居多功能模块,旨在为用户提供实际、便捷的智能信息服务和准确、实时安全的生活安全保障。

信息服务流展示模块够为用户提供直观及时的电子时钟及日期、实时天气信息及预报、客制化文本展示、实时新闻内容等服务,并定期进行更新。智能语音交互平台模块完成娱乐服务、智能家居控制、生活资讯检索等功能。实时监控安全防范平台模块可定时获取摄像头拍摄的图像,或在镜面、电脑端、手机端等平台看到实时监控画面。

3 硬件设计及实现

本文所设计的智能“魔镜”系统使用的硬件包括核心处理器树莓派、数据流展示界面显示屏、感知层非结构化多媒体数据采集和录入设备麦克风和摄像头、音频数据输出设备音箱等,主要作用是为用户提供者与“魔镜”系统之间进行数据交互式传输,提供输入输出服务。

3.1 核心硬件树莓派

为满足智能“魔镜”系统在家居应用场景中灵活、轻便的需求,本设计选择使用树莓派(Raspberry Pi)3代B+型作为该系统核心处理器和关键硬件平台,接收感应层

采集到的数据并传向应用层,同时接收应用层的反馈数据向使用者进行呈现。

3.2 信息流显像屏

智能“魔镜”信息流显像平台是智能镜显示和人机交互的基础模块,依托显示技术,选择对可见光具有高反射、能实现单向透视原理的单向透视玻璃镜与和液晶屏显示器改造组装而成。既可维持传统镜子的成像特性,也可将镜子后液晶显示屏上的信息显现而出。

3.3 图像视频捕捉传输硬件

影像捕捉硬件连接方式采用了CSI接口连接方法,选取了摄像头RPI Camera作为元器件,该摄像头具有500万像素,模组为OV5647,视场角约为60.6°。使用15PIN FFC即一条十五芯的排线与树莓派的CSI相机接口进行连接。

4 软件设计及实现

智能“魔镜”系统软件设计流程图如图3所示。

4.1 Diet-Pi 系统部署

Diet-Pi是一个专为树莓派设计的纤薄版操作系统。其总镜像(image, Linux系统内核映像文件)大小始于400MB左右,足足轻于树莓派官方提供的Raspbian Lite系统有三倍之多。

经实验可以得出,即使在安装桌面及基本软件之后Diet-Pi系统占用空间亦不到1.3GB。

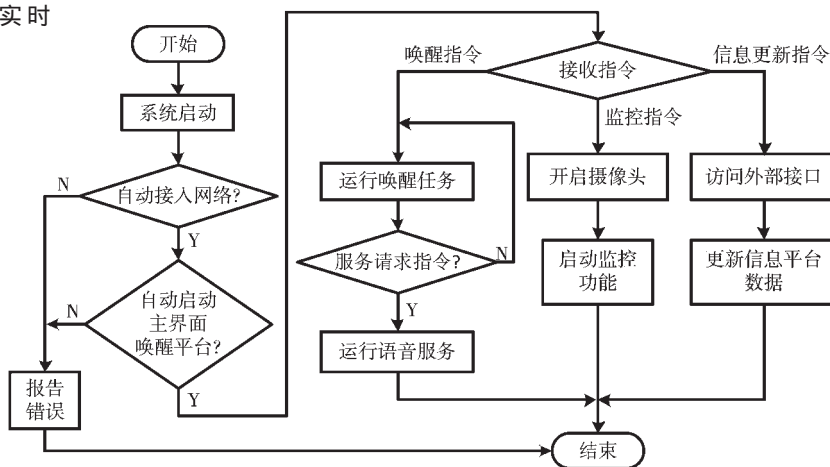


图3 系统软件设计流程图

4.2 信息流展示模块设计与实现

主界面信息流展示模块采用浏览器/服务器模式(Browser/Server),由一个汇集了HTML(超文本语言)、CSS(层叠样式表)、JavaScript等静态资源和提供数据接口API的本地网页构成,浏览器访问该网页后持续不断地通过API接口以TCP/IP的传输协议向相对应的服务器端发出数据获取的请求,而API接口所对应的数据库和应用程序即远程服务器,在建立连接和接收到请求命令

后将会返回响应结果。

4.3 语音唤醒模块设计实现

语音唤醒模块采用 Snowboy 作为唤醒引擎。Snowboy 是一个嵌入式的实时语音监听检测引擎,能够运行一个完整的自动语音识别过程(Automatic Speech Recognition, ASR^[4])来进行触发词的监控检测,语音唤醒流程如图 4 所示。树莓派等设备获取到语音后,会持续监视自动语音识别转录过程中特定触发词的出现,很好地解决了自动语音识别过程消耗大量设备和带宽资源问题,同时为用户使用基于云的解决方案时提供隐私安全保护的服务。

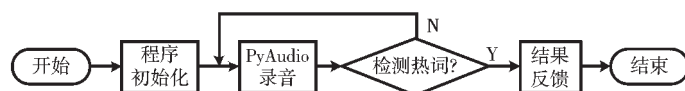


图 4 语音唤醒流程

4.4 实现语音交互模块设计与实现

语音交互模块采用百度 DuerOS 提供的 Linux 版本 DCS 协议的智能设备语音交互开发包完成开发。该 SDK 提供了如有声直播、新闻、生活信息等全方面云服务 BOT 配置,可调用媒体播放器,获取设备信息和对设备进行操控,配合小度之家 SDK 能完成网络配置配对等物联网基础功能。

语音识别流程如图 5 所示。客户端通过 get 请求向服务平台发送数据建立长链接,以 HTTP post 方式上传指令音频。服务器端通过语音活动检测(Voice Activity Detection, VAD^[5])检测语音输入完成后,发送反馈 Directive 推送命令,随即客户端播放反馈音频。



图 5 语音识别流程

4.5 图像监控模块实现

该模块运行图片监控脚本程序代码,定时通过摄像头获取实时图像,基于七牛云的 Python SDK 建立本地客户端并连接互联网后,将本地获取的图片实时上传至七牛云云端存储器。七牛云存储服务是以键值对的方式提供非结构化资源存储服务。图像监控流程如图 6 所示。

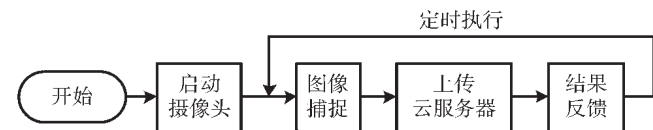


图 6 图像监控流程

4.6 实时监控模块实现

该模块主要采用 Nodejs+WebSocket 和内网穿透的方式实现功能。WebSocket^[6]是 HTML5 的一项持久化的协议,只需要一个 HTTP 握手即可建立长事件的连接。NAT

内网穿透技术^[7]实现不同局域网内的主机进行通信。启动 WebSocket 继电器,在本地终端执行 HTTP-Server 命令将摄像头获取的视频流提供给浏览器,通过内网穿透客户端生成 Ngrok 服务器,启动隧道,部署内网地址映射到域名上,通过域名查看当前摄像头所拍摄的画面。实时监控流程如图 7 所示。

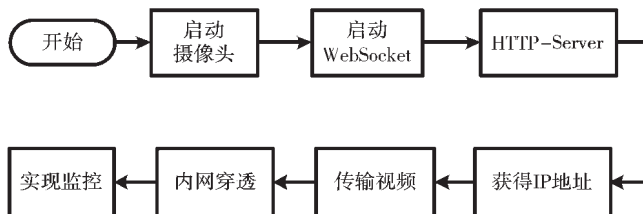


图 7 实时监控流程

5 系统测试与实现效果

最终镜面效果如图 8 所示,整体内容轻重得当,UI 简洁清晰,合理美观的布局确保了使用者能够在短时间内看到关键信息和资讯内容。



图 8 镜面效果图

语音功能测试中,程序能够主动识别语音中所包含的唤醒词,并进入高频率记录语音信息状态,所有语音数据信息都能被程序解析和记录。当使用者说出“播放一段新闻”的指令后,程序开始对该指令进行识别,并将字词解析为文本,上传云端调用相应的资讯信息并播放反馈,最终完成整个语音交互过程。语音终端运行结果如图 9 所示。

监控功能测试中,系统自动定时开启摄像头进行拍摄,使用者能够通过 PC 端或手机端对摄像头实时拍摄的画面进行监看,家居安全防范的基本需求得到满足。

经大量实际测试,“魔镜”系统工作稳定,语音交互成功率为 95%,实时监控部分存在图像延迟 0.5 s~1 s 现象,但能平稳运行。

6 结论

本文主要提出了一个汇集智能数据展示、语音交互生活服务、简易安全防范生活保障等方面的智能家居中心平台解决方案,同时结合了树莓派、网页前端设计、物联网通信技术、云端服务平台等多项技术与服务支持,基于多项


```

on record data:4096
In:0.00\ 00:27:52.68 [00:00:00.00] Out:73.8M [ ===|=== ]
Hd:3.6 Clip:46.1kon record data:4096
In:0.00\ 00:27:52.90 [00:00:00.00] Out:73.8M [ --|-- ]
Hd:3.6 Clip:46.1kon record data:4096
In:0.00\ 00:27:53.00 [00:00:00.00] Out:73.8M [ =| = ]
Hd:3.6 Clip:46.1kon record data:4096

On directive:{
  "directive":{
    "header":{
      "name":"RenderVoiceInputText",
      "namespace":"ai.dueros.device.interface.screen",
      "messageId":"1557145110_742bvu7hr",
      "dialogRequestId":"wp1557145105466"
    },
    "payload":{
      "text":"播放一段新闻",
      "type":"FINAL"
    }
  }
}
function
mplayer directive openFile
["http://xiaodu.baidu.com/speech/api/media?resource_id=3125cc41e5a3061cd51d12854adce12a&token=4e110745095a6a50011a0e..."]
mplayer directive play
On directive:{
  "directive":{
    "header":{
      "messageId":"YXVkw9fbmV3cysxNTU3MTQ1MTEwXzkxMjk5YzF4ag==",
      "name":"Play",
      "namespace":"ai.dueros.device.interface.audio_player",
      "dialogRequestId":"wp1557145105466"
    },
    "payload":{
      "audioItem":{
        "audioItemId": "9187860899188853997",
        "stream":{
          "expiryTime": "2030-01-01T00:00:00+08:00",
          "offsetInMilliseconds":0,
          "progressReport":{
            "progressReportIntervalInMilliseconds":15000
          },
          "streamFormat": "AUDIO_MPEG",
          "token": "eyJib3RfaWQ1OiJhdwRpb19uZXdzIiw1..",
          "url": "http://xiaodu.baidu.com/speech/api/media?resource_id=3125cc41e5a3061cd51d12854adce12a&token=4e110745..."
        }
      }
    }
  }
}

```

图9 语音终端运行结果

硬件的部署和软件的开发设计,基本实现了能够提供多项技能和生活服务功能的智能家居“魔镜”系统。

从创新和实际使用角度分析,该系统以树莓派为平

(上接第40页)

- [10] HASSIN R, HAVIV M. To queue or not to queue: equilibrium behavior in queueing systems[M]. Kluwer Academic Publishers, 2003.
- [11] LI L, REN J, YAN G, et al. An incentive-aware job offloading control framework for multi-access edge computing[J]. IEEE Transactions on Mobile Computing, 2015, 14(8): 2694-2707.
- [12] TSE D, VISWANATH P. Fundamentals of wireless communication[D]. Cambridge University Press, 2005.
- [13] MIETTINEN P, NURMINEN J. Energy efficiency of mobile clients in cloud computing[C]. HotCloud'10 Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, 2010: 1-7.
- [14] KWAK J, KIM Y, LEE J, et al. Dynamic resource and task allocation for energy minimization in mobile cloud systems[J]. IEEE Areas Communications, 2015, 33(12): 2510-2523.
- [15] SPATARU A, RICCI L, PETCU D, et al. Decentralized cloud scheduling via smart contracts: operational constraints and costs[C]. 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS), 2019.

台,在系统运行、外设扩展和网络性能等方面都要比基于51单片的系统或以Arduino为平台的系统更具有优势,同时在运行系统上采用小巧灵动的Diet-Pi系统,不但具有丰富的软件支持,而且在使用和操作上更加简易和方便,是区别于更多人使用相对厚重的Raspbian系统的一次成功的实验和创新。

参考文献

- [1] 刘历.智能家居服务系统的设计与实现[D].南京:东南大学,2017.
- [2] SMITH S. Raspberry Pi assembly language programming[M]. Apress, 2019.
- [3] Ding Fei, Li Zhiliang, Ai Chengwan, et al. Design of an IoT-based efficient security scheme in home wireless system[M]. Springer International Publishing, 2019.
- [4] 李雪林.基于人机互动的语音识别技术综述[J].电子世界,2018(21):105.
- [5] 林奕欧,雷航,李晓瑜,等.自然语言处理中的深度学习:方法及应用[J].电子科技大学学报,2017,46(6):913-919.
- [6] 单振华,王舒憬,强杰.基于Websocket的智能家居远程控制系统的实现[J].电子技术应用,2017,43(10):124-127.
- [7] 陈永东.基于多类型NAT的TCP穿透技术研究[D].成都:四川师范大学,2016.

(收稿日期:2020-03-11)

作者简介:

潘志倩(1996-),女,硕士,主要研究方向:媒体大数据、舆情分析。

- [16] Song Jinyue. Smart contract-based computing resources trading in edge computing[C]. 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, 2020.
- [17] Xu Chenhan, Wang Kun, Guo Mingyi. Intelligent resource management in blockchain-based cloud datacenters[J]. IEEE Cloud Computing, 2018, 4(6): 50-59.
- [18] Xu Jinlai. Zenith: utility-aware resource allocation for edge computing[C]. 2017 IEEE International Conference on Edge Computing (EDGE), 2017: 47-54.
- [19] Song Jinyue, Gu Tianbo, Ge Yunjie, et al. Smart contract-based computing resources trading in edge computing[C]. 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, 2020.
- [20] Liu Tonglai. Smart contract-based long-term auction for mobile blockchain computation offloading[J]. IEEE Access, 2020, 8(99): 36029-36042.

(收稿日期:2020-09-21)

作者简介:

杨戈(1974-),通信作者,男,博士,副教授,主要研究方向:边缘计算, E-mail: yangge@pkusz.edu.cn。

张衡(1999-),女,本科,主要研究方向:边缘计算。

版权声明

经作者授权，本论文版权和信息网络传播权归属于《电子技术应用》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、DOAJ、美国《乌利希期刊指南》、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《电子技术应用》编辑部

中国电子信息产业集团有限公司第六研究所