Design of Intelligent Home Lighting Control System Based on Speech Recognition

Bao Liu[1], Xiaoyang He[1], Yuanqi Li[1], Yuxu Xiao[1], Xin Feng[1], Lingping Chen[1], Jiayuan Lin[1], and Ling Yu[2(✉)]

[1] Research Institute of Photonics, Dalian Polytechnic University, Dalian 116034, China
[2] Network Information Center, Dalian Polytechnic University, Dalian 116034, China
yuling@dlpu.edu.cn

Abstract. Home lighting has always played a very important role in people's quality of life. How to make home lighting more humane is the trend of future development. This article designed a lighting control system that can be controlled by speech. The system receives the voice signal from the microphone, then uploads the voice signal to the server through the Raspberry Pi. The system uses the speech cloud service to identify and analyze the speech signal. The result of speech recognition is transmitted to the STM32 by the Raspberry Pi. STM32 controls single lamp or multiple lamps according to the recognition result. Since the speech cloud provides semantic analysis services, our speech instructions are less restrictive and more closer to everyday language. This system realizes the function of far-field identification and control linkage, while getting rid of the dependence on mobile phones, and the lighting control mode is more humanized.

Keywords: Speech recognition · Cloud computing · Lighting control · Raspberry Pi · STM32

1 Introduction

With the improvement of material levels, people pay more and more attention to high-quality life [1, 2], and smart homes gradually enter daily life. In 2014, the value of the global smart home market was US$20.38 billion, and it is expected to reach 58.68 billion yuan by 2020 [3, 4]. Most of the smart home products on the market now have a single structure, relatively complex control, and need to be operated at close range. The maneuverability and convenience cannot meet the requirements of people at present [5, 6], this article is intelligent. The intelligent control of lighting in the home and the convenience of handling are studied, breaking the limitation of the distance control in the past, and adding the speech recognition, which greatly improves the intelligence.

Supported by Science Foundation for Goldlamp Co., Ltd (2017-228195).

J. Jin et al. (Eds.): GreeNets 2019, LNICST 282, pp. 150–158, 2019.
https://doi.org/10.1007/978-3-030-21730-3_17

2 Principle of Speech Recognition Technology

Speech recognition process mainly includes speech signal preprocessing, feature extraction, training, pattern matching, and identify the output [7, 8]. The training is usually done offline, and the signal processing and feature extraction of the massive voice and language databases collected in advance are obtained. The “acoustic model” and “language model” required for the speech recognition system are obtained and stored as a template library. The recognition phase is usually completed online. The user’s real-time voice signal through the same channel to get the voice characteristics of the parameters, generate test templates, and match with the reference templates. And the highest similarity of the reference template is the recognition result [9–11]. The basic block diagram of the speech recognition process is shown in Fig. 1.

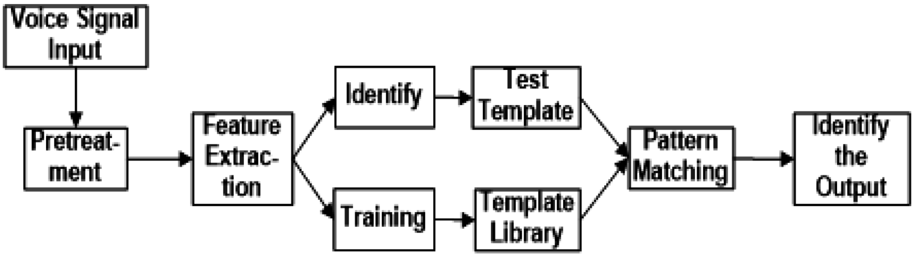

Fig. 1. The speech recognition process

3 System Architecture

The research content of this subject is to realize the intelligent control of the lighting in the home by means of voice. The system block diagram is shown in Fig. 2. The voice is collected through the microphone, and the collected signal is sent to the Raspberry Pi. The Raspberry Pi generates the wav audio file to upload to the cloud server. After the cloud server performs voice recognition, the recognized result is transmitted to the STM32 through the Raspberry Pi, and the STM32 is based on the voice command.

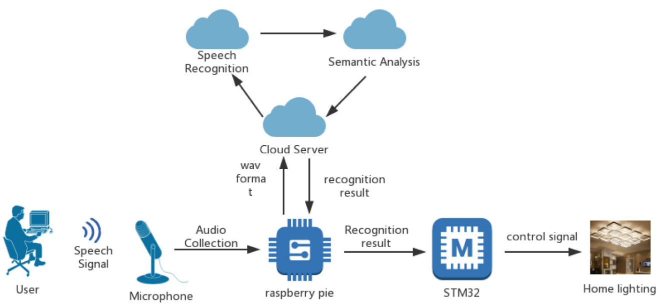

Fig. 2. The block diagram of system

Achieve control of the corresponding luminaire. The areas in the home that need illumination are living room, kitchen, bedroom, bathroom, study, etc. By coding and dividing the lamps in these areas, it is possible to realize the linkage control of multiple lamps with a single lamp.

4 Hardware Selection

The hardware part mainly consists of microphone, raspberry pie, STM32 and LED driver.

4.1 Raspberry Pi

Raspberry pie is an open source microcontroller with very small volume. It can support running Linux and Windows systems. The CPU is an ARM system. It is a popular microcontroller in the Internet of Things. Its minimum system schematic diagram is shown in Fig. 3.

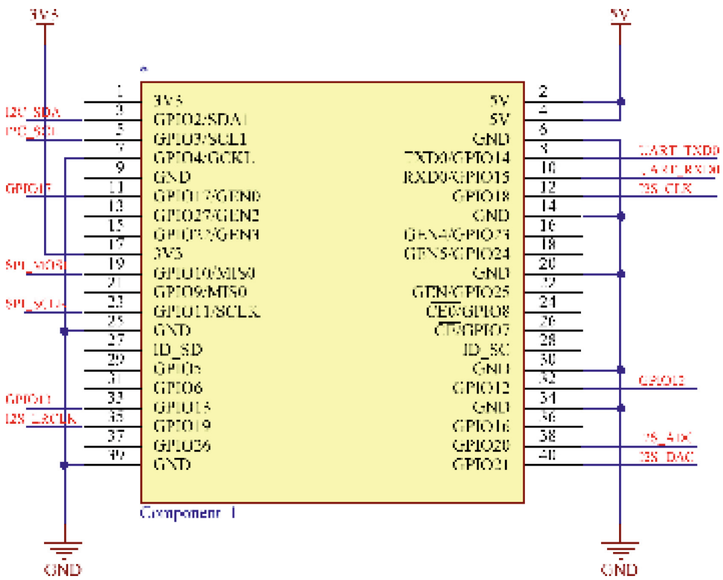

Fig. 3. The minimum system schematic diagram of RPI

Table 1 shows the performance comparison of different types of raspberry pies. As can be seen from Table 1, Raspberry Pi zero w has the advantages of small size, low power consumption and sufficient memory, and its built-in WiFi can solve the networking problem well without inserting wires, making it more independent.

Table 1. Performance comparison of different types of raspberry pie

Content	Type A	Type B	Type B+	Zero w
Soc	BCM 2835			
CPU	ARM11 700 MHz	4 core 900 MHz	1 core 1 GHz	
GPU	Broadcom			
RAM	256 M	512 M	1 G	512 M
USB2.0	1	2	4	1
Video output	Support PAL and NTSC, support HDMI (1.3 and 1.4), Resolution is $640 \times 350 \sim 1920 \times 1200$.			
Audio output	3.5 mm Socket, HDMI Interface			
Borad stage	Standard SD interface			
Net interface	None	RJ45	RJ46, built-in WiFi	built-in WiFi
GPIO port	8	26	40	40
Power	5 V			
Power consumption	Low	High	High	Low
Size	$65 \times 56 \times 10$ mm	$85 \times 56 \times 17$ mm		$65 \times 30 \times 5$ mm

4.2 STM32

STM32 is a high-performance 32-bit microcontroller, a low-cost, high-performance, low-power microcontroller that uses the ARM Cortex-M3 core architecture. The following Figure (Fig. 4) is the minimum system schematic diagram of STM32. STM32 processing speed is 36 M, the power consumption of this MCU is very low, only 36 mA, which is the lowest power consumption of 32-bit MCU products. The product is only quite 0.5 microamps per megahertz. The most noteworthy is that the internal bus of the MCU adopts the Harvard structure, and its execution instructions are quite fast, which can reach 1.25 DMIPS/MHz. This chip is used as their main controller in more and more occasions.

4.3 Microphone

The voice acquisition module uses ReSpeaker 2-Mics Pi HAT. ReSpeaker 2-Mics Pi HAT is a Raspberry Pi dual microphone expansion board designed for AI and voice applications. This module can be applied to many versions of raspberry pie and has good compatibility. The board is a low power stereo codec based on WM8960. There are two microphones on both sides of the circuit board to collect sound, which can realize the function of far-field recognition, and has a longer recognition distance than ordinary microphones. The module provides three APA102 RGB LEDs, one user button and two on-board Grove interfaces for extending applications.

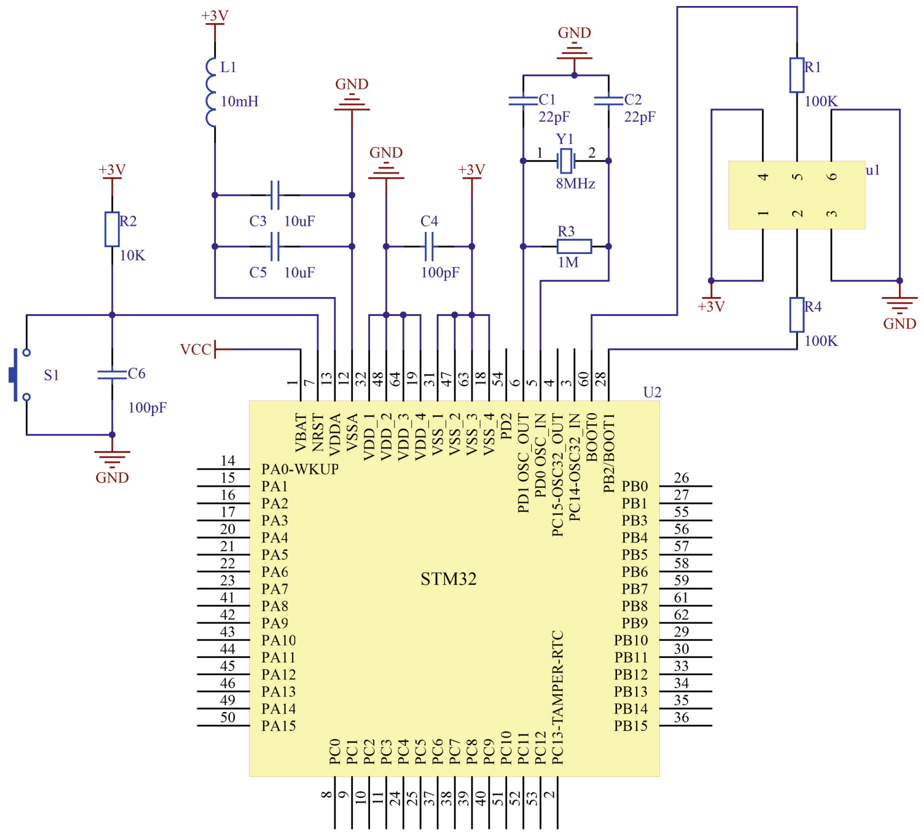

Fig. 4. The minimum system schematic diagram of STM32

5 System Software Design

5.1 Development Environment Configuration

Prepare a 16G SD card and burn the Debian system into raspberry pie using SD_CardFormatter and Win32 diskimager software. Then, we configure WIFI in Linux system and create a new blank document SSH in boot directory to facilitate remote login. After successful WIFI connection, open cmd, login remotely through PUTTY tool, and input user name and password to further configure the system.

5.2 Development Environment Configuration

First, enter the command sudo Su into super user mode to enhance privileges. Enter sudo nano/etc./apt/sources.list to switch the source to Tsinghua. Then, download and install the sound card driver in the raspberry pie. Enter arecord-f cd-Dhw:1 | aplay-Dhw:1 command for recording and playback test, and use the alsamixer command to enter the graphical interface to adjust the volume (Fig. 5).

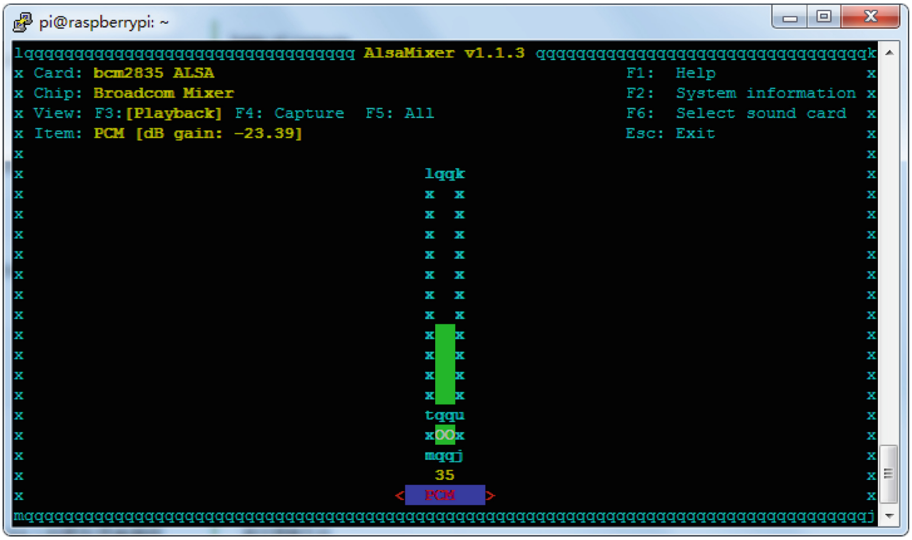

Fig. 5. Volume adjustment interface

5.3 Speech Recognition Environment Configuration

After adjusting the dual array microphone, Python 2 virtual environment can be installed in raspberry pie by downloading and installing commands. The virtual environment is named Env and placed in the ~ directory. Then, configure the Voice engine by executing the following commands in turn.

Enter the kws_doa.py file and modify lines 14–21, 2-Mics, as follows:

```
cd ~/4mics_hat
sudo apt install libatlas-base-dev     #install snowboy dependencies
sudo apt install python-pyaudio     #install pyaudio
pip install ./snowboy*.whl     # install snowboy
pip install ./webrtc*.whl     # install webrtc
cd ~/
git clone https://github.com/voice-engine/voice-engine  #write by seeed
cd voice-engine/
python setup.py install
cd examples
nano kws_doa.py
```

```
from voice_engine.doa_respeaker_4mic_array import DOA
def main( ):
    src = Source(rate=16000, channels=2)
    ch1 = ChannelPicker(channels=2, pick=1)
    kws = KWS()
    doa = DOA(rate=16000)
```

Then, run Python kws_doa.py in a virtual environment. Use snowboy to wake up.

5.4 Speech Recognition Environment Configuration

The flow chart of the system is shown in the Fig. 6. After the system is powered on, the system is initialized, the raspberry pie is connected to the Internet, and the array microphone is in working state. Then, it enters the state of voice detection. When the wake-up words are detected, the user's voice is recorded and the WAV format audio file is produced. The raspberry pie uploaded the audio file after the voice input was completed. When the recognition result is read, the recognition result is matched. If the match is passed, the recognition result is transmitted to STM32 through I2C communication protocol, and the corresponding control instructions are executed. Otherwise, play the audio file "Sorry, I didn't understand, please say it again."

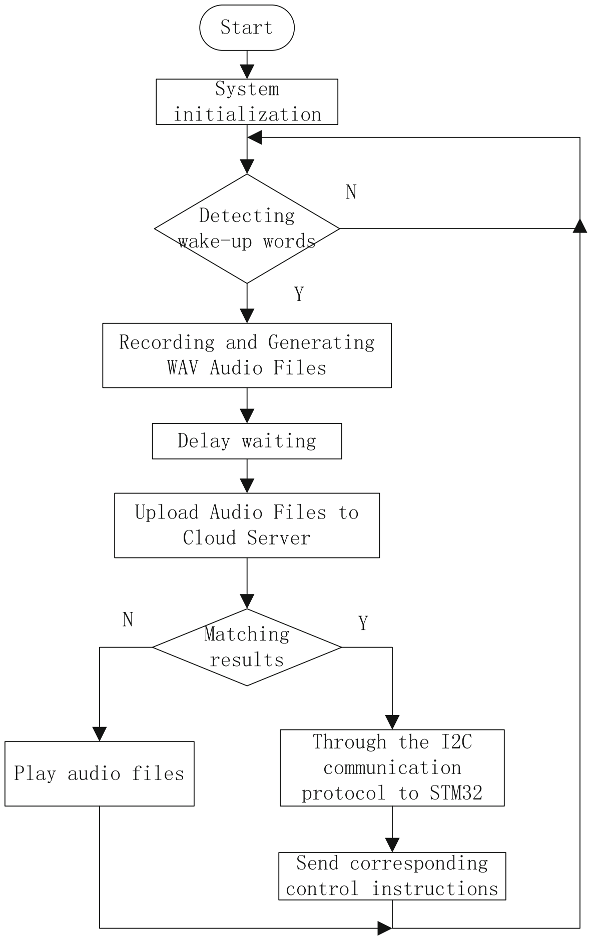

Fig. 6. The flow chart of system

6 Test

In order to test the recognition and practicability of the system, we tested the recognition rate of voice commands at different distances to different sexes. The test results are shown in Table 2.

From Table 2, we can see that within 2 m, the recognition rate of the system is very high, and within the range of 2–3 m, the speech recognition rate is higher than 95%. It has the ability of far-field recognition, and can meet the actual needs of use. The recognition rate decreases with the increase of distance more than 3 m. The main reason is that the effective recognition distance of microphone is limited, and different gender voices have no effect on the recognition rate of the system.

Table 2. Speech recognition rate test results

Distance L/m	Gender	Recognition rate
L < 1	Male	99/100
	Female	99/100
2 < L < 3	Male	97/100
	Female	95/100
L > 3	Male	82/100
	Female	79/100

7 Conclusion

The main content of this project is to achieve the precise control of home lighting through voice. The project uses voice cloud service, array microphone, raspberry pie and STM32 to complete this function. After testing, the system can realize the far-field recognition function and has the actual promotion ability. Unlike the traditional speech recognition system, this system not only realizes the function of voice interaction, but also realizes the linkage control of lighting with embedded technology. The project uses voice cloud service, electret microphone, raspberry pie and STM32 to complete this function. The voice cloud belongs to leasing service, but it needs to be integrated with our system to promote as a whole. In addition, STM32 has very powerful functions, and only part of its functions are used here. Therefore, the system also has a good expansion ability, which is conducive to the further development.

References

1. Wang, H., Bian, H.N., Zhang, H.X., Zhang, F.: The research and design of smart home system based on MQX. In: 2015 International Conference on Computer Science and Applications, pp. 188–190 (2015)
2. Rashid, M.A., Han, X.: Gesture control of ZigBee connected smart home Internet of Things. In: International Conference on Informatics, pp. 667–670 (2016)

3. Petnik, J., Vanus, J.: Design of smart home implementation within IoT with natural language interface. IFAC PapersOnLine **51**(6), 174–179 (2018)
4. Vanus, J., Stratil, T., Martinek, R., Bilik, P., Zidek, J.: The possibility of using VLC data transfer in the smart home. IFAC PapersOnLine **49**(25), 176–181 (2016)
5. Vanus, J., Novak, T., Koziorek, J., Konecny, J., Hrbac, R.: The proposal model of energy savings of lighting systems in the smart home care. IFAC Proc. Volumes **46**(28), 411–415 (2013)
6. Vanus, J., Stratil, T., Martinek, R., Bilik, P., Zidek, J.: The possibility of using VLC data transfer in the smart home. IFAC PapersOnLine **49**(25), 76–181 (2016)
7. Qian, Y., Chang, X., Yu, D.: Single-channel multi-talker speech recognition with permutation invariant training. Speech Commun. **104**, 1–11 (2018)
8. Song, J., Iverson, P.: Listening effort during speech perception enhances auditory and lexical processing for non-native listeners and accents. Cognition **179**, 163–170 (2018)
9. Errattahi, R., El Hannani, A., Ouahmane, H.: Automatic speech recognition errors detection and correction: a review. Procedia Comput. Sci. **128**, 32–37 (2018)
10. Marwan, M., Kartit, A., Ouahmane, H.: A cloud based solution for collaborative and secure sharing of medical data. Int. J. Enterp. Inf. Syst. (IJEIS) **14**(3), 128–145 (2018)
11. Ogawa, A., Hori, T.: Error detection and accuracy estimation in automatic speech recognition using deep bidirectional recurrent neural networks. Speech Commun. **89**, 70–83 (2017)