# Lesson 7 How to use the voice control function

**7.1 Voice control module**

This product uses the function of using voice to control the movement of robot. The method we used comes from the SpeechRecognition of the Github project. Our installation script will automatically install this program and related dependent libraries. If you don’t operate our installation script, please install other dependent libraries before installing it manually:

sudo apt-get install -y swig

sudo apt-get install -y portaudio19-dev python3-all-dev python3-pyaudio

sudo pip3 install pyaudio

sudo apt-get install -y flac

sudo wget https://sourceforge.net/projects/cmusphinx/files/sphinxbase/5prealpha/sphinxbase-5prealpha.tar.gz/download -O sphinxbase.tar.gz

sudo wget https://sourceforge.net/projects/cmusphinx/files/pocketsphinx/5prealpha/pocketsphinx-5prealpha.tar.gz/download -O pocketsphinx.tar.gz

sudo tar -xzvf sphinxbase.tar.gz

sudo tar -xzvf pocketsphinx.tar.gz

cd sphinxbase-5prealpha/ && ./configure -enable-fixed && make && sudo make install

sudo pip3 install pocketsphinx

cd pocketsphinx-5prealpha/ && ./configure && make && sudo make install

After installing the dependent libraries, you can install SpeechRecognition:

sudo pip3 install SpeechRecognition

**7.2 Install microphone**

Insert the microphone module into the USB port of the Raspberry Pi.

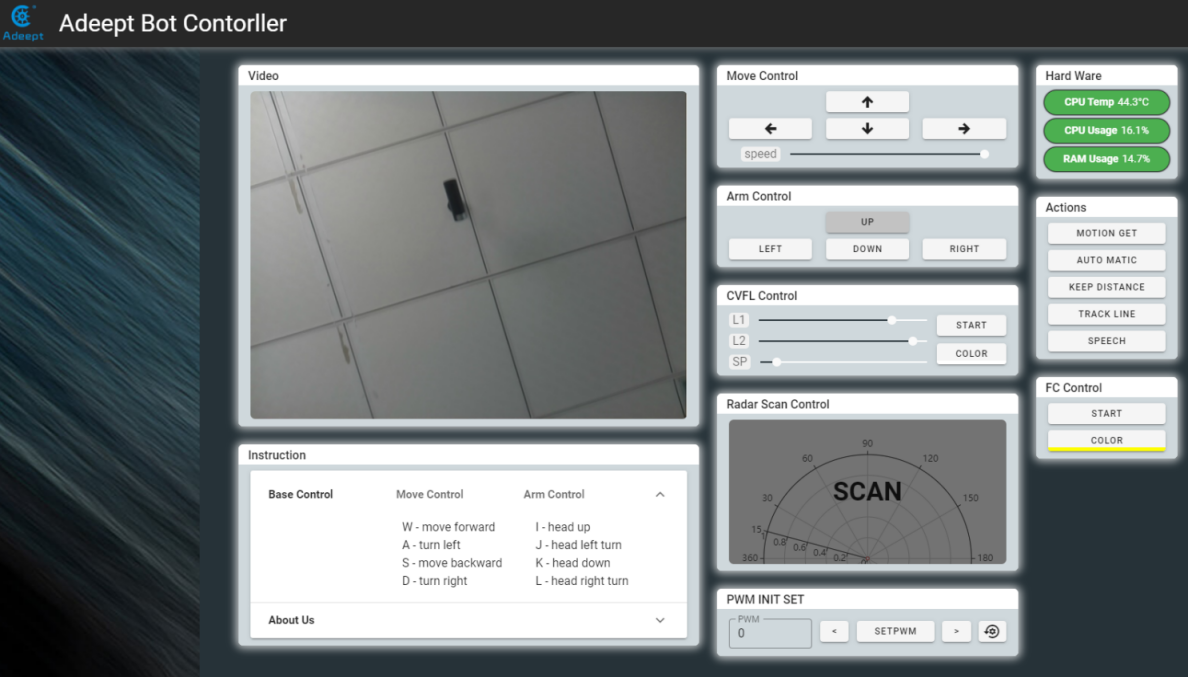


**7.3 Turn on the tracking function**

### Run the tracking program

1. Start up PiCar-b for about 1 minute.

2. When PiCar-b is started up, enter the IP address of your Raspberry Pi on the Google browser of the mobile phone or computer, and access Port 5000, for example: 192.168.3.44:5000. Then the web controller will be displayed on the browser.



1. Click "SPEECH", PiCar-b starts voice detection.

2. Approach the microphone and say "'forward', 'backward', 'left', 'right', 'stop', the microphone detection range is small, and the detection accuracy is poor.

3. After the Raspberry Pi detects the voice command, PiCar-b will act accordingly.

4. When you want to terminate the voice function, you can click "SPEECH" again.

**7.4 The main code program**

Please see the complete code in **speech.py.**

|  |
| --- |
| 1. #!/usr/bin/python3 2. **import** speech\_recognition as sr 3. **import** move 4. **import** RPIservo 5. **import** robotLight 6. **import** time 8. scGear = RPIservo.ServoCtrl() 9. scGear.moveInit() 11. move.setup() 13. RL=robotLight.RobotLight() 15. v\_command='' 16. speed\_set = 80 18. **def** setup(): 19. move.setup()  22. **def** run(): 23. **global** v\_command 24. # obtain audio from the microphone 25. r = sr.Recognizer() 26. with sr.Microphone(device\_index =2,sample\_rate=48000) as source: 27. r.record(source,duration=2) 28. #r.adjust\_for\_ambient\_noise(source) 29. RL.both\_off() 30. RL.yellow() 31. **print**("Command?") 32. audio = r.listen(source) 33. RL.both\_off() 34. RL.blue() 36. **try**: 37. v\_command = r.recognize\_sphinx(audio, 38. keyword\_entries=[('forward',1.0),('backward',1.0), 39. ('left',1.0),('right',1.0),('stop',1.0)])        #You can add your own command here 40. **print**(v\_command) 41. RL.both\_off() 42. RL.cyan() 43. **except** sr.UnknownValueError: 44. **print**("say again") 45. RL.both\_off() 46. RL.red() 47. **except** sr.RequestError as e: 48. RL.both\_off() 49. RL.red() 50. **pass** 52. #print('pre') 54. **if** 'forward' **in** v\_command: 55. scGear.moveAngle(2, 0) 56. move.motor\_left(1, 0, speed\_set) 57. move.motor\_right(1, 0, speed\_set) 58. time.sleep(2) 59. move.motorStop() 61. **elif** 'backward' **in** v\_command: 62. scGear.moveAngle(2, 0) 63. move.motor\_left(1, 1, speed\_set) 64. move.motor\_right(1, 1, speed\_set) 65. time.sleep(2) 66. move.motorStop() 68. **elif** 'left' **in** v\_command: 69. scGear.moveAngle(2, 45) 70. move.motor\_left(1, 0, speed\_set) 71. move.motor\_right(1, 0, speed\_set) 72. time.sleep(2) 73. move.motorStop() 74. scGear.moveAngle(2, 0) 76. **elif** "right" **in** v\_command: 77. scGear.moveAngle(2,-45) 78. move.motor\_left(1, 0, speed\_set) 79. move.motor\_right(1, 0, speed\_set) 80. time.sleep(2) 81. move.motorStop() 82. scGear.moveAngle(2, 0) 84. **elif** 'stop' **in** v\_command: 85. move.motorStop() 87. **else**: 88. **pass** |