AI Voice Interaction: Voice Intelligent Vision

AI Voice Interaction: Voice Intelligent Vision

- Function Introduction
- Function Experience
- Program source code
- Functional principle

Function Introduction

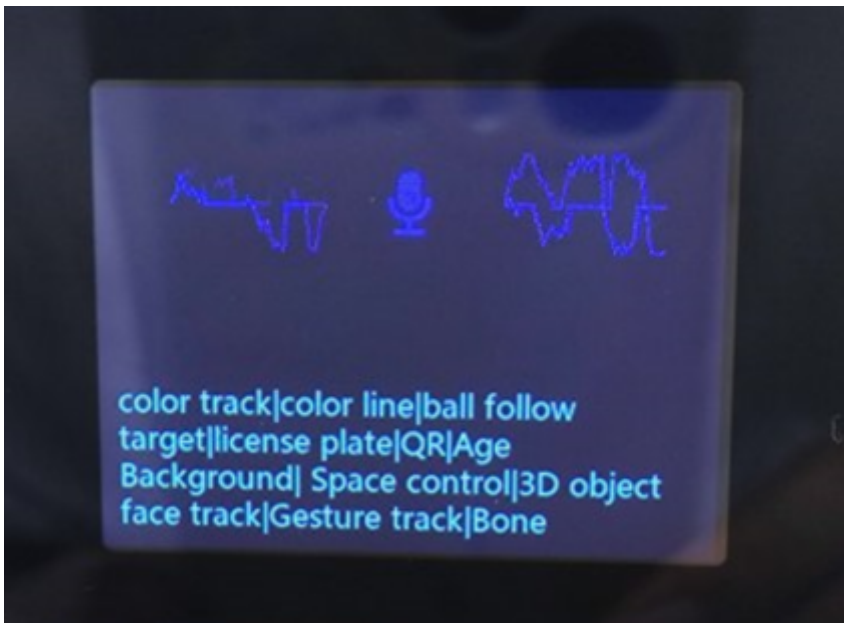
This case is a game of booting up the big program, and the corresponding visual recognition mode is entered through the voice command words of the firmware.

This function requires an Internet connection to run properly

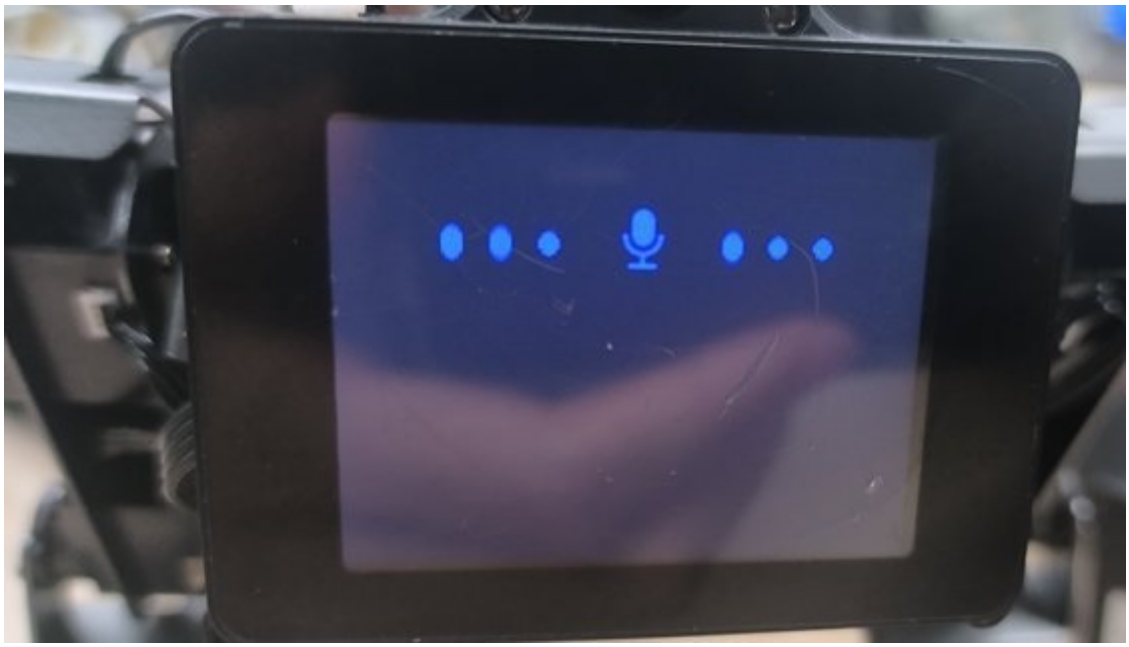
Function Experience

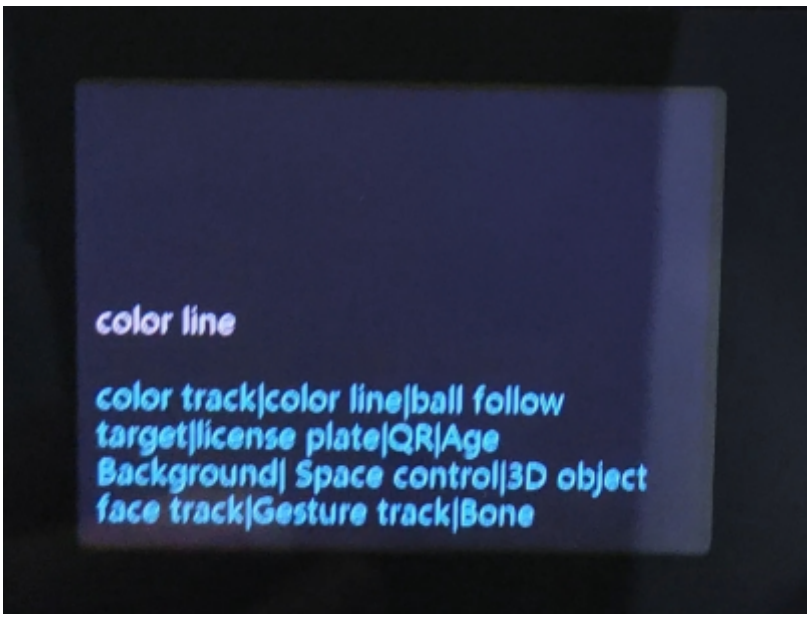
1. Turn on the robot dog first, enter the sample mode by pressing the button on the upper right of the "dog head", and then select the voice intelligent vision function.

2. After entering the voice recognition function, wake up by voice first, "lulu".



3. When you hear a ding, you can say the prompt word at the bottom of the screen.





4. The robot dog will recognize the semantics and enter the corresponding visual mode.

Program source code

1. First, log in to the robot dog system through VNC
2. Then enter the terminal

```
cd /home/pi/RaspberryPi-CM4-main/demos/speechAI/
tree
```

3. Directory structure description

```
├── audio.py #recording file
├── auto_platform.py #system environment dependency
├── language_recognize.py #speech recognition
├── libnyumaya.py #voice wake-up
└── speechAI.py #speech intelligent vision main function
```

```
python main function analysis
def actions_AI(act):
    if act == 0:
        return False

    command_actions_cn = [
            "颜色跟踪",
            "颜色巡线",
            "小球跟随",
            "目标检测",
            "车牌识别",
            "二维码识别",
            "年龄检测",
            "背景分离",
            "隔空控制",
            "三维物体检测",
            "人脸跟踪",
            "手势跟踪",
            "骨骼识别",
            "手势识别",
            "巡线",
            "车牌",
            "小球",
            "人脸",
        ]


    command_actions_en = [
             "color track",
              "color line",
              "ball follow",
              "target",
              "license plate",
              "QR",
              "Age",
              "Background",
              "Space control",
              "3D object",
              "face track",
              "Gesture track",
              "Bone",
```

```
            ]
```

actions_AI: defines the visual function mode executed after speech recognition. If you want to add more custom visual function modes, you can refer to the program writing in **speechAI.py** to add them yourself.
Note:
command_actions_cn: Chinese command terms
command_actions_en: English command terms

Functional principle

The specific flow chart is as follows: