

Smart Dog

Smart Dog

[Function Introduction](#)

[Function Experience](#)

[Program source code](#)

[How to add more commands and actions](#)

[How to replace the large model interface](#)

[How to run this case in the terminal](#)

[Notes:](#)

[How to change the length of the recording time](#)

[How to end a running task](#)

[Functional principle](#)

Function Introduction

This case is the gameplay of starting the big program. You need to complete the configuration of the API-KEY related to the configuration big model before you can use it normally.

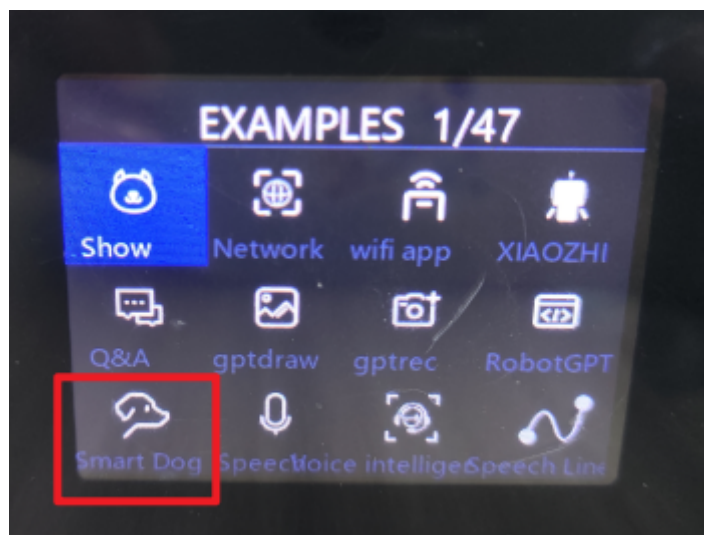
This function also belongs to the embodied intelligence gameplay, adding visual recognition processing gameplay to the multimodal interaction gameplay.

This function needs to be connected to the Internet to work properly

The visual big model and the text big model use different big models

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 smart dog function.



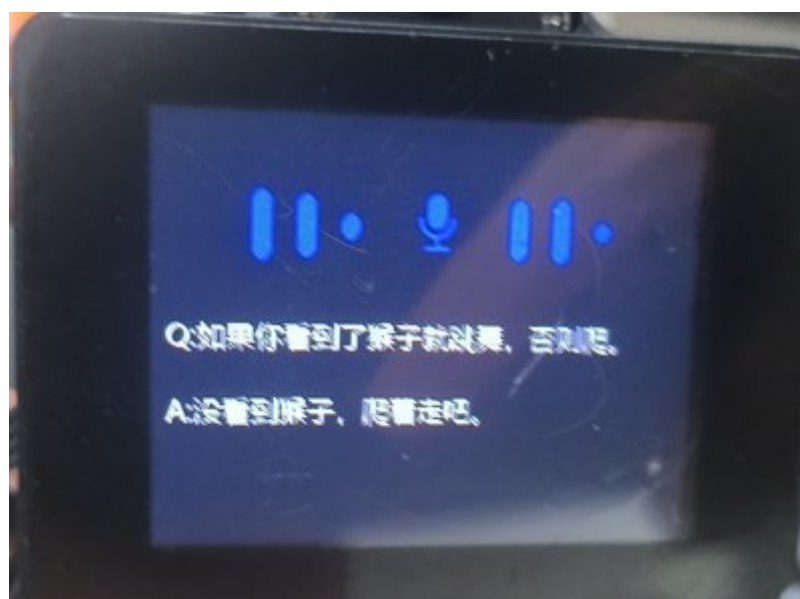
2. After entering the smart dog function, wake it up by voice first, "lulu".



3. When you hear a ding, you can say the meaning of the command.



4. The robot dog will answer and perform actions based on the user's meaning and take a live picture.



Notes

- Once you enter the cases of face detection, bone detection, license plate detection, line patrol, and object tracking, you need to press the button on the upper right corner of the screen to exit the current mode.

For example: face detection is followed by skeleton recognition and then dancing. If you want to enter skeleton recognition after entering face detection, first press the button in the upper right corner of the screen to exit face detection before entering skeleton recognition. The same is true if you want to reach the dance action.

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/dog_agent/  
tree -L 1
```

3. Directory structure description (only the files related to this project are described)

- ├── Alagent_go_image.py #Main function of the intelligent agent - Chinese version
- ├── AIMImage_en.py #Main function of the intelligent agent - English version
- ├── auto_platform.py #Environment dependency
- ├── DAgent_Image_en.py #Intelligent dog's intelligent agent - English version
- ├── dog_agent_Image.py #Intelligent dog's intelligent agent - Chinese version
- ├── dog_API_en.py #Intelligent dog's large model interface - English version
- ├── dog_base_control.py #Basic action control interface
- ├── dog_caw_api.py #Clamping wood block interface
- ├── dog_football_api.py #Football API
- ├── dog_ImageAPI.py #Image API
- ├── dog_record.py #Recording API
- ├── dog_speakiat_en.py #Speech recognition API - Chinese version
- ├── dog_speak_iat.py #Speech recognition API - English version
- ├── dog_tongyiAPI.py #Smart dog large model API - Chinese version
- ├── dog_tts.py #Synthetic audio interface
- └── libnymaya.py #Voice wake-up file

How to add more commands and actions

- First open the DAgent_Image_en.py file to add a sample command

```

27 stretch:dog_stretch()
28 Wave motion:Dog_Wave_Body()
29 Rocking motion:Dog_Swing()
30 handshake:Dog_Handshake()
31 dance:Dog_Dance()
32 Climb Stairs:Climb_The_Stairs()
33 push-up:Dog_push_up()
34 Display robotic arm:Dog_show_arm()
35 The robotic arm moves upwards:arm_up()
36 Robot arm grasping:arm_middle()
37 The robotic arm moves downwards:arm_down()
38 Facial detection: Face_decog()
39 License plate detection: car_license()
40 Human posture detection or bone detection: pose_api()
41 Object detection: yolo'api()
42 Interface for tracking objects: Tarck_Sood (str) # where str represents the object to be tracked, such as tracking the object next
43 Pick up wooden blocks of the specified color, with a total of four colors: red, yellow, blue, and green,For example, picking up gr
44 Pick up wooden blocks of the specified color and place them in their corresponding positions. There are a total of four colors of gr
45 Kick away the balls of the designated color,There are a total of four colors for the balls: red, yellow, blue, and green. For exam
46 According to the specified color, patrol the line and clear obstacles along the way. There are four optional colors: "red, yellow,
47 Scream (Surprise Scream):play_sound_surprised()
48 Scream (Angry Scream):play_sound_anger()
49 Introduce yourself: play_rysel()
50 Rest and wait, such as waiting for two seconds:time.sleep(2)
51 There are also some color related meanings: for example, the sky color is blue, apples are red, bananas are yellow, and leaves are
52 It should be noted that only words related to self introduction and self introduction should be used to call the play_rysel() fun
53 It should be noted that when I ask you what you see, there is no need to call the interfaces for face detection, object detection,
54 Here are some executable action groups
55 【Output JSON format】
56 You can directly output JSON, starting from {, do not output the beginning or end of JSON containing ```
57 In the 'function' key, output a list of function names, where each element is a string representing the name and parameters of the
58 In the 'response' key, according to my instructions and your choreographed actions, output your reply to me in the first person, n
59
60 【Here are some specific examples】 Here are some command statement examples
61 My command: What did you see? You just need to output what you see: {'function': [], 'response ': 'Describe the image '}
62 My instructions: Move forward for 3 seconds, then lie down, show the robotic arm, and finally pee. You output: {'function': ['Dog_fc
63 My instructions:Start exercising.You output: {'function': ['Dog_Squat()', 'Dog_Squat()', 'Dog_push_up()', 'Dog_Wave_Body(
64 My instructions:Turn around and help me pick up the yellow wooden block. You output: {'function': ['Dog_Turn_Around()', 'caw_color_bl
65 My instructions:First, perform three-axis rotation, and then kick the green ball away. You output: {'function': ['Dog_Turn_Around()
66 My instructions:Just describe what you saw, then scream a few times and lie down. You output: {'function': ['play_sound_surprised()
67 My instructions:Move forward for 3 seconds, then move the robotic arm a few times, and finally climb the stairs.You output: {'funct

```

- Then you need to encapsulate some action execution functions and save them in the dog_base_control.py file.
For example, if you encapsulate a forward function, it will be as shown in the figure below.

```

15 ##前进 forward
16 def Dog_forward(delay_time):
17     xgo.move_x(20)
18     time.sleep(delay_time)
19     xgo.stop()
20
21 ## 后退 back
22 def Dog_back(delay_time):
23     xgo.move_x(-20)
24     time.sleep(delay_time)
25     xgo.stop()
26

```

- Finally, add the interface and usage method of the encapsulated function to the DAgent_Image_en.py file.

```

1 from dog_API_en import *
2
3 #DOGZILLA lite Action choreography agent description
4
5 AGENT_SYS_PROMPT = '''
6 You are my mechanical dog butler. Please output the corresponding function to be run and your reply to me in JSON format according to my instructions
7
8 【Here is an introduction to all built-in functions】
9 Forward movement:Dog_forward(time) #Among them, time represents the number of seconds of the action,Advance 1 second:Dog_forward(1)
10 Step back action:Dog_back(time) #Among them, time represents the number of seconds of the action,Step back for 1 second:Dog_forward(1)
11 Left translation action:Dog_Left_move(time) #Among them, time represents the number of seconds of the action,Left shift for 1 second:Dog_Left_move(1)
12 Right translation action:Dog_Rihgt_move(time) #Among them, time represents the number of seconds of the action,Right translation for 1 second:Dog_Rihgt_move(1)
13 Left rotation action:Dog_LeftTurn(time) #Among them, time represents the number of seconds of the action,Rotate left for 1 second:Dog_Rihgt_move(1)
14 Right rotation action:Dog_RightTurn(time) #Among them, time represents the number of seconds of the action,Rotate right for 1 second:Dog_Rihgt_move(1)
15 Looking up action:Dog_Looking_up()
16 Head up movement:Dog_look_straight()
17 Looking down action:Dog_look_down()
18 get down:Dog_get_down()

```

How to replace the large model interface

The large model used in this function is the Chinese version of iFlytek Spark + Tongyi Qianwen (visual processing), and the English version is OpenRouter**

- You can first start from the python version program of the platform corresponding to the interface, and fill in the necessary information according to the platform interface and instructions.

- Then encapsulate the executable file into a function. You can refer to the method of "dog_API_en.py" and put it in the path /home/pi/RaspberryPi-CM4-main/demos/dog_agent. For example, the added file name is "mychatgpt.py"
- Open **DAgent_Image_en.py** and add from mychatgpt import * in the head
- Then find this place and replace it with the API function interface you encapsulated.

```
1 from dog_imageAPI import *
2 from dog_tongyiAPI import *
```

- Then restart the car and enter this function again. You can run the model platform you replaced. If it cannot run, it means that there is an error. You need to check the syntax and logic of the newly added file yourself.

How to run this case in the terminal

1. End the big program first to prevent the screen from being distorted. For how to end it, please go to the tutorial of ending the big program in Chapter 1. It will not be described here.
2. Enter the following command in the terminal

```
cd ~/RaspberryPi-CM4-main
python3 demos/dog_agent/AIMImage_en.py
```

3. When the waiting for keyword appears, wake up the robot dog with "lu lu". After a few seconds, a scrolling prompt "current volume, boot threshold, end threshold" will appear. You can then input voice to the robot dog.

```
pi@raspberrypi: ~/RaspberryPi-CM4-main
File Edit Tabs Help
<module>
  from DAgent_en import * #动作编排 choreography
  File "/home/pi/RaspberryPi-CM4-main/demos/dog_agent/DAgent_en.py", line 1, in
<module>
  from dog_API_en import *
  File "/home/pi/RaspberryPi-CM4-main/demos/dog_agent/dog_API_en.py", line 23
    messages=[
    ^
SyntaxError: invalid syntax
pi@raspberrypi:~/RaspberryPi-CM4-main $ python3 demos/dog_agent/AIMain_en.py
System:Linux
Release:6.1.21-v8+
Machine:aarch64
Uname:uname_result(system='Linux', node='raspberrypi', release='6.1.21-v8+', ver
sion='#1642 SMP PREEMPT Mon Apr  3 17:24:16 BST 2023', machine='aarch64')
LITE
en
en
la is en
start
Loading Library
Initialize Functions
Waiting for keyword...
```

```

120000 55482.0
Current volume: 54055.0, boot threshold: 120000, End threshold: 40000
120000 54055.0
Current volume: 57700.0, boot threshold: 120000, End threshold: 40000
120000 57700.0
Current volume: 55730.0, boot threshold: 120000, End threshold: 40000
120000 55730.0
Current volume: 49108.0, boot threshold: 120000, End threshold: 40000
120000 49108.0
Current volume: 54048.0, boot threshold: 120000, End threshold: 40000
120000 54048.0
Current volume: 56320.0, boot threshold: 120000, End threshold: 40000
120000 56320.0
Current volume: 53656.0, boot threshold: 120000, End threshold: 40000
120000 53656.0
Current volume: 58747.0, boot threshold: 120000, End threshold: 40000
120000 58747.0
Current volume: 58447.0, boot threshold: 120000, End threshold: 40000
120000 58447.0
Current volume: 52563.0, boot threshold: 120000, End threshold: 40000
120000 52563.0
Current volume: 57278.0, boot threshold: 120000, End threshold: 40000
120000 57278.0
Current volume: 62327.0, boot threshold: 120000, End threshold: 40000

```

4. The large model analyzes the recognized voice, prints the reply content on the terminal and broadcasts it. (The picture shows the English version of the reply)

```

pi@raspberrypi: ~/RaspberryPi-CM4-main
File Edit Tabs Help
auto end
The recording has been saved as: ./demos/dog_agent/myrec.wav
tell me what do you like.

Agent
**The intelligent agent arranges actions as follows**
{'function': ['play_ryself()'], 'response': 'I love dancing with my robotic arm
and kicking colorful balls—just like a disco party for robots!'}
**Start speech synthesis and play**: I love dancing with my robotic arm and kick
ing colorful balls—just like a disco party for robots!
Start executing action play_ryself()
Waiting for keyword...
MPlayer 1.4 (Debian), built with gcc-10 (C) 2000-2019 MPlayer Team
do_connect: could not connect to socket
connect: No such file or directory
Failed to open LIRC support. You will not be able to use your remote control.

Playing ./demos/dog_agent/Dogagent.mp3.
libavformat version 58.45.100 (external)
Audio only file format detected.
Load subtitles in ./demos/dog_agent/
=====
Opening audio decoder: [mpg123] MPEG 1.0/2.0/2.5 layers I, II, III
AUDIO: 16000 Hz, 2 ch, s16le, 48.0 kbit/9.38% (ratio: 6000->64000)

```

Notes:

1. If the terminal reports an error timeout, the reason may be that the large model interface is blocked or the network is affected. Just restart the program.


```
pi@raspberrypi:~/RaspberryPi-CM4-main $ python3 demos/dog_agent/AIMain_en.py
System:Linux
Release:6.1.21-v8+
Machine:aarch64
Uname:uname_result(system='Linux', node='raspberrypi', release='6.1.21-v8+', version='#1642 SMP PREEMPT Mon Apr 3 17:24:16 BST 2023', machine='aarch64')
LITE
en
en
la is en
Network check failed: HTTPConnectionPool(host='www.baidu.com', port=80): Read timed out. (read timeout=2)
```

How to change the length of the recording time

1. Enter in the terminal

```
nano ~/RaspberryPi-CM4-main/demos/dog_agent/dog_record.py
```

2. Find here and change the place indicated in the figure below to adjust the recording time according to your environment.

```
231 def start_recording(timel = 3, save_file=SAVE_FILE):
232     global automark, quitmark
233     start_threshold = 120000
234     end_threshold = 40000
235     endlast = 15
236     max_record_time = 20
237     Maximum recording time
238     CHUNK = 1024
239     FORMAT = pyaudio.paInt16
240     CHANNELS = 1
241     RATE = 16000
242     WAVE_OUTPUT_FILENAME = save_file
243
```

Note: `start_threshold > end_threshold`, the adjustment of these two values needs to be adjusted according to your own environment.

How to end a running task

If the reply word is long or the action execution time is long and you want to end the unfinished task in advance, if you are using a large program screen, exit and re-enter the function. If you are running the program in the terminal, exit and re-enter.

Functional principle

The specific flow chart is as follows:

