Rooster
AI development manual

V1.0.0 version

Document change records

Serial number	Change description	Author	Version Number	Date
1	Initial Document	Fu Yaozong	V1.0.0	27 July 2020
2				
3				
4				

Directory

Chapter I Introduction.. 7

1.1. Document profile.. 7

1.2. Network topology and modes of communication.................7

1.2.1. Robot network topology.. 7

1.2.2. Software architecture as a whole...8

1.3. Data content...9

1.3.1. Data content..9

1.3.2. Mode of communication and data header.............................. 10

1.3.3. Data header..10

1.3.4. Communication processes...11

1.3.5. Demo..12

1.3.6. Correct switching methods for robots................................. 12

Chapter 2 Introduction of Debugging and Testing Tools....12

2.1. Introduction to the mapping tool.....................................12

2.2. ADB debugging tools..12

2.2.1. ADB wired commissioning...12

2.2.2. ADB wireless debugging..14

2.3. Introduction to debugging tools..16

2.3.1. Introduction to the tool interface.. 16

2.3.2. Instructions for use..16

Chapter III Android SDK Integration Guidelines.................17

3.1. Introduction..17

3.2. How to integrate..17

3.3. Interfaces and call procedures.. 19

Chapter IV. Detailed description of the interface.................19

4.1. Voice services.. 19

4.1.1. Access to voice services..19
4.1.2. Shut down voice services.. 20
4.1.3. Multiple speech recognition.. 20
4.1.4. Shut down multiple speech recognition...21
4.1.5. Reporting of speech recognition results...21
4.1.6. Reporting of speech and semantic recognition results..22
4.1.7. Voice wake-up call..25
4.1.8. Voice anomaly notification...26
4.1.9. Voice broadcast...27

4.2. Chassis and navigation..28

4.2.1. Get the current location... 28
4.2.2. Chassis movement instructions.. 29
4.2.3. Point-specific navigation...30
4.2.4. Point-specific navigation notifications..31
4.2.5. Point-specific navigation cancellation.. 31
4.2.6. Turning to specific angles.. 32
4.2.7. Step angle...33
4.2.8. Recharging point..34
4.2.9. Setting current speed of movement.. 34
4.2.1 0. Gets the current speed of movement...35
4.2.1 1. Storage maps.. 35
4.2.1 2. Loading map.. 36
4.2.1 3. Map initialization status query..37
4.2.1 4. Navigation status queries...38
4.2.1 5. Navigation mode settings..38

4.3. Face recognition..39

4.3.1. Turn on the video stream..39
4.3.2. Shut down video streaming...40
4.3.3. Camera photography.. 41
4.3.4. Face registration...41
4.3.5. Deletion of face information..42
4.3.6. Face sensing information reporting.. 43
4.3.7. Face recognition information reporting..44
4.3.8. Face database access..45

4.4. Upper body control panel (welcome)....................................46

4.4.1. Robotic limb operation command.. 46

4.5. Upper body control panel (Lesser Snow)..............................48

4.5.1. Robotic limb operation command.. 48

4.5.2. Robot left and right swing open..48

4.5.3. Stop swinging...48

4.6. Configuration information..48

4.6.1. Access to version number.. 48

4.6.2. Acquisition of SN number.. 49

4.6.3. Access to robot types.. 50

4.6.4. Access to robot hardware version information...51

4.7. Alarm related...51

4.7.1. Self-inspection instructions...51

4.7.2. Configuration of robotic sensor enable switches...53

4.8. Other orders.. 54

4.8.1. Heartbeat request.. 54

4.8.2. Robot power acquisition.. 55

4.8.3. Robot emergency state acquisition...56

4.8.4. Robot emergency push... 56

4.8.5. Robot charging status acquisition...57

4.8.6. Robot charging status push.. 58

4.8.7. Message broadcasts.. 58

4.8.8. Motor overload query command...60

4.8.9. Motor overload clearance order..60

4.8.1 0. Motor overload push...61

4.9. Expression settings..61

4.9.1. Setting up facial expressions...61

Chapter V Best Practices... 62

5.1. Voice-to-answer process...62

5.2. Face registration process... 63

5.3. Face recognition process.. 63

5.4. Navigation processes... 64

Chapter VI Frequently Asked Questions................................ 64

Chapter 1 Guidebook

1.1. Document Overview

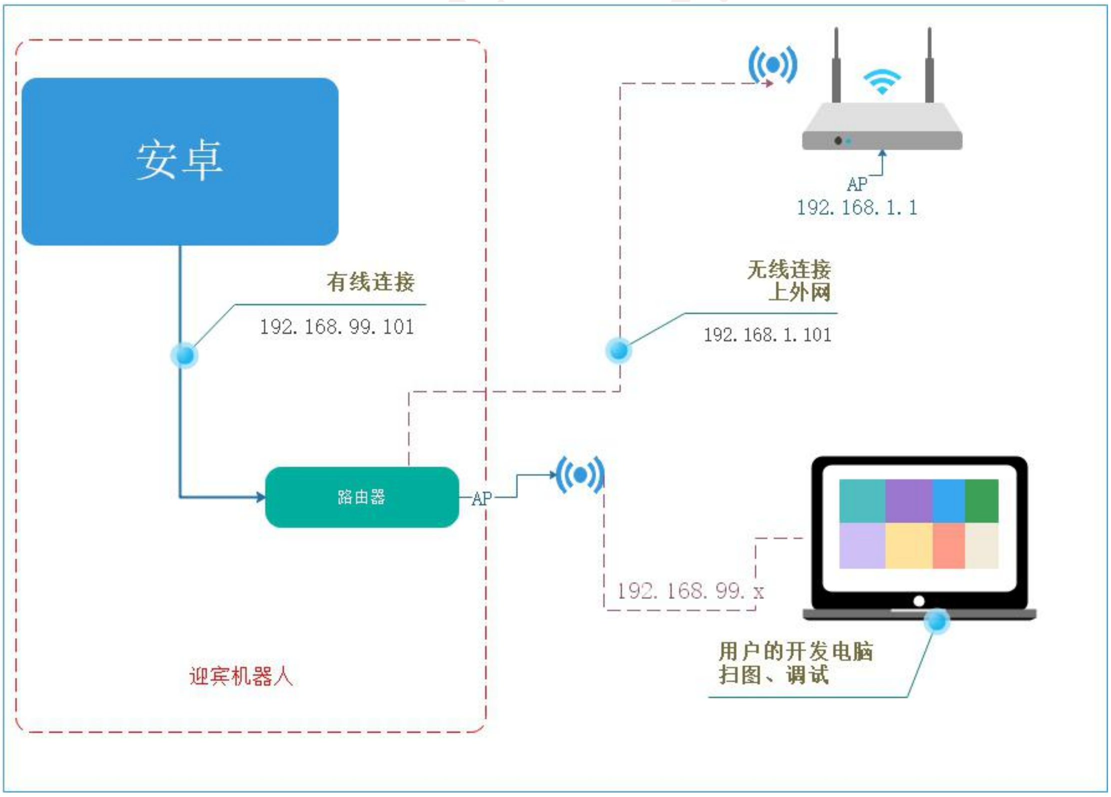
This document mainly provides the SDK interface usage method of pangolin entity robot for third party manufacturer to use robot to complete custom requirement.

Suitable models, drawings

1.2. Network topology and communication

1.2.1. Robot Network Topology

The robot has a router built in and is connected by a built-in router.

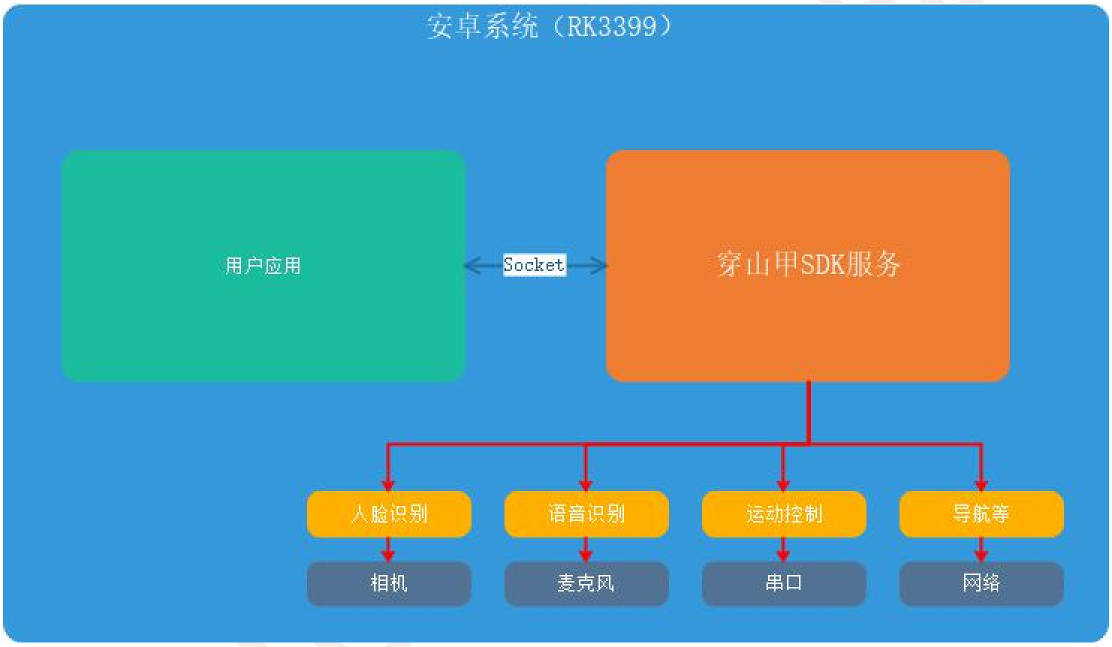


The IP segment of the intranet is 192.168.99. x, generally speaking, the IP of the robot Android screen is 192.168.99.101.

The robot router is connected to the customer's wireless AP by bridge mode (this port is wan port on the robot router), and can be used as a AP, sharing hot spot for computer connection such as scanning, debugging and so on.

If the robot can access the Internet, the computer connected to the robot router can also access the Internet, when the robot's Android system, the user's development computer and the robot's internal network equipment are in a local area network.

1.2.2. Software architecture



The pangolin SDK service combines face recognition ability, speech recognition, semantic understanding ability, motion control ability, through Socket and customer application communication, essentially [pangolin SDK service] is a TCP Server, so it can communicate through all accessible networks according to protocol.

Robot Bottom System

This paper mainly explains the interaction between the robot bottom system and the upper application. It is worth noting that if the upper computer is android,, please use the tts voice mode provided in the android end demo, and the tts mode provided in this paper is invalid.

For foreign development customers: the speech service in this paper is only for China, if you want to develop your own non-Chinese speech recognition engine, there are two ways:

1. contact sales, using the international voice engine supported by Android

2. refer to Android demo, find their own voice recognition engine for voice services.

1.3. Data content

1.3.1. Data content

All messages are in json format, as follows:

Request data sample

```
{
  "msg_id"：  "XXX_XXX _REQ"
}
```

Parameters	Note	Type
m sg _id	Message type	String

Returns a result sample

```
{
 "error_code"：  0,
 "msg_id"：  "XXX_XXX_RSP"
}
```

Parameters	Note	Type
m sg _id	Message type and message counterpart	String
error_code	Error code	i nt

m sg_id message categories

Message suffix	**Meaning**
REQ _	To indicate a request (req uest), generally in pairs with a RSP
RSP _	Response (r e sp onse), generally paired with REQ .2
CMD _	An instruction (c o m man d) that does not necessarily have a reply
NTF _	A notification (n o t i f y), mainly used for the underlying message push

1.3.2. Mode of communication and data header

TCP network protocol is used in all interfaces, and the interface supports both common socket and websocket.

Server address :192.168.99.101

Porto	Port description
60001	Websocket (for web end, no data header required)
60002	Add a header to the S ocket((s)

1.3.3. Data header

The data header must be added and parsed when sending and receiving data if normal socket,, websocket please ignore it.

Full data format:

Header (24 bytes)	Data content

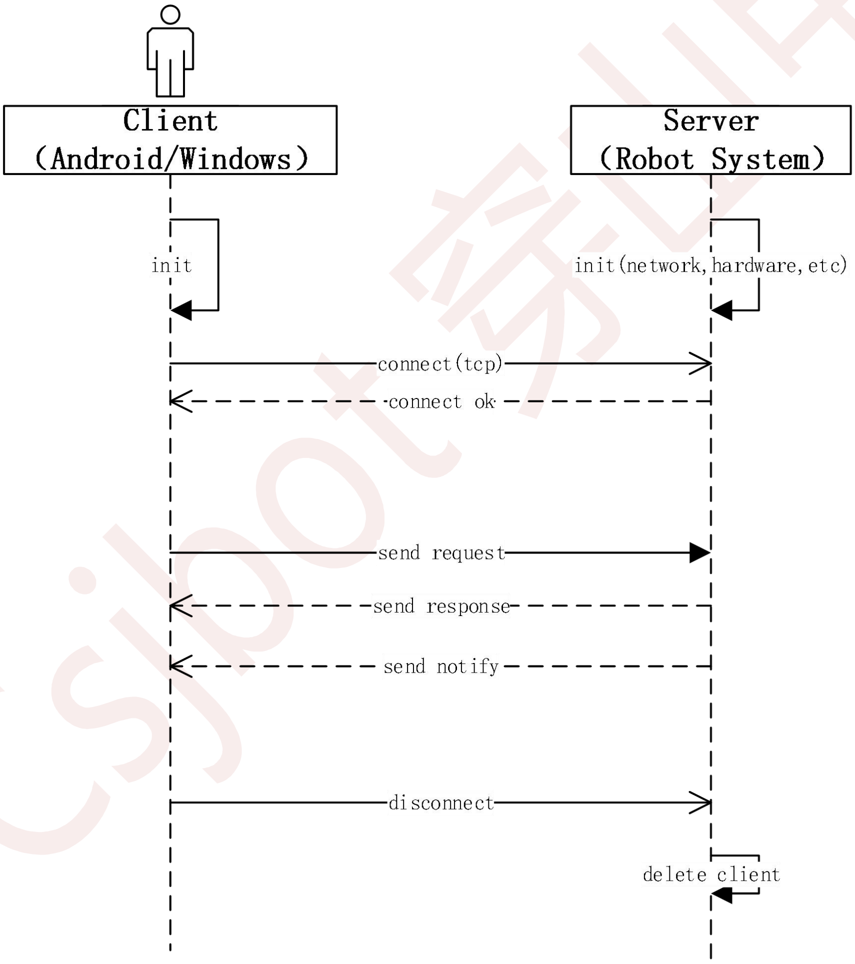
The header format is as follows:

Section Title	Data content	Value	Data definition
1~4	Int	1(fixed value)	Acceptance node
5~8	int	1(fixed value)	Send node
9~12	int	1(fixed value)	id accepted
13~16	int	25(Fixed value)	id sent

17~18	short	**1(fixed value)**	id of messages
19~20	short	**7002(fixed value)**	Event id
21~24	i nt	Requirements	Length of data content sent (excluding header)

Note: send using large-end mode, specific use can refer to the demo program.

1.3.4. Communications process



1. Client can keep trying to connect the bottom Server, of the robot until the connection succeeds
2. C lient send req requests S erver respond to the corresponding Rsp
3. Server side pushes NTF messages asynchronously
4. Client side can choose to manually disconnect, but do not need

PS： if the host computer is Android, call the native method to switch Ethernet repeatedly until the ip address is obtained, otherwise the tcp connection may not be established.

1.3.5. Demo

Android. are Demo versions currently supported If necessary, please contact after-sale, you can see some call process at the end of the document.

Customers can use sdktool to simulate sending and receiving tests of instructions.

1.3.6. Correct Switch Machine Method for Robot

Boot: if there is a charging pile, the charging pile position must be fixed and each time the robot starts on the charging pile, if there is no charging pile, please also fix a position to boot, because the robot automatically matches the map when it starts to restore the map.

Shutdown: please call the robot shutdown instructions to shut down, do not press the button to achieve hard shutdown, hard shutdown may cause map loss and other problems.

Chapter 2 Introduction of debugging and testing tools

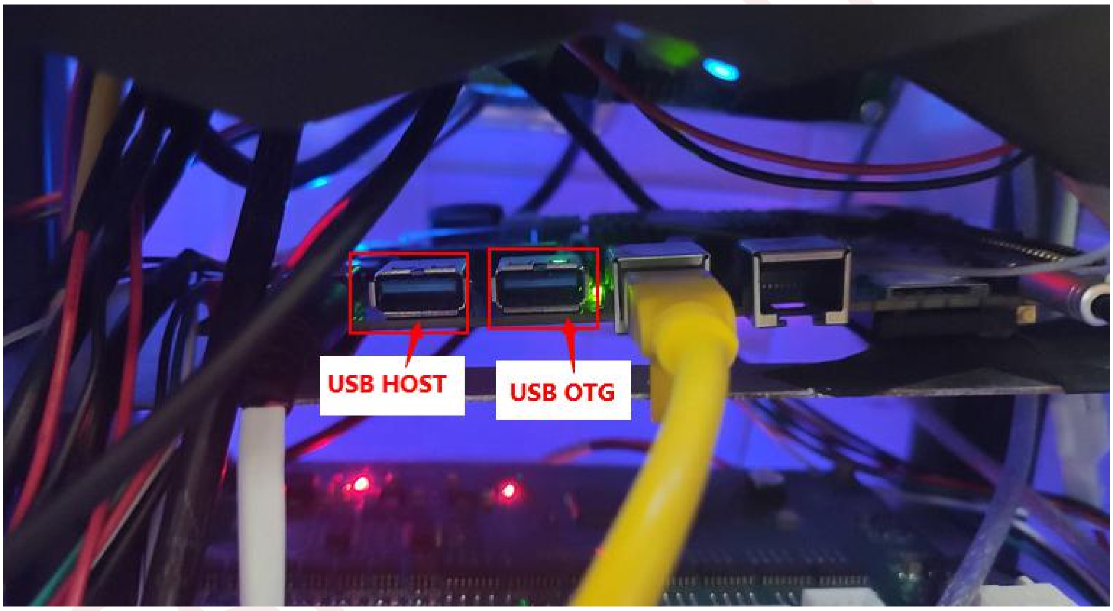
2.1. Introduction to Scanner Tool

2.2. ADB debugging tools

2.2.1. ADB wired debugging

Robot support Android wired debugging, need to open the robot back cover, through the double-headed USB line connection,

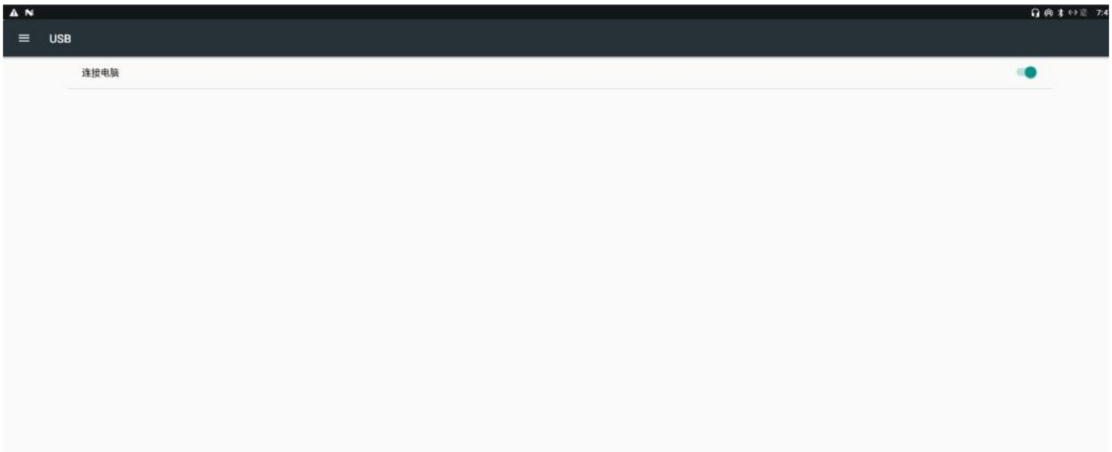
The steps are as follows:



a、 Open the robot rear cover

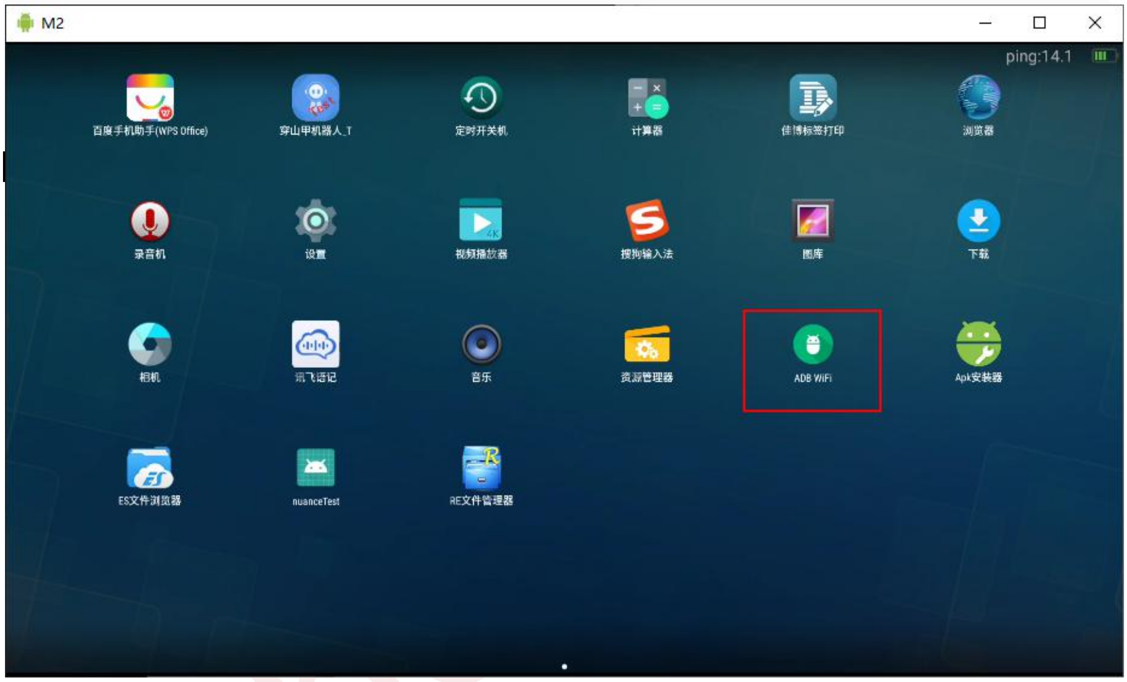
b、 Look for the Android version, find the USB OTG port, insert one end of the data line USB OTG, insert one end into USB computer port

c、 Go to Android settings, find USB options, and turn on the Connect computer



USB

连接电脑 ●

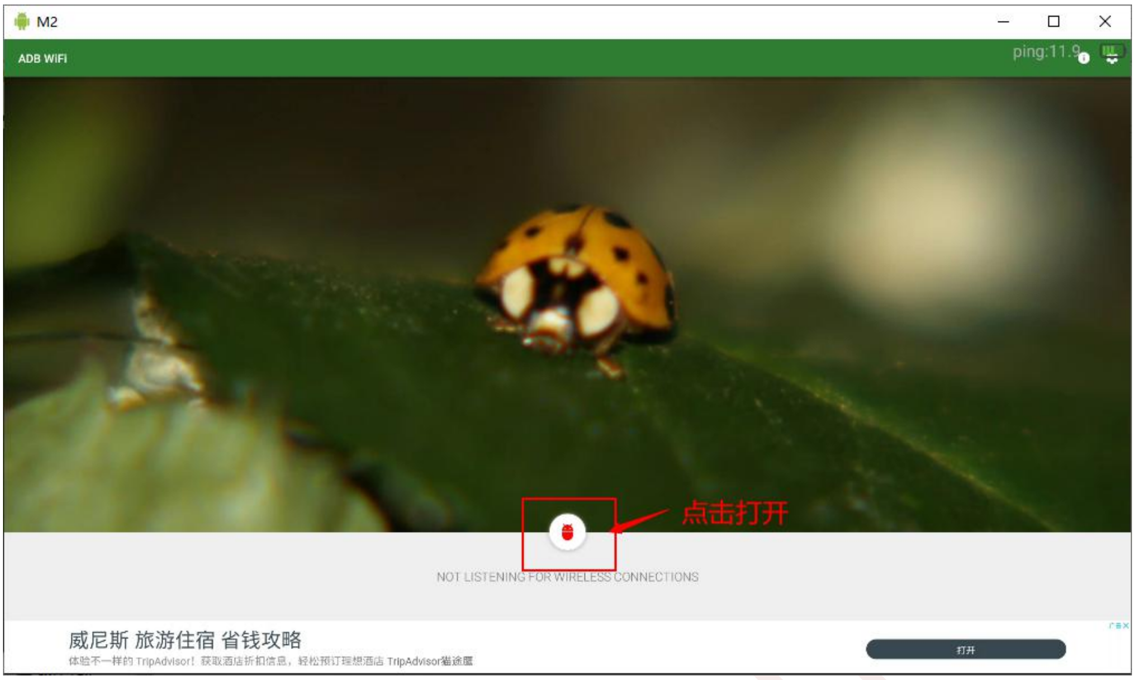


2.2.2. ADB wireless debugging

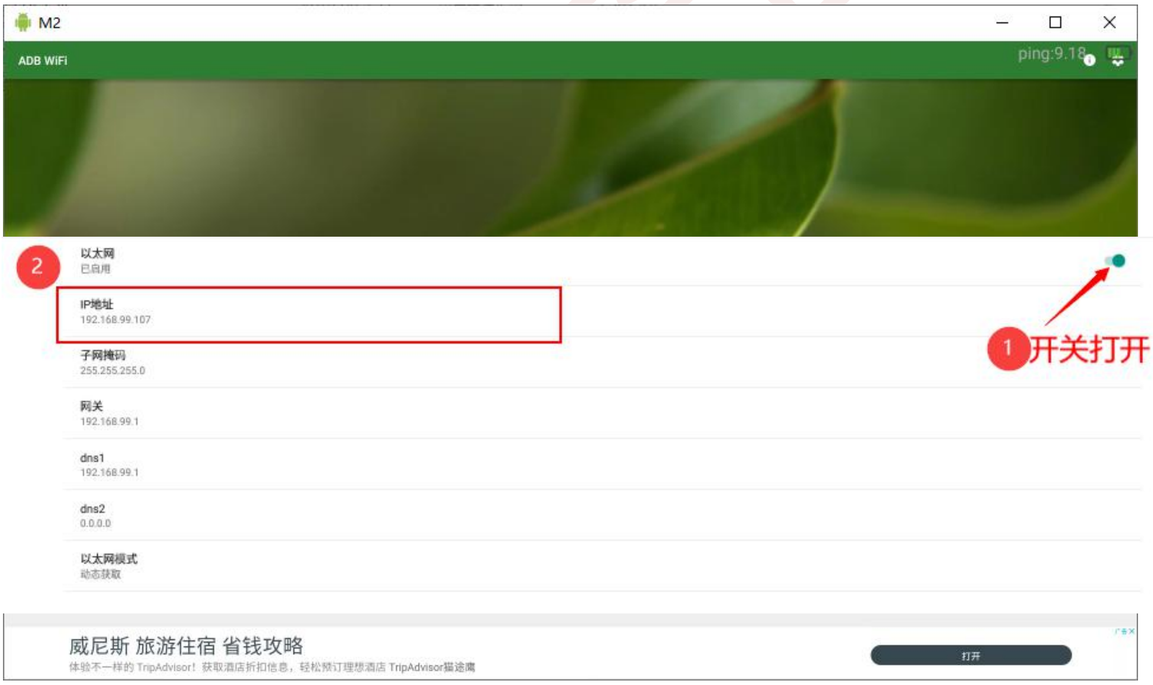
If there is no condition to use data lines for debugging, you can also use ADB WIFI tools for debugging

a、Installation ADB WIFI Application adb-wifi-eth.apk》 Click Download

b、Open the ADB WIFI app, click the red button, and open the ADBWIFI


M2
ADB WIFI
ping:11.9
点击打开
NOT LISTENING FOR WIRELESS CONNECTIONS
威尼斯 旅游住宿 省钱攻略
体验不一样的 TripAdvisor！获取酒店折扣信息，轻松预订理想酒店 TripAdvisor猫途鹰
打开

Success turns green


M2
ADB WIFI
ping:9.18
2
1 开关打开
威尼斯 旅游住宿 省钱攻略
体验不一样的 TripAdvisor！获取酒店折扣信息，轻松预订理想酒店 TripAdvisor猫途鹰
打开

以太网
已启用

IP地址
192.168.99.107

子网掩码
255.255.255.0

网关
192.168.99.1

dns1
192.168.99.1

dns2
0.0.0.0

以太网模式
动态获取

c、View IP address, typically 192.168.99.101, if not, click System Settings → Ethernet

2.3. Introduction to debugging tools

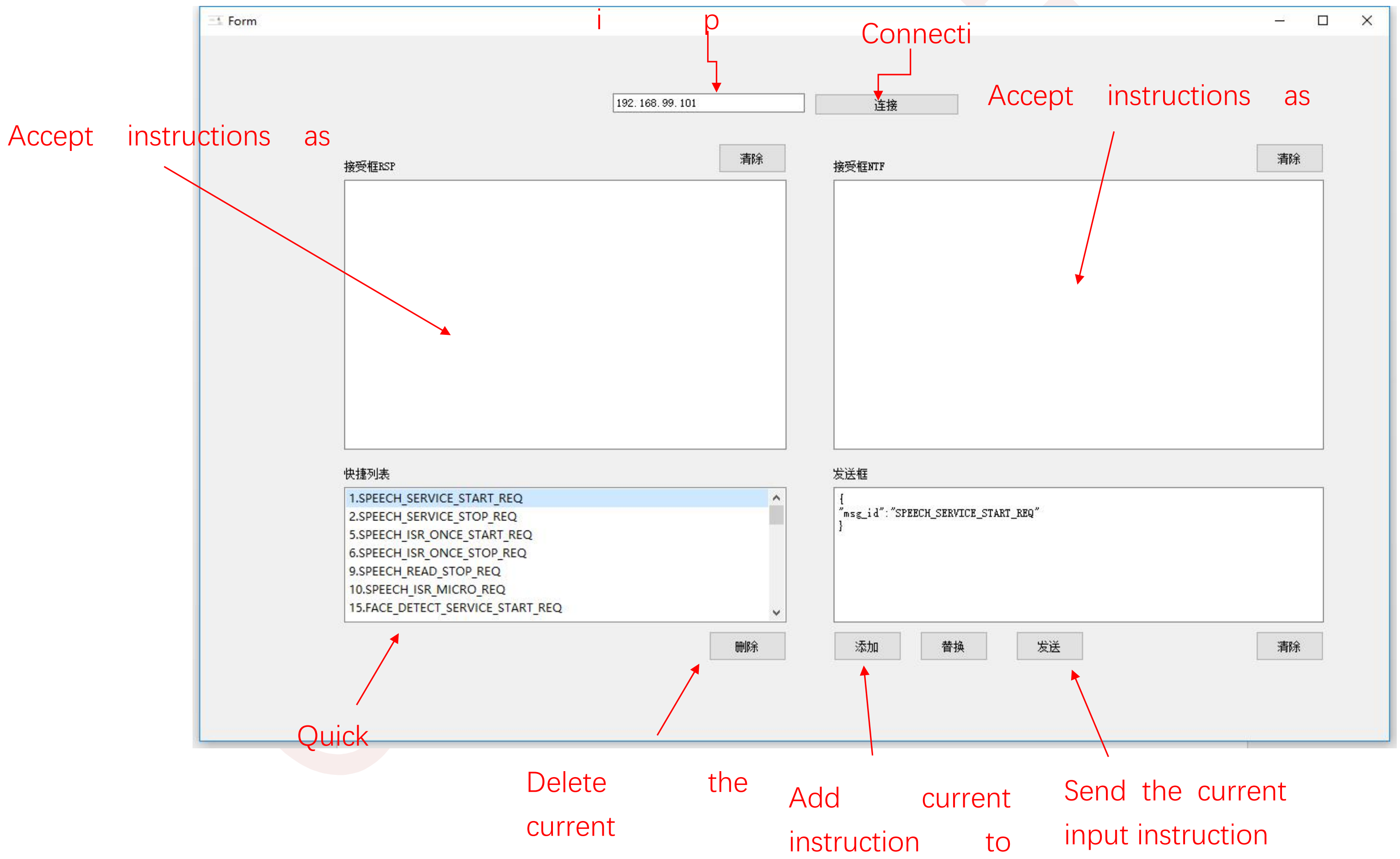
Operating environment :1. windows7,windows10 a notebook

2.A pangolin robot

3.sdk tool tool, download address:

This tool is used to simulate the sending and receiving of test message instructions, equivalent to the final software effect

2.3.1. Tool interface presentation



1.SPEECH_SERVICE_START_REQ
2.SPEECH_SERVICE_STOP_REQ
5.SPEECH_ISR_ONCE_START_REQ
6.SPEECH_ISR_ONCE_STOP_REQ
9.SPEECH_READ_STOP_REQ
10.SPEECH_ISR_MICRO_REQ
15.FACE_DETECT_SERVICE_START_REQ

```
{
"msg_id":"SPEECH_SERVICE_START_REQ"
}
```

i p

Connecti

Accept instructions as

Accept instructions as

Quick

Delete the current

Add current instruction to

Send the current input instruction

2.3.2. Description of use

1. Notebook connection robot wifi, preferably a 5 g band signal
2. Change the ip address to 192.168.99.101, click the Connect button
3. If the connection is successful, the word "connect" will change to "disconnect ", which is normal. If" not connected "is displayed, check that the notebook is connected to the robot router

4. After the connection is successful, according to the instructions in this document, you can fill in the corresponding instructions in the sending box, click the send button, if the instructions are correct, the robot will make the corresponding action, and respond to the content.

Chapter 3 Android SDK Integration Guide

Robot chest screen is 7.1.2 Android, so integrated Android SDK is recommended for development.
SDK encapsulates all the interfaces of Chapter 4 and can be called directly.

3.1. Introduction

3.2. How to integrate

Compatibility

Category	Compatibility Scope
System	Android 7.1.2 for robotic systems
Network	Support for mobile networks (including 2 G, etc.), WIFI and other network environments
Development environment	Development using the latest version Android Studio recommended

Download SDK
Click here to download SDK

Android Studio users
First step Import aar file

aar file added to the libs file

```
implementation (name :'csjsdk-beta', ext :'aar')
```

})

Step 2 introduces dependency libraries

```
implementation (name :'csjsdk-beta', ext :'aar')
```

Under the android{} structure of the app build.gradle file, add the following code

```
repositories {
flatDir {
dirs'libs'
    }
}
```

Configuration AndroidManifest for Step 3

```
<uses-permission android : name = android.permission.READ_EXTERNAL_STORAGE<uses-permission android: name =
<uses-permission android : name = android.permission.WRITE_EXTERNAL_STORAGE<uses-permission android: name =
<uses-permission android: name = android.permission.INTERNET<uses-permission android: name =
<uses-permission android : name = android.permission.RECORD_AUDIO<uses-permission android: name =
```

Step 4 Initializes the SDK in the Application

Please ensure that the device has a network for code authorization

Sign up on the pangolin developer platform

Request API Key and User Key after filling in information

To find API management in the right menu, API Key corresponding secret

Fill in when initializing, note that application permissions need to be given

```
CsjRobot.authentication (this, yourkey, yoursecret, new On AuthenticationListener (){
Override @
public void success(){}
Log.d (" TAG "," Authorized Success! ");
    }

Override @
public void error(){}
Log.d (" TAG "," Authorization failed! ");
    }
});
```

3.3. Interface and call procedure

Android-Sample details

Chapter 4 Interface details

4.1. Voice Services

4.1.1. Turn on Voice Services

Interface description

By calling this interface to open the voice service, the default will open this service.

Request data sample

```
{
"msg_id": "SPEECH_SERVICE_START_REQ"
}
```

Returns a result sample

Asynchronous process, return NTF notification after success.

```
{
"msg_id": "SPEECH_INIT_NTF"
}
```

4.1.2. Shut down voice services

Interface description

By calling this connection to turn off the voice service, the microphone will be released and the hardware release will be delayed.

Request data sample

```
{
    "msg_id"： "SPEECH_SERVICE_STOP_REQ"
}
```

Returns a result sample

```
{
    "error_code"： 0,
    "msg_id"： "SPEECH_SERVICE_STOP_RSP"
}
```

4.1.3. Multiple speech recognition

Interface description

By calling this interface to open continuous speech recognition, the robot will continue to recognize speech and push the results of each recognition. As long as this interface is turned on, the user can speak to the robot.Turn on will monopolize the microphone.

Request data sample

```
{
  "msg_id"： "SPEECH_ISR_START_REQ"
}
```

Returns a result sample

```
{
  "msg_id"： "SPEECH_ISR_START_RSP"
  "error_code"： 0
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
error_code	Error code	Int	0 Success

4.1.4. Turn off multiple speech recognition

Interface description

Shut down continuous speech recognition by calling this interface

Request data sample

```
{
    "msg_id"： "SPEECH_ISR_STOP_REQ"
}
```

Returns a result sample

```
{
    "error_code"： 0,
    "msg_id"： "SPEECH_ISR_STOP_RSP"
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
error_code	Error code	Int	0 Success

4.1.5. Speech Recognition Report

Interface description

Push speech recognition results from the bottom layer to the upper layer

Request data sample

Returns a result sample

```
{
  "msg_id"： "SPEECH_ISR_ONLY_RESULT_NTF",
  "text"： " recognize what the audience says"
```

```
    }
```

4.1.6. Speech and Semantic Recognition Results Report

Interface description

Push speech recognition and semantic results to the upper level

Request data sample

Returns a result sample

```
    {
      "msg_id"： "SPEECH_ISR_LAST_RESULT_NTF",
      "result"： {
      "data"： {
      "graphic"： "SN_NOT_FOUND"
                 },
      "error_code"： 10119,
      "How text"： " the weather today? "
          }
    }
```

Results Parameter Description

Parameters	Note	Type	Remarks
state	Language type	int	0 Leisure 1 Professional Library
data	Source format		

Specific examples:

Small talk returns:

```
{
  "msg_id"： "SPEECH_ISR_LAST_RESULT_NTF",
 "result"： {
"data"： {
 "actionList"： [
                ],
 "answer"：  " don't know, but every day seems to be busy ",
 "say"：  " don't know, but every day seems to be busy ",
 "serviceId"：  "other",
 "type"：  "chat"
          },
 "error_code"：  0,
 "text"：  " what do you do? "
     }
}
```

Text Answer Return:

```
{
 "msg_id"：  "SPEECH_ISR_LAST_RESULT_NTF",
 "result"：  {
 "data"：  {
 "actionList"：  [
                ],
 "answer"：  " test questions ",
 "graphic"：  "{\" type\"：  \"1\",\" answer\"：  \" test questions \\",
 "say"：  " test questions ",
 "type"：  "satisfy"
          },
 "error_code"：  0,
 "text"：  " test questions. "
     }
}
```

Image Return Example:

```
{
"msg_id": "SPEECH_ISR_LAST_RESULT_NTF",
"result": {
"data": {
"actionList": [],
"answer": " look at the picture, Hua looks at what and ",
"graphic": "{\" type\": \"2\",\" answer\": \" look at the picture, what do
you see and \\",\\" imgFile\": [{\" url\": \"
https://csjbot-test.su.bcebos.com/eBpn2G8jRejPm47Bnc4GwrTiG76GNzSmJmSn7
CtN.jpg\"}}",
"say": " look at the picture, Hua looks at what and ",
"type": "satisfy"
        },
"error_code": 0,
"How did text": " man come from? "
    }
}
```

Video return:

```
{
"msg_id": "SPEECH_ISR_LAST_RESULT_NTF",
"result": {
"data": {
"actionList": [],
"answer": " watch the video below ",
"graphic": "{\" type\": \"4\", "answer\": \" please watch the following
video\\", "videoFile\": [{\"url\":
\"https://csjbot-test.su.bcebos.com/DAf5k6SCZT22pC48NcGAtXpNKPTxtXThSEs
Hza6c.mp4\", \\" name\": \" Jack Junyi-take me to the top of the mountain
live version-Guoyu 1080 P( free). mp4\"}]}",
"say": " watch the video below ",
"type": "satisfy"
        },
"error_code": 0,
"text": " video. "
    }
}
```

The hyperlink returns:

```
{
"msg_id": "SPEECH_ISR_LAST_RESULT_NTF",
"result": {
"data": {
"actionList": [],
"answer": " enjoy the video below ",
"graphic": "{\" type\": \"10\",\" answer\": \" enjoy the following video ,"
link\": \"https://v.qq.com/x/page/n09202aiy6k_0.html\"}",
"say": " enjoy the video below ",
"type": "satisfy"
       },
"error_code": 0,
"text": " hyperlinks. "
   }
}
```

4.1.7. Voice wake-up call

Interface description

When the voice function is awakened (voice wake up), this message is notified by itself.

Request data sample

Request parameter description

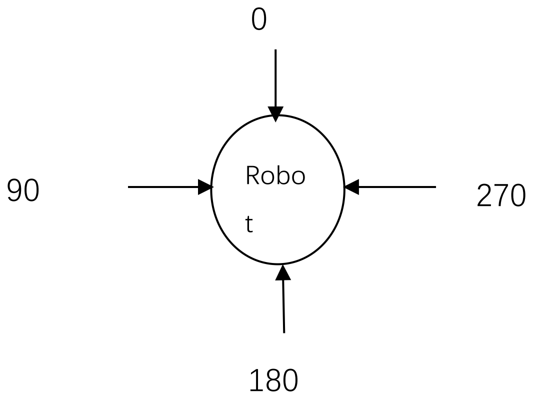
No

Returns a result sample

```
{
"angle": 34,
"error_code": 0,
"msg_id": "SPEECH_ISR_WAKEUP_NTF",
"wakeType": 0
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
wakeType	Wake-up mode	int	0:　Voice Wake
a ngle	Awakening angle	i nt	For angle values, please refer to the following figure (six wheat support)
error_code	Current status	int	0 success Other error codes



Note: Head facing direction above, arrow pointing to arousal angle

4.1.8. Voice Abnormal Notification

Interface description

When the voice function is abnormal, the message is notified by itself.

Request data sample

Request parameter description

No

Returns a result sample

```
{
"msg_id": "SPEECH_ISR_ERROR_NTF",
"failed desc": " engine initialization ",
"error_code": -1
}
```

error_code	desc	Error code interpretation
-1	Engine initialization failure	
-2	Single Word Hit.	A single word is identified and filtered out
-3	Content empty	A sentence returns an empty value depending on whether the business is handled
Other	See linkshttps://www.xfyun.cn/document/error-code	

4.1.9. Voice broadcast

Interface description

Can use this interface, let robot voice broadcast, Text to Speech

Here's how this technology uses Google TTS, non-genitor, supports Chinese, as follows

Map

Request data sample

```
{
  "Hello content"： "",

  "utterance_id"： "1595561389990",

  "msg_id"： "SPEECH_TTS_REQ"

}
```

Request parameter description

Parameters	Type	Note
content	String	If you need to speak, support Chinese
utterance_id	String	There will be id, in the NTF to identify the passage

Returns a result sample

```
{

  "Hello content"： "",

  "utterance_id"： "1595561389990",

      "msg_id"： "SPEECH_TTS_R SP"

}
```

where the utterance_id is incoming id, if not incoming or empty, a timestamp is used to mark this TTS as a session for NTF parsing

```
    {
        "error_code"： 0,
        "msg_id"： "SPEECH_TTS_START_SPEAK_NTF",
        "utterance_id"： "1595561389990"
    }

    {
        "error_code"： 0,
        "msg_id"： "SPEECH_TTS_SPEAK_COMPLETED_NTF",
        "utterance_id"： "1595561389990"
    }
```

SPEECH_TTS_START_SPEAK_NTF started talking
SPEECH_TTS_SPEAK_COMPLETED_NTF for the end of the speech, id for the same TTS

4.2. Chassis and navigation

4.2.1. Gets the current location

Interface description

Get the coordinates of the robot's current world coordinate system

Request data sample

```
    {
        "msg_id"： "NAVI_GET_CURPOS_REQ"
    }
```

Request parameter description

No

Returns a result sample

```
    {
            "error_code"： 0,
        "msg_id"： "NAVI_GET_CURPOS_RSP",
        "rotation"： 0,
        "x"： 0,
        "y"： 0,
        "z"： 0
    }
```

Parameters	Type	Note
x	double	Coordinate x axis
y	double	Coordinate y axis
z	double	Coordinates z axes generally 0
rotation	double	Turn round

4.2.2. Chassis movement instructions

Interface description

Manually control robot movement, keep at least 300 ms interval for continuous transmission

Request data sample

```
    {
        "msg_id"： "NAVI_ROBOT_MOVE_REQ",
        "direction"： 0
    }
```

Request parameter description

Parameters	Note	Remarks
direction	Direction	Before 0 After 1 2 Left 3 Right

Returns a result sample

```
    {
        "msg_id"： "NAVI_ROBOT_MOVE_RSP"
        "error_code"： 0
```

```
}
```

Results Parameter Description

Parameters	Note	Remarks
error_code	Current status	0 success

4.2.3. Specific point navigation

Interface description

Plan and march to the destination path and push the message of the point-specific navigation notification upon arrival

Request data sample

```
{
    "msg_id"： "NAVI_ROBOT_MOVE_TO_REQ",
    "pos"： {
        "x"： 2,
        "y"： 1,
        "z"： 0,
        "rotation"： 30
    }
}
```

Request parameter description

Parameters	Note	Type	Remarks
x	x coordinates	double	
y	y coordinates	double	
z	z coordinates	double	Default 0
rotation	Angle	double	Angle

Returns a result sample

```
{
    "msg_id"： "NAVI_ROBOT_MOVE_TO_RSP ",
    "error_code"： 0
}
```

Results Parameter Description

Parameters	Note	Remarks
error_code	Current status	0 success

4.2.4. Specific point navigation notifications

Interface description

When the robot calls 2.3.3 point-specific navigation, this message must be pushed unless 2.3.5 is called to cancel the movement.

Request data sample

Request parameter description

Returns a result sample

```
{
    "msg_id"：  "NAVI_ROBOT_MOVE_TO_NTF",
    "error_code"：  0
}
```

Results Parameter Description

Parameters	Note	Remarks
error_code	Current status	0 Success arrives Target 20004 not reached 20007 Robot watched by many people

4.2.5. Specific point navigation cancellation

Interface description

During navigation (2.3.3), remove the robot's march

Request data sample

```
{
    "msg_id"：  "NAVI_ROBOT_CANCEL_REQ"
}
```

Request parameter description

Returns a result sample

```
{
    "msg_id"：  "NAVI_ROBOT_CANCEL_RSP",
    "error_code"：  0
}
```

Results Parameter Description

Parameters	Note	Remarks
error_code	Current status	0 success Other error codes

4.2.6. Turn to a particular angle

Interface description

Take the robot world coordinate system as the reference coordinate to rotate the angle. (i.e. angle coordinates in sweep software)

Request data sample

```
{
    "msg_id"：  "NAVI_GO_ROTATION_TO_REQ",
    "rotation"：  0
}
```

Request parameter description

rotation if greater than 360, take rotation -n*360, for example rotation =380, the same effect as rotation =20

Returns a result sample

```
{
    "msg_id": "NAVI_GO_ROTATION_TO _RSP",
    "error_code": 0
}
```

Map here

4.2.7. Step angle

Interface description

Take the current position of the robot as the reference coordinate to rotate the angle.

Request data sample

```
{
      "msg_id": "NAVI_GO_ROTATION_REQ",
      "rotation": 0
}
```

Request parameter description

Parameters	Note	Remarks
rotation	Turn angle	Can be greater than 360 Positive turn left Negative turn right 20 to turn left 20 degrees -20 degrees to right

Returns a result sample

```
{
    "msg_id": "NAVI_GO_ROTATION _RSP",
    "error_code": 0
}
```

4.2.8. Back charging point

Interface description

Back to the charging pile, do not call this interface if there is no charging pile. A RSP message is pushed when the robot returns to the pile.

Request data sample

```
  {
     "msg_id"： "NAVI_GO_HOME_REQ"
  }
```

Request parameter description

No

Returns a result sample

```
  {
    "msg_id"： "NAVI_GO_HOME _RSP",
    "error_code"： 0
  }
```

4.2.9. Set current moving speed

Interface description

Speed is not saved and needs to be set every time the map is restored

Attention: Risk of collision over 0.5 m/s

Request data sample

```
{
       "msg_id"： "NAVI_ROBOT_SET_SPEED_REQ",
       "speed"： 0.5
}
```

Request parameter description

Parameters	Note	Type	Remarks
speed	Error code	double	0.1-0.7, default 0.5

Returns a result sample

```
{
  "msg_id"： "NAVI_ROBOT_SET_SPEED_RSP",
  "error_code"： 0
}
```

4.2.10. Gets the current moving speed

Interface description

Request data sample

```
{
   "msg_id"： "NAVI_ROBOT_GET_SPEED_REQ"
}
```

Request parameter description

Returns a result sample

```
{
  "msg_id"： "NAVI_ROBOT_GET_SPEED_RSP",
   speed"： 0.50,
   "error_code"： 0
}
```

Parameters	Note	Type	Remarks
speed	Speed	double	0.1-0.7, default 0.5 Go to two decimal places, if other platforms have long data, please keep it yourself

4.2.11. Storage maps

Interface description

Save the map in the scanning software to the robot body.

Request data sample

```
{
    "msg_id": "NAVI_GET_MAP_REQ",
    "name": "map"
}
```

Request parameter description

Parameters	Note	Type	Remarks
name	Name of map to be saved	string	

Returns a result sample

```
{
  "msg_id": "NAVI_GET_MAP_RSP ",
  "error_code": 0
}
```

4.2.12. Load map

Interface description

Load the map saved in the robot body, this interface is a synchronous interface, can not be sent synchronously in a short time

Request data sample

```
{
      "msg_id": "NAVI_SET_MAP_REQ",
      "name": "map"
}
```

Request parameter description

Parameters	Note	Type	Remarks
name	Map name to load	string	

Returns a result sample

```
{
   "msg_id"：  "NAVI_SET_MAP_RSP ",
   "error_code"：  0
}
```

Parameters	Note	Type	Remarks
error_code	Wrong	Int	0： normal -1: The designated map does not exist 2: The map is too small 3: Navigation not connected -4: Damage to map documents

4.2.13. Map Initialization Status Query

Interface description

Used to determine the state of the loaded map.

Request data sample

```
{
    "msg_id"：  "NAVI_GET_MAPSTATUS_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
   "msg_id"：  "NAVI_GET_MAPSTATUS_RSP",
   state"：  true,
   "error_code"：  0
}
```

Parameters	Note	Type	Remarks
state	Determines whether map	bool	true： map restored false ： map not

	information has been initialized		restored

4.2.14. Navigation status query

Interface description

Query the current navigation status.

Request data sample

```
{
   "msg_id"： "NAVI_GET_STATUS_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
  "msg_id"： "NAVI_GET_STATUS_RSP",
  stat e "： 0,
  "error_code"： 0
}
```

Return parameter description

Parameters	Note	Type	Remarks
status	Navigation state	int	0： free 1： is navigating

4.2.15. Navigation mode settings

Interface description

Set navigation mode.

Obstacle-avoidance mode: obstacles are bypassed by default

Fixed obstacle mode: walk along the virtual track, encounter obstacles will stop until the obstacles leave.

Request data sample

```
{
   "msg_id"：  "NAVI_S ET_MODE_REQ".
  mode"：  0
}
```

Request parameter description

No

Returns a result sample

```
{
  "msg_id"：  "NAVI_S ET_MODE_RSP",
  "error_code"：  0
}
```

Return parameter description

Parameters	Note	Type	Remarks
mode	Navigation state	int	0 ： obstacle avoidance mode, default 1： obstacle fixing mode (walking along virtual track)

4.3. Face Recognition

4.3.1. Open Video Stream

Interface description

Open video stream transmission interface, this interface sent socket connection, port number is 60003. demo. to use

As long as this interface is opened, the server will continue to transfer video content to the client, the data is jpg picture stream:

Each frame has a header of xff ,0xfe ,0xfd ,0xfc ,0xfb ,0xfa ,0xd8 0

At the end of each frame xff ,0xfe ,0xfd ,0xfc ,0xfb ,0xfa ,0xd9 0

Request data sample

```
{
    "msg_id"： "FACE_DETECT_OPEN_VIDEO_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
    "error_code"： 0,
    "msg_id"： "FACE_DETECT_OPEN_VIDEO_RSP"
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
error_code	Current status	int	0 success

4.3.2. Turn off video stream

Interface description

Close the video stream, port 60003 no longer sends the jpg stream

Request data sample

```
{
    "msg_id"： "FACE_DETECT_CLOSE_VIDEO_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
    "error_code"： 0,
    "msg_id"： "FACE_DETECT_CLOSE_VIDEO_RSP"
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
error_code	Current status	int	0 success

4.3.3. Camera camera

Interface description

Click on the photo or call the following interface to complete the photo action, please use the video stream of the picture to screenshot.

Request data sample

```
{
   "msg_id"：  "FACE_SNAPSHOT_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
      "error_code"：  0,
      "face_position"：  0,
      "msg_id"：  "FACE_SNAPSHOT_RESULT_RSP"
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
error_code	Error code	int	0 means face Other No Face

4.3.4. Face Registration

Interface description

By calling this interface (maintaining the current photo face name), the face in the current photo is registered to the database.You must first perform 2.2.5 camera photography and return the face in the photo before using this feature.

Request data sample

```
{
    msg_id "：  " FACE_SAVE_REQ",
    "name"：  " Zhang San"
}
```

Request parameter description

Parameters	Note	Type	Remarks
name	Registered Name	string	

Returns a result sample

```
{
    "msg_id"：  "FACE_SAVE_RSP",
    person_id "：  " personx20170107161021mRJOVw ",
   error_code"：  0
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
person_id	Face Unique Identifier	string	
error_code	Current status	int	0 success

4.3.5. Deletion of Face Information

Interface description

By calling this interface (self-maintenance needs to delete photo face ID) to delete a face information in the database, can also be deleted in the pangolin background account.

Request data sample

```
 {
    "msg_id"：  "FACE_DATA_DEL_REQ"
    "person_id"：  "123456"
  }
```

Request parameter description

Parameters	Note	Type	Remarks
face_id	ID face	string	

Returns a result sample

```
  {
    "msg_id"：  "FACE_DATA_DEL_RSP"
    "error_code"：  0
  }
```

Results Parameter Description

Parameters	Note	Type	Remarks
error_code	Current status	int	0 success

4.3.6. Face Induction Information Report

Interface description

When someone faces near the robot camera, the message is automatically pushed when the state changes. When the value of the person changes from true to false or from false to true, it is pushed automatically, but not at other times.)

Request data sample

Request parameter description

Returns a result sample

```
    {
        "msg_id"： "FACE_DETECT_PERSON_NEAR_NTF",
        "person"： true
    }
```

Results Parameter Description

Parameters	Note	Remarks
person	Is there a face nearby	

4.3.7. Face Recognition Information Report

Interface description

Report face information recognized by current robot camera

Request data sample

Request parameter description

Returns a result sample

```
{
      "msg_id"： "FACE_DETECT_FACE_LIST_NTF",
      "face_num"： 2,
      "face_list"： [
          {
                "face_detect"： {
                      "age"： 20,
                      "gender"： 2,
                      "smile"： 34
                },
                "face_recg"： {
                      "confidence"： 94,
                      "name"： " Zhang San"
                }
```

```
            },
            {
                    "face_detect"： {
                          "age"： 28,
                          "gender"： 2,
                          "smile"： 45
                    },
                    "face_recg"： {
                          "confidence"： 887,
                          "name"： " Lee Si ",
                          "person_id"： "personx20170107161021mRJOVw"
                    }
            }
      ]
}
```

Results Parameter Description

4.3.8. Face database acquisition

Interface description

The face information of the corresponding msg_id can be obtained from the database by calling this interface (the corresponding face msg_id is maintained by itself.

Request data sample

```
   {
      "msg_id"： "FACE_DATABASE_REQ"
   }
```

Request parameter description

Returns a result sample

```
{
        "msg_id"： "FACE_DATABASE_RSP",
        "data_list"： [
            {`
```

```
                "id"： "asdw1",
                "name"： " Zhang San"
            },
            {
                "id"： "gfhdf2",
                "name"： " Li Si"
            }
        ],
         "list_num"： 2,
         "all_num"： 2,
         "page_num"： 0
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
list_num	Number of faces in the list		Up to 20 face messages
all_num	Total number of faces	int	
page_num	Number of pages currently transmitting face list information	int	Face information is transmitted multiple times (when face number exceeds 20) count from 0
id	id face	string	

4.4. Upper body control panel (welcome)

4.4.1. Robot Body Operation Command

Interface description

Control robot upper body

Request data sample

```
{
        "msg_id"："ROBOT_BODY_CTRL_CMD",
        "body_part"：3,
        "direction"：3,
        "angle"：20,
        "speed"：2
}
```

Request parameter description

Action	Signal	Range of values
body_part	Joint type	0：none ,1: head, 2：neck ,3: left shoulder, 4：right shoulder;
direction	Reverse turn	0：no action ,1: positive turn ,2: reverse;

angle	Motion angle	Head :-15-20 degrees, default 0 degrees (horizontal);
		Neck :-45--45 degrees, default 0 degrees (middle);
		Left shoulder :-30-120 degrees, default 0 degrees (vertical);
		Right shoulder :-30-120 degrees, default 0 degrees (vertical);

speed	Speed	1:1(slowest);2:2;
		3:3;4:4;5:5(fastest);

Returns a result sample

Results Parameter Description

4.5. Upper body control panel (Lesser Snow)

4.5.1. Robot Body Operation Command

Interface description

Control robot upper body

Request data sample

```
{
   msg_id ":  " ROBOT_BODY_CTRL_CMD",
   body_part":  1,
   action":  2
}
```

Request parameter description

Action	body_part	action (swing times ,<=20)
Right arm swing	1	1
Left arm swing	2	1
Arms swing	3	1

Returns a result sample

Results Parameter Description

4.5.2. Robot left and right swing open

4.5.3. Left and right swing stop

4.6. Configuration Information

4.6.1. Obtain version number

Interface description

Access to the underlying service version number

Request data sample

```
{
    msg_id ":  " GET_VERSION_REQ"
```

```
        }
```

Request parameter description

No

Returns a result sample

```
        {
              msg_id ":  " GET_VERSION_RSP"
              version ":  " V1.0.0"
        }
```

Results Parameter Description

4.6.2. Acquisition SN number

Interface description

Get each hardware sn number

Request data sample

```
        {
                                        msg_id          "          :          "          GET_SN_REQ
" """""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
 .
""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""

        }
```

Request parameter description

Returns a result sample

```
        {
        msg_id ":  " GET_SN_RSP",
        sn ":  "",
        error_code":  0
        }
```

Results Parameter Description

Parameters	Type	Note
error_code	String	0 success 1 : didn't find sn host computer 2: robots are not stored

sn	String	sn

4.6.3. Acquisition of robot type

Interface description

Update the internal robot type and restart the robot after updating.

Request data sample

```
{
msg_id "： " GET_ROBOT_TYPE_REQ "
}
```

Request parameter description

Returns a result sample

```
{
    msg_id "： " GET_ROBOT_TYPE_RSP",
type "： " alice",
  error_code"： 0
}
```

Results Parameter Description

Parameters	**Note**	**Type**
type	a lice ： welcome robot snow： Snow amy： meals alicebig： welcome screen robot amybig ： large screen feeding robot amyrail： rail feeding robot foodcar： Dining Car Robot	S tring

4.6.4. Get the robot hardware version information

Interface description

Request data sample

```
{
msg_id "：  " G ET_HARDWARE_INFO_REQ"
}
```

Request parameter description

Returns a result sample

```
{
msg_id "：  " G ET_HARDWARE_INFO_RSP",
    version "：  " HARDWARE _V1.0.0"
}
```

Results Parameter Description

Parameters	Note	Type
version	Hardware Version Number	S tring

Version Description

Version Number	Note
HARDWARE_V1.0.0	2016 robots
HARDWARE_V 2.0.0	The 2017 new robot, with self-checking, adds hardware support to the feeding robot
HARDWARE_V 3.0.0	The voice board was changed to improve the gain

4.7. Alarm related

4.7.1. Self-inspection instructions

Interface description

Acquisition of robot hardware self-checking information

Request data sample

```
{
          "msg_id"："WARNING_CHECK_SELF_REQ"
}
```

Request parameter description

Returns a result sample

```
{
     "msg_id"："WARNING_CHECK_SELF_RSP",
      list"：[
           {
                type "： " mainboard",
                model "： " sw2315",
                serialnumber "："201800021001",
                state "： " ok",
                firmwareversion "："510"
           },
           {
                type "： " navimodule",
                model "： " sw2315",
                serialnumber "："201800021001",
                state "： " ok",
                firmwareversion "："510"
           }
      ]
     "error_code"：0
}
```

Results Parameter Description

Parameters	Note	Type	Is it necessary
t ype	Type of equipment	String	Yes
model	Specifications	String	No
serialnumber	S n No	String	No
state	Check item status OK/NG	String	Yes
f irmwareversion	Firm number	String	No

type Corresponding List

t type	Name of name
mainboard	Main control board
dcboard	DC motor drive board
steboard	Stepper motor drive board
powerboard	Power Management Board
micboard	5/6 MIC boards
ultsensor	Ultrasonic sensors
antsensor _front_left	Left before drop sensor
antsensor_front_middle	Drop sensor front
antsensor_front_right	Front right of anti-drop sensor
antsensor_behind_left	Left after drop sensor
antsensor_behind_middle	Back of drop sensor
antsensor_behind_right	Drop sensor right
tousensor_lefthead	Touch sensor left hand/head
tousensor_right	Touch sensor right hand
ioasensor_left	Plate Infrared Avoidance Sensor Left
i oasensor_right	Right of plate infrared avoidance sensor
pycsensor	Thermal Release Sensor
leftmotor	Left wheel motor
rightmotor	Right Wheel Motor
lefthandmotor	Left elbow motor (welcome)
righthandmotor	Right elbow motor (welcome)
nodmotor	Nod motor (welcome)
shakemotor	Shaking motor (welcome)
navmodule	Navigation module
camera	Camera

4.7.2. Equipped with robot sensor enable switch

Interface description

This interface is for debugging only. Once used, the robot sensor will not be controlled by default. To restore the original state, restart the robot.

Note: different types of robots do not necessarily have all sensors.

Request data sample

```
{
        "msg_id"： "SET_SONAR _SWITCH _REQ",
        "touch "： false,
        table_infrared"： false,
        pyroelectric"： false,
        ultrasonic"： false,
        fall"： false
}
```

Request parameter description

Parameters	Note	Type
touch	Touch sensor	b ool
table_infrared	Disc infrared sensor	b ool
pyroelectric	Thermal Release Sensor	b ool
ultrasonic	Ultrasonic sensors	b ool
fall	Anti-drop sensor	b ool

Returns a result sample

```
{
        "msg_id" ： "SET_SONAR _SWTICH _RSP"
        "error_code"： 0
}
```

Results Parameter Description

Parameters	Type	Note	Remarks
error_code	int	Error code	0 success

4.8. Other orders

4.8.1. Heartbeat request

Interface description

Request data sample

```
{
    "msg_id"： "HEART_BEAT_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
    "msg_id"： "HEART_BEAT_RSP"
}
```

Results Parameter Description

4.8.2. Robot Power Acquisition

Interface description

Request data sample

```
{
    "msg_id"： "ROBOT_GET_BATTERY_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
    "battery"： 61,
    "error_code"： 0,
    "msg_id"： "ROBOT_GET_BATTERY_RSP"
}
```

Results Parameter Description

Parameters	Note	Remarks
battery	Battery power	Percentage

4.8.3. Acquisition of robot emergency stop state

Interface description

Request data sample

```
{
    "msg_id"："GET_EMERGENCY_STATUS_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
    "status"：1,
    "error_code"：0,
    "msg_id"："GET_EMERGENCY_STATUS_R SP"
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
status	State of emergency	int	0： pressed 1 ： release of emergency

4.8.4. Robot emergency stop push

Interface description

This message is automatically pushed when the state of the robot stop button changes.

Request data sample

Request parameter description

No

Returns a result sample

```
{
    "status"： 1,
    "error_code"： 0,
    "msg_id"： "ROBOT_GET_EMERGENCY_NTF"
}
```

Results Parameter Description

Parameters	Note	Type	Remarks
status	State of emergency	int	0： pressed 1 ： release of emergency

4.8.5. Robot charging state acquisition

Interface description

Request data sample

```
{
    "msg_id"： "ROBOT_GET_CHARGE_REQ"
}
```

Request parameter description

No

Returns a result sample

```
{
    "msg_id"： "ROBOT_GET_CHARGE_RSP"
    "charge"： 0,
    error_code"： 0
}
```

Results Parameter Description

Parameters	Note	Remarks
charge	Battery charging status	0： don't charge 1： is charging

4.8.6. Robot charging state push

Interface description

This message is pushed when the charging state changes

Request data sample

Request parameter description

No

Returns a result sample

```
{
    "msg_id"： "ROBOT_CHARGE_STATE_NTF"
    "charge_state"： 0,
    error_code"： 0
}
```

Results Parameter Description

Parameters	Note	Remarks
charge_state	Battery charging status	0： don't charge 1： is charging 2： power is full (sent every 5 seconds at full power)

4.8.7. News broadcasts

Interface description

Broadcast messages to all people connected to the robot

Request data sample

```
{
```

```
"msg_id":  "ROBOT_DISPATCH_REQ",
 "msg_body":
     {
          msg_id ":  " id" of messages to be forwarded
          …….
     }
}
```

Request parameter description

Parameters	Note	Remarks
msg_body	Content to be forwarded	Complete the msg_id of the content to be forwarded

Returns a result sample

Results Parameter Description

Additional parameter description

2 special req for communication with host computer interface

Continue to guide:

```
{
  "msg_id":  "ROBOT_DISPATCH_REQ",
  msg_body":
      {
           msg_id ":  " NAVI_CONTINUE_GUIDE_REQ"
      }
}
```

Suspension of guidance:

```
{
  "msg_id":  "ROBOT_DISPATCH_REQ",
  msg_body":
      {
           msg_id ":  " NAVI_PAUSE_GUIDE _REQ"
      }
}
```

4.8.8. Motor Overload Enquiry Command

Interface description

Check the status of the robot motor overload

Request data sample

```
    {
        "msg_id"：  "GET_MOTOR_OVERLOAD_STATUS_REQ"
    }
```

Return data sample

```
    {
        "msg_id"：  "MOTOR_OVERLOAD_STATUS_RSP ",
        status "：  0
    }
```

Parameters	Note	Remarks
status	State	0： normal
		1. overload

4.8.9. Motor Overload Clearance Command

Interface description

Clear motor overload

Request data sample

```
    {
        "msg_id"：  "MOTOR_OVERLOAD_CLEAR_REQ"
    }
```

Return data sample

```
    {
          "error_code"：  0,
        "msg_id"：  "MOTOR_OVERLOAD_CLEAR_RSP"
    }
```

Parameters	Note	Remarks
error_code	State	0： clearance successfully
		1. clearance failed

4.8.10. Motor overload push

Interface description

Overload status returns error_code "：1

Returns a result sample

```
{
     "error_code"：1,
"    msg_id "：" MOTOR _OVERLOAD _NTF "
}
```

4.9. Expression settings

4.9.1. Set up facial expressions

Interface description

For Alice only

Request data sample

```
{
  "msg_id"："SET_ROBOT_EXPRESSION_REQ",
  "expression"：5003,
  "once"：1,
  "time"：0
}
```

Request parameter description

Parameters	**Note**	**Remarks**
expression	Expression selection, int	LCD face: 5000: Pleasure 5001: Sadness 5004: General 5005: Anger 5006: Lightning 5007: Sleepiness
once	Would you cut back normal expression once, int it	1： yes 0： No

time	Play by time, time to default cut back normal expression, int (Conflicting with once parameters, only one)	0：　not enabled Other: millisecond units

Returns a result sample

```
{
    "msg_id"： "SET_ROBOT_EXPRESSION_RSP",
    "error_code"： 0
}
```

Return parameter sample

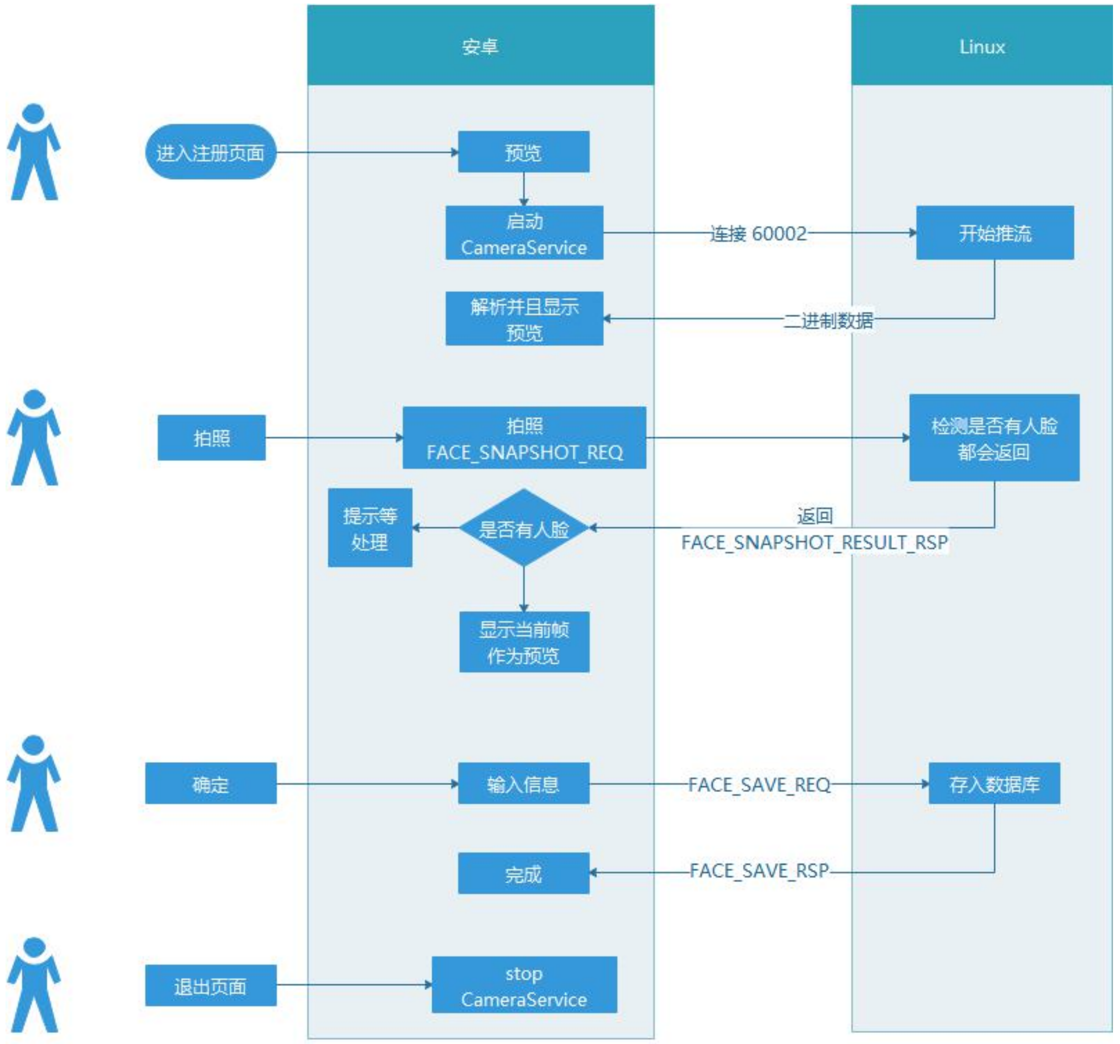
Parameters	Note	Remarks
error_code	Error code, int	0：　normal

Chapter 5 Best practices

5.1. Voice Response Process

1. multiple speech recognition
2. wait for 3.1.4 or 3.1.5 speech recognition content and answers
3. take the answer and call 3.1.8 tts text to voice (Android calls your own tts)
4. cycle 2,3 steps

5.2. Face Registration Process



5.3. Face Recognition Process

1. when someone approaches the camera, first push 3.2.6 face sensing information to report, then push 3.2.7 face recognition information to report, the field "person" is true" at this time

When 2. robot doesn't have anyone in sight for about five seconds, it pushes 3.2.8 face sensing information to report, but the field "person" is false" at this time

5.4. Navigation process

Preparation in advance:

1. scan the map.

2. call save map interface to save map.

The 3. uses the scanning map software to move the robot to the target position, then calls to obtain the target position coordinate point 3.3.1, records the x,y,rotation information to the upper computer.

When users interact:

After the 1. boot, first call a load map interface, waiting for the robot repositioning completed.

2. waiting for the movement instruction, using the previously recorded coordinate data, move to the target position 3.3.3.

3. wait for 3.3.4 specific navigation notifications to know that the target point has now been reached.

4. wait for instructions for the next moving position.Repeat steps 2 and 3

Chapter 6 Frequently Asked Questions