

5. Colour recognition

Quick use

1. DOGZILLA POWER UP

First of all, we switch on the switch power of the robot dog and start the robot dog



After startup, we can view the IP address on the robot dog's small screen.

2. Open shell to connect to DOGZILLA

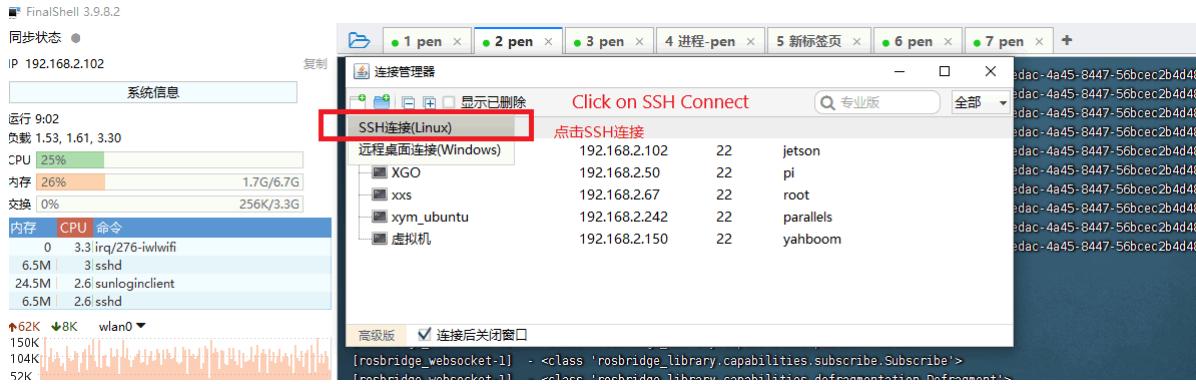
Then use the ssh terminal to connect to robot dog.

Note: At the time of writing this tutorial, the IP address used is 192.168.2.102 and the username is pi and the password is yahboom, so the actual IP address will prevail.

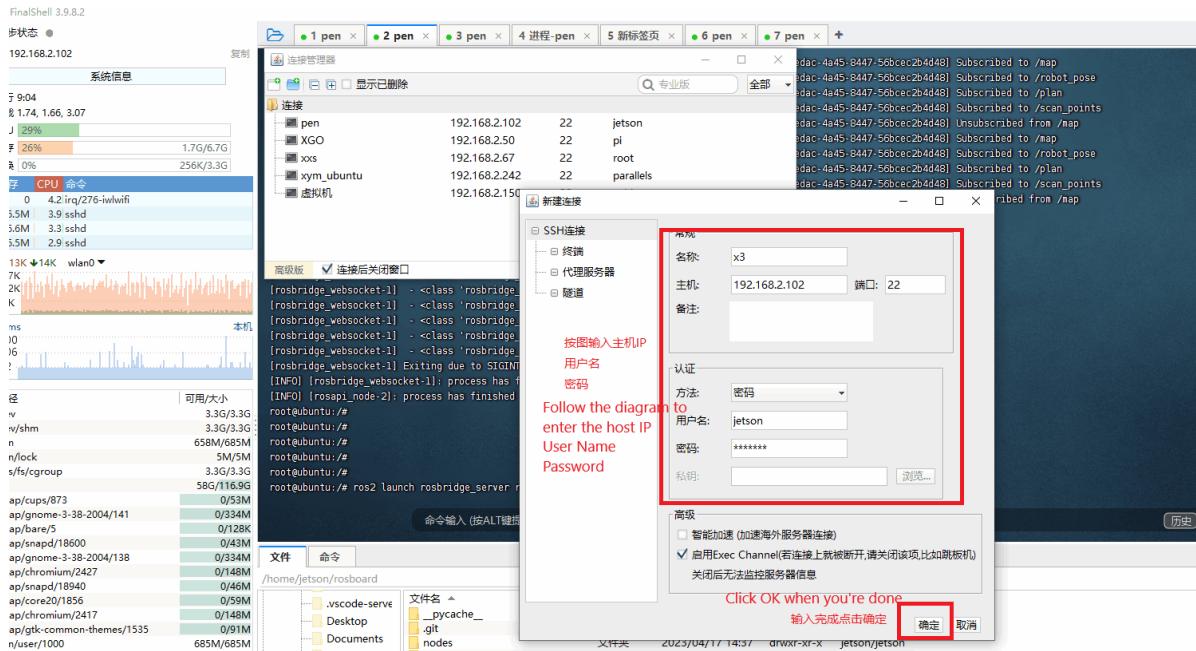
Open the shell utility, here I use FinalShell, enter the username, password, port, connection name and other information.

```
[INFO] [rosbridge_websocket-1] process has finished cleanly [pid 22592]
[INFO] [rosapi_node-2]: process has finished cleanly [pid 22594]
root@ubuntu:/#
```

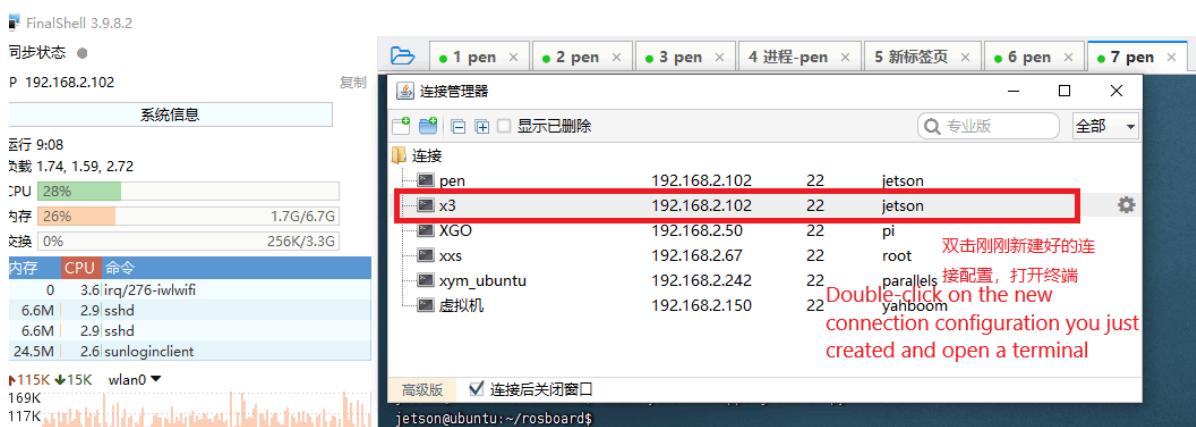
Select ssh connection to create a new ssh connection



Here username fill in pi, password fill in yahboom, ip address fill in the real robot dog's IP address.



Here select the new ssh connection you just created.



3. Starting the DOGZILLA chassis

Start the chassis task by entering the command in the terminal.

```
sudo systemctl restart YahboomStart.service
```

```
pi@yahboom:~$  
pi@yahboom:~$  
pi@yahboom:~$  
pi@yahboom:~$  
pi@yahboom:~$ sudo systemctl restart YahboomStart.service
```

4. Start the image publishing node

Enter the following command in the terminal

```
cd cartographer_ws2/
```

```
source install/setup.bash
```

```
pi@yahboom:~$ cd cartographer_ws2/  
pi@yahboom:~/cartographer_ws2$  
pi@yahboom:~/cartographer_ws2$  
pi@yahboom:~/cartographer_ws2$ source install/setup.bash  
pi@yahboom:~/cartographer_ws2$
```

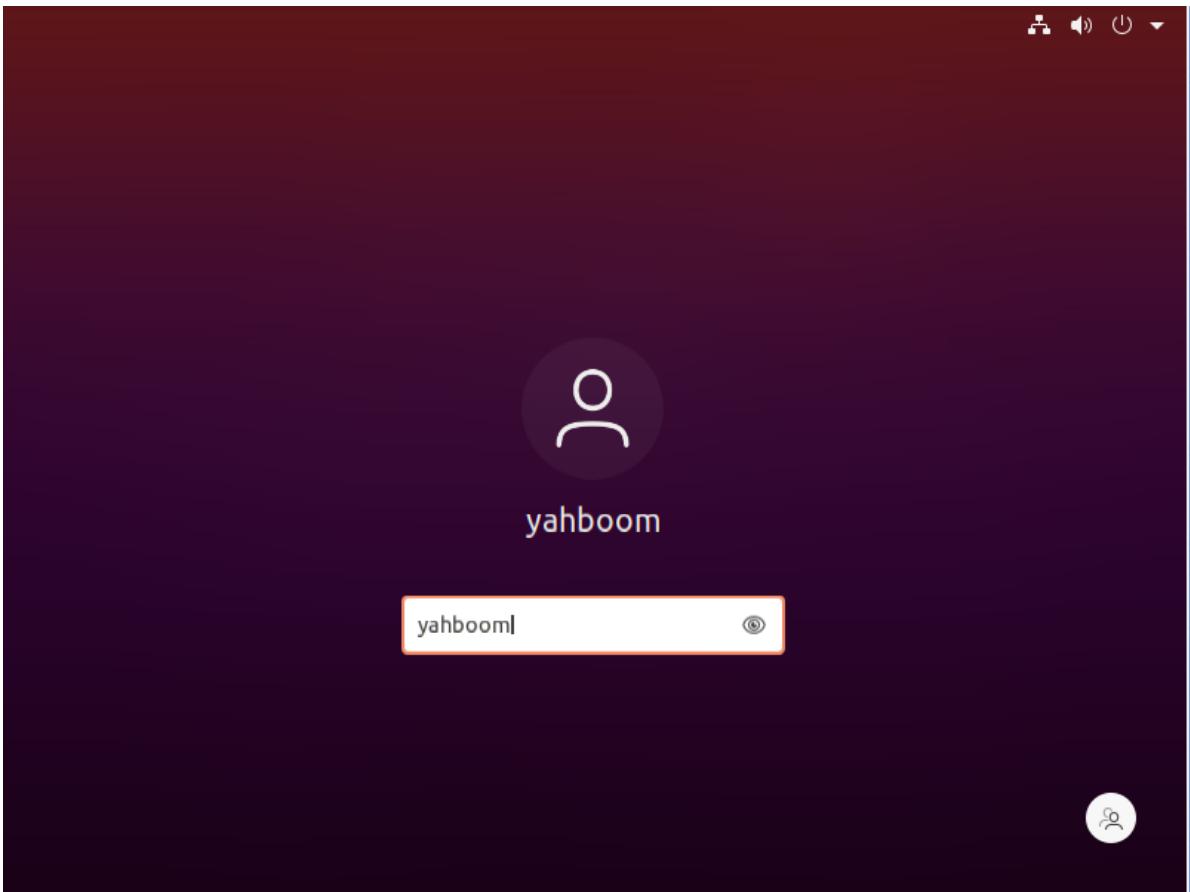
Then enter the following command

```
ros2 run yahboom_image_publisher_c yahboom_image_publish_c
```

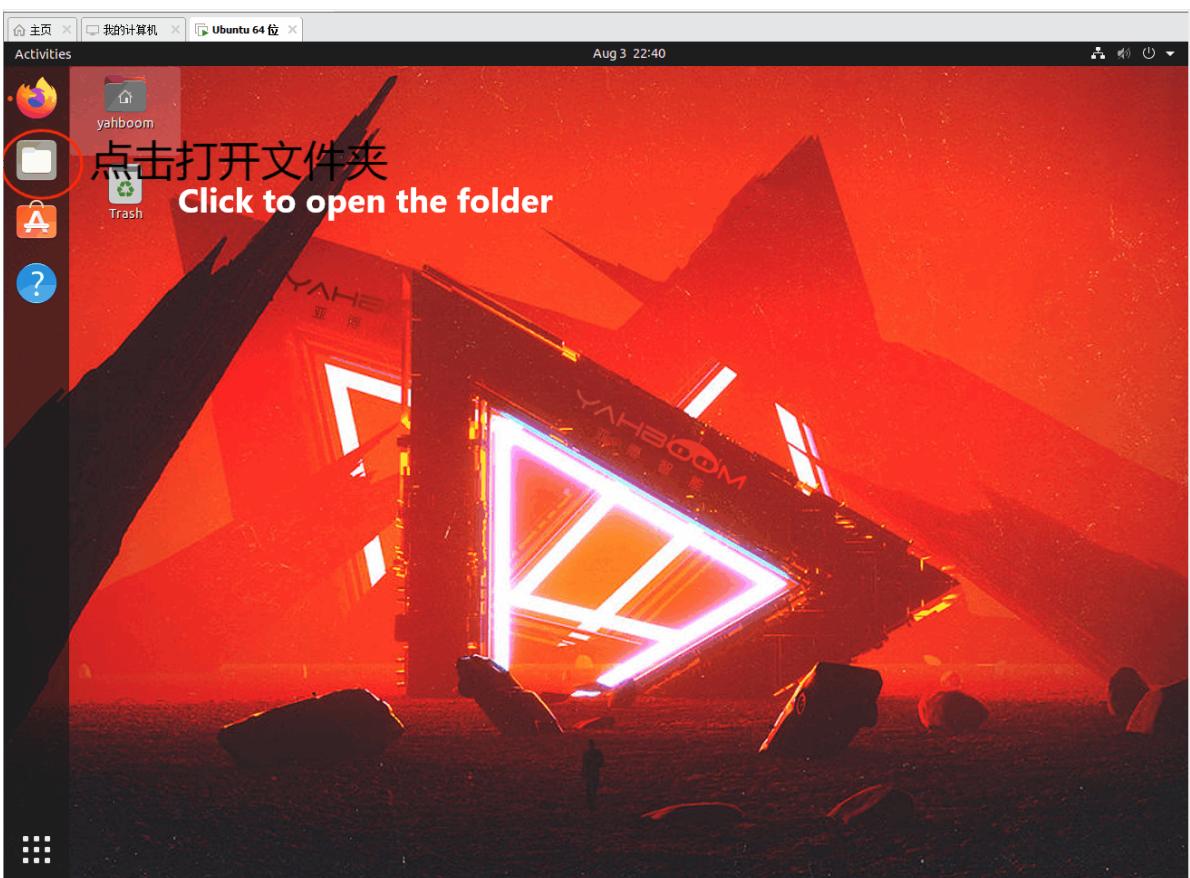
```
pi@yahboom:~/cartographer_ws2$  
pi@yahboom:~/cartographer_ws2$ source install/setup.bash  
pi@yahboom:~/cartographer_ws2$  
pi@yahboom:~/cartographer_ws2$  
pi@yahboom:~/cartographer_ws2$  
pi@yahboom:~/cartographer_ws2$ ros2 run yahboom_image_publisher_c yahboom_image_publish_c  
[ WARN:0] global ..../modules/videoio/src/cap_gstreamer.cpp (1758) handleMessage OpenCV | GStreamer warning: Embedded video playback halted; module source reported: Could not read from resource.  
[ WARN:0] global ..../modules/videoio/src/cap_gstreamer.cpp (888) open OpenCV | oStreamer warning: unable to start pipeline  
[ WARN:0] global ..../modules/videoio/src/cap_gstreamer.cpp (480) isPipelinePlaying OpenCV | GStreamer warning: GStreamer: pipeline have not been created
```

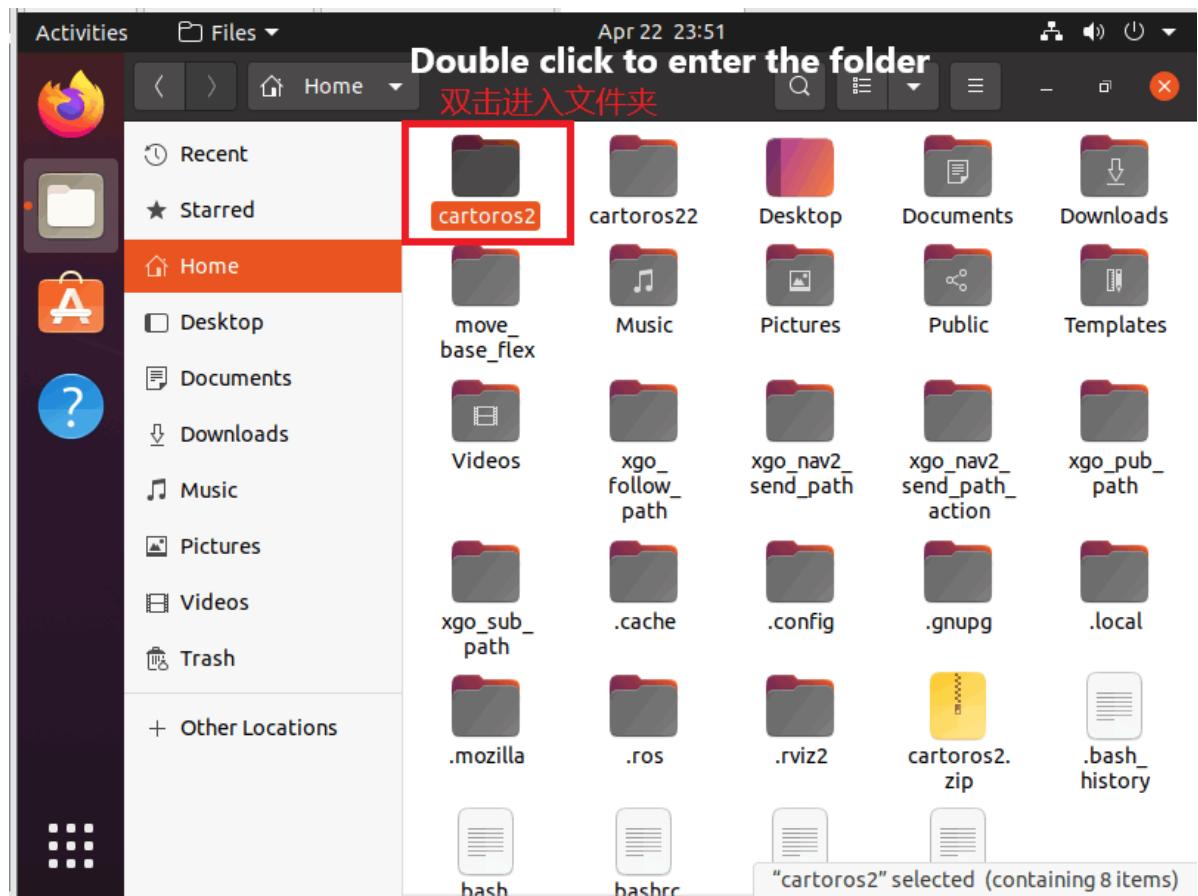
5. Setting the recognition colours through the web interface

Open the virtual machine and enter the username yahboom, password yahboom.

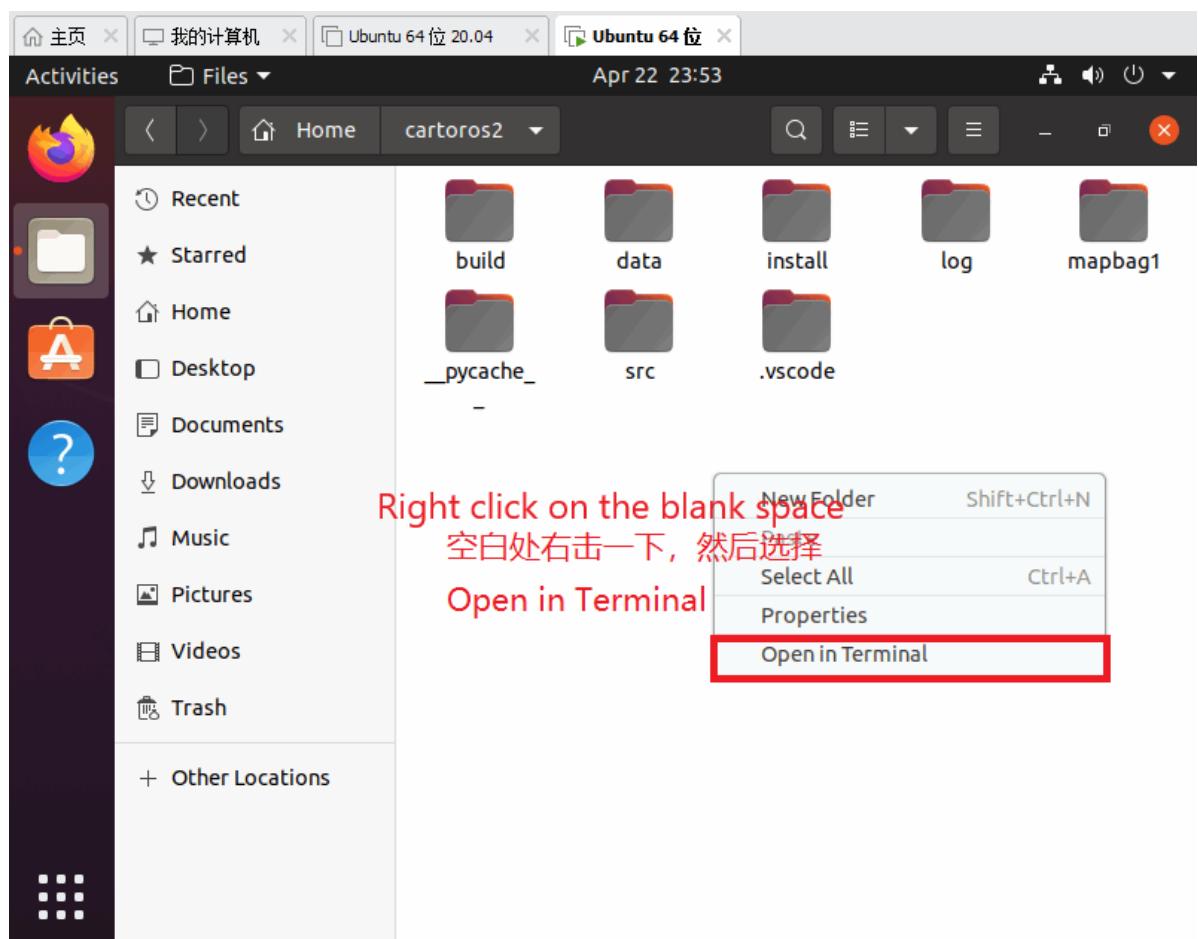


Click on the folder to open the cartoros2 folder.



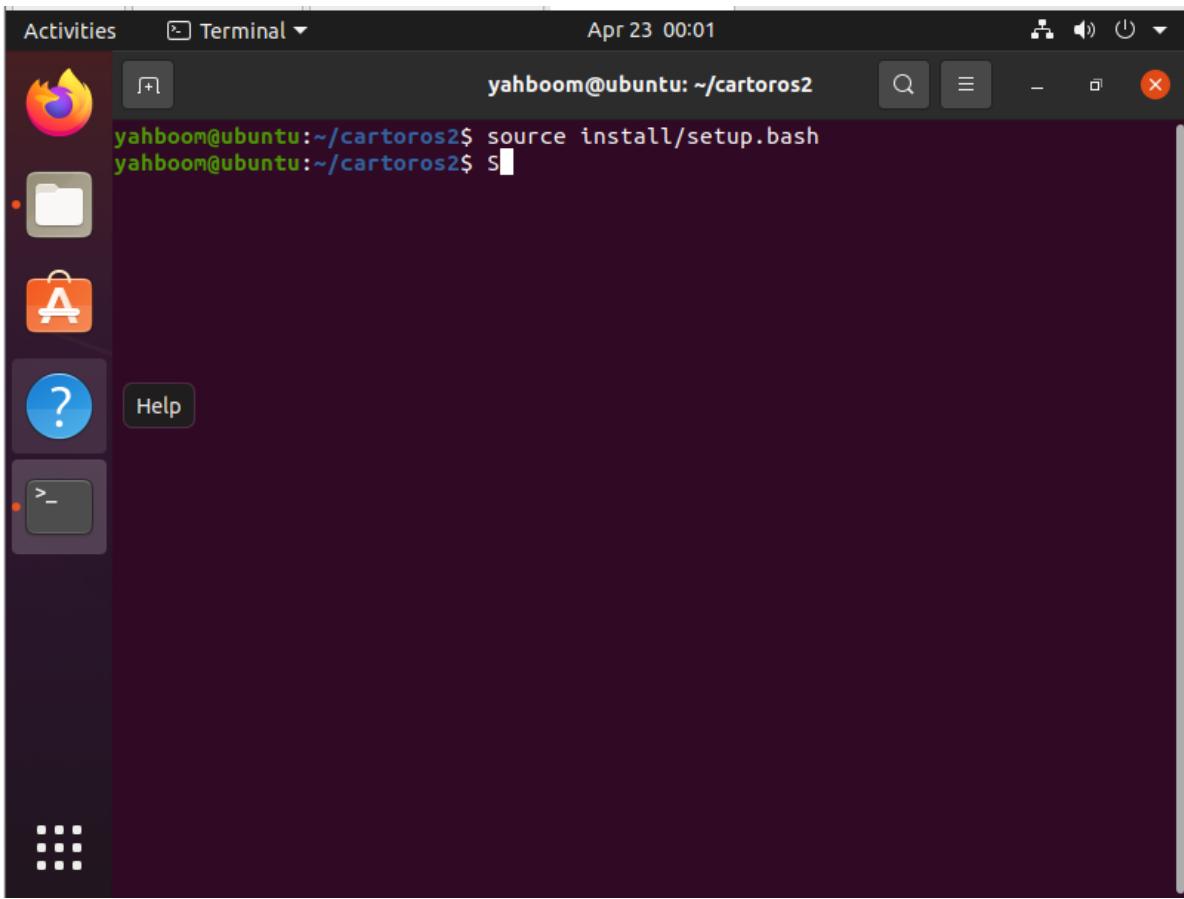


Open a terminal under the folder



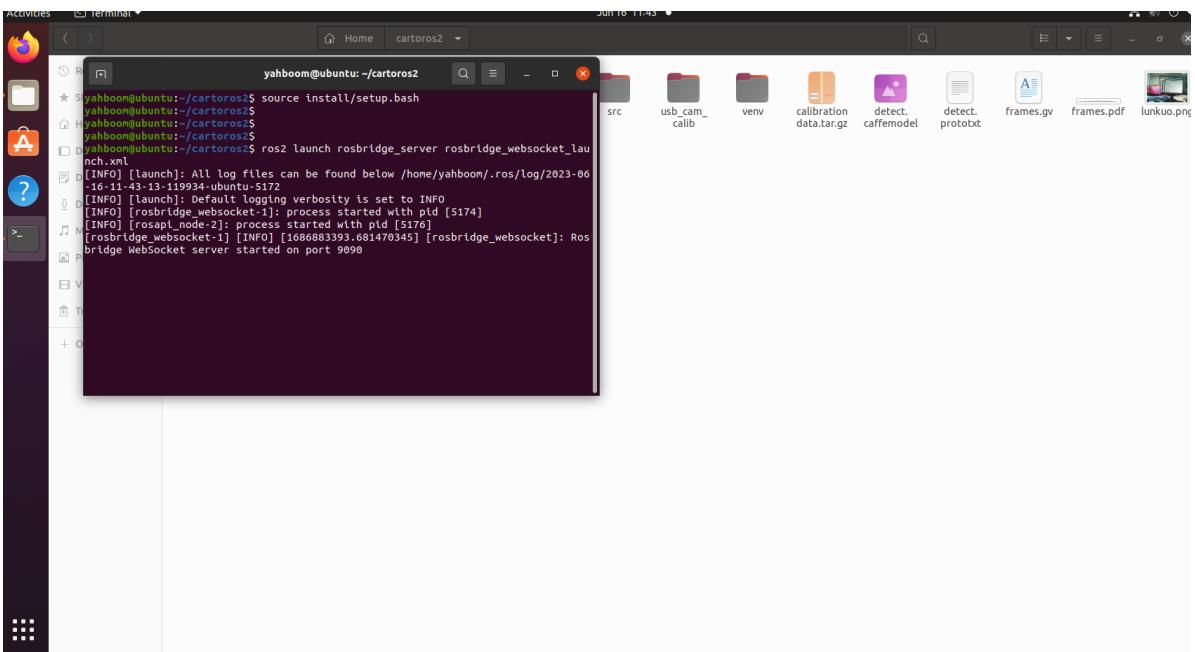
Then enter the following command

```
source install/setup.bash
```



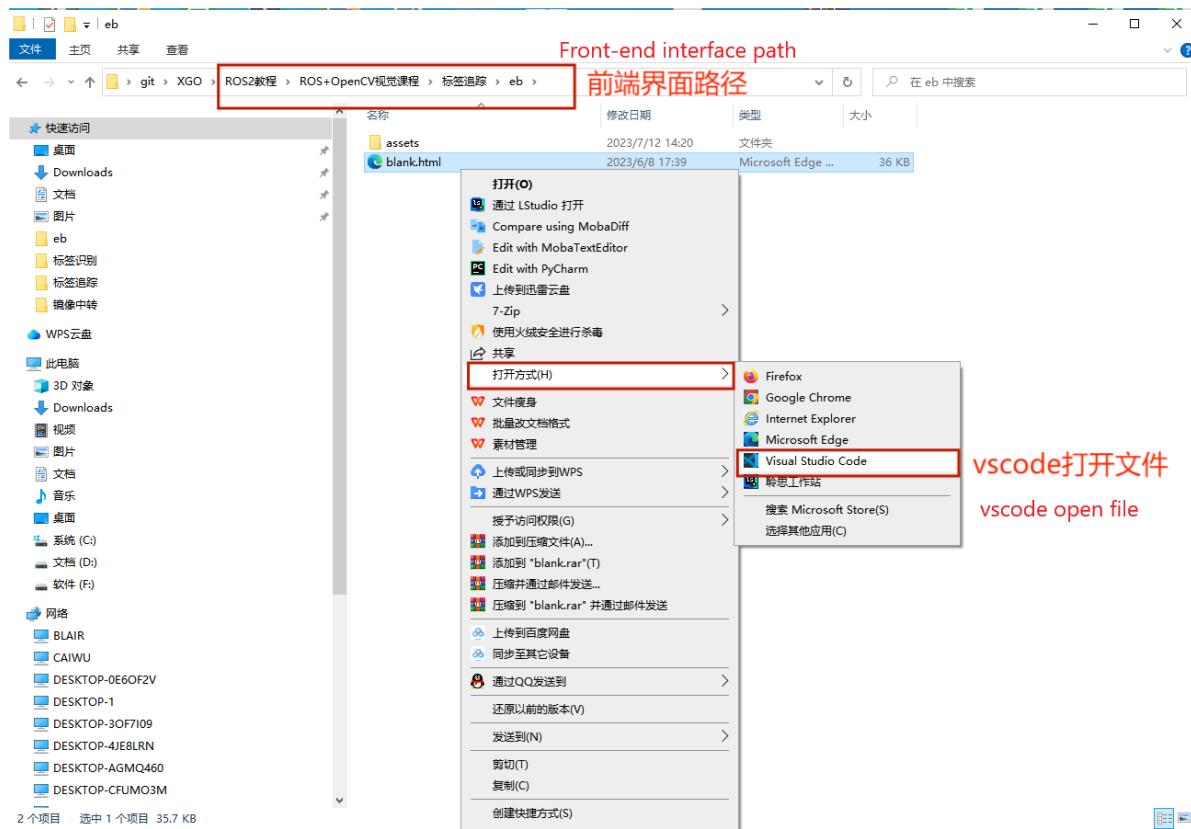
Then start rosbridge and enter the following command

```
ros2 launch rosbridge_server rosbridge_websocket_launch.xml
```



Find the blank.html file in the eb folder in the more directory of the tutorial and open it with Google Chrome.

Note: Here you need to set the IP address of rosbridge. Get the IP address of the virtual machine, then open the blank.html file, change the IP address in line 363 of the code and save it as shown below.

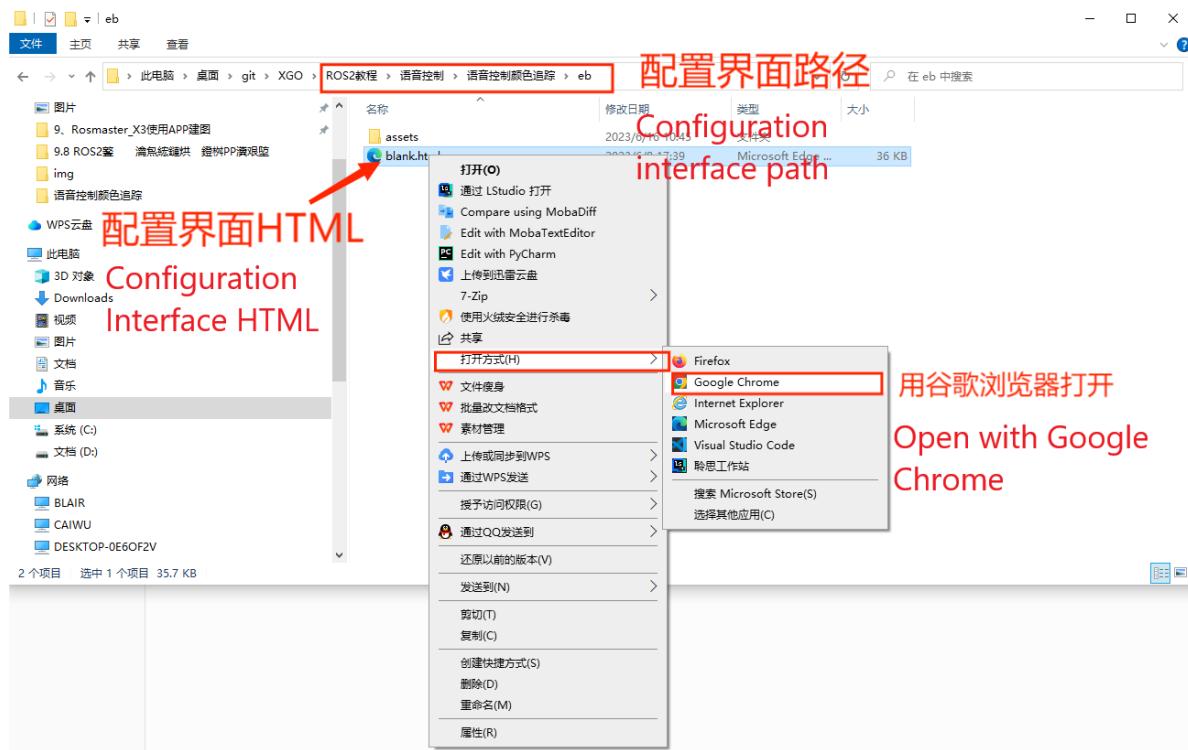


修改这里的IP为实际rosb_ridge的的IP地址
Modify the IP here to the IP address of the actual rosbridge

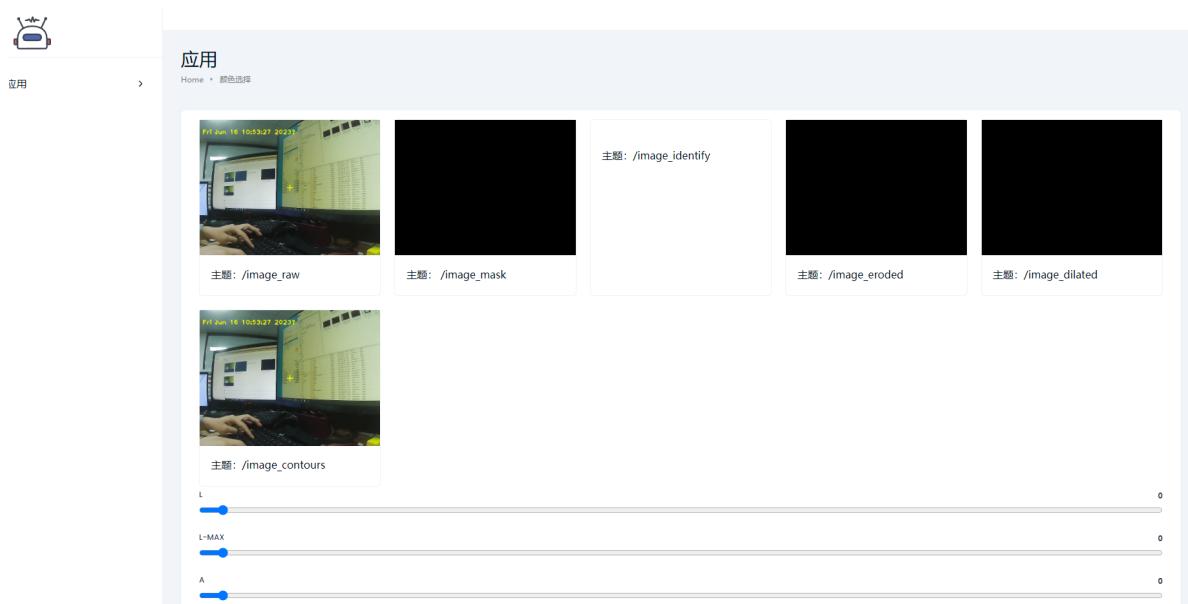
```

345
346
347
348     </div>
349
350     </div>
351
352     <script src="assets/js/roslib.js"></script>
353     <script src="assets/js/alpine.js"></script>
354     <script src="assets/js/perfect-scrollbar.js"></script>
355     <script src="assets/js/choices.js"></script>
356     <script src="assets/js/chart.js"></script>
357     <script src="assets/js/apexchart.js"></script>
358     <script src="assets/js/quill.js"></script>
359     <script src="assets/js/rangeslider.min.js"></script>
360     <script src="assets/js/main.js"></script>
361
362     var ros = new ROSLIB.Ros({
363       url : 'ws://192.168.2.117:9090'
364     });
365
366     ros.on('connection', function() {
367       console.log('Connected to websocket server.');
368     });
369
370     ros.on('error', function(error) {
371       console.log('Error connecting to websocket server: ', error);
372     });
373
374     ros.on('close', function() {
375       console.log('Connection to websocket server closed.');
376     });
377
378     var imageListener_compressed= new ROSLIB.Topic({
379       ros : ros,
380       name : 'image_raw/compressed',
381       messageType : 'sensor_msgs/msg/CompressedImage',
382       throttle_rate : 100
383     });
384     var that = this;
385     imageListener_compressed.subscribe(function(data) {
386       // update the robots position on the path

```



As shown in the figure below, you can see the pictures transferred by the camera.



We then set the LAB value of the colour via the slider bar.

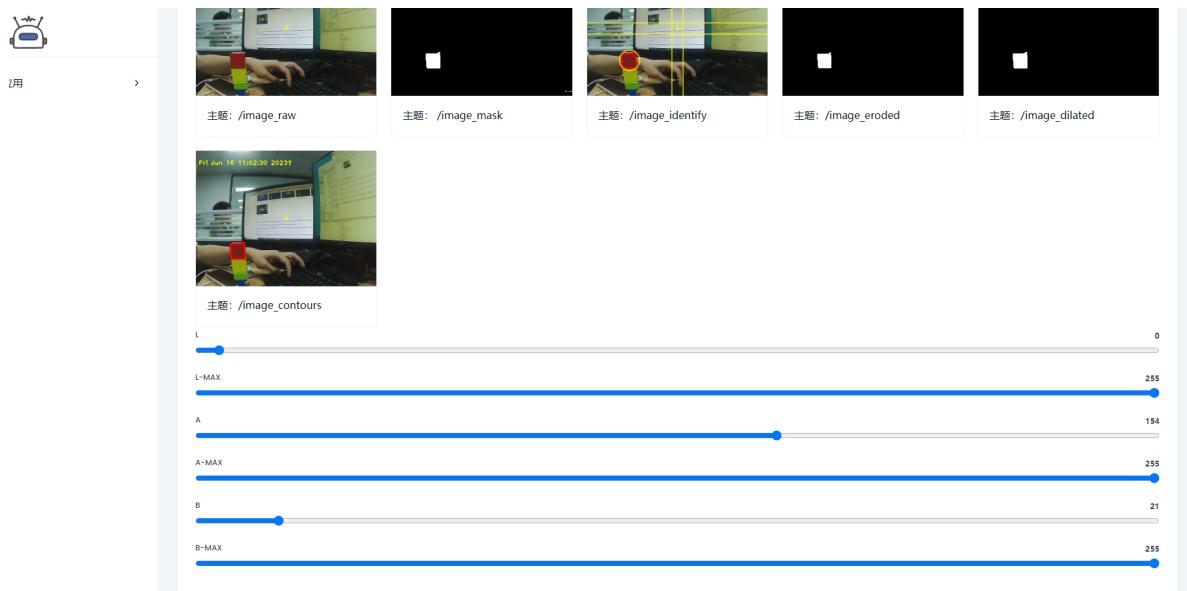
```
yellow: {"l":96, "a": 55, "b":188, "l_max": 252 , "a_max": 141, "b_max": 255}
```

```
red: {"l":0, "a": 155, "b":21, "l_max": 255 , "a_max": 255, "b_max": 255}
```

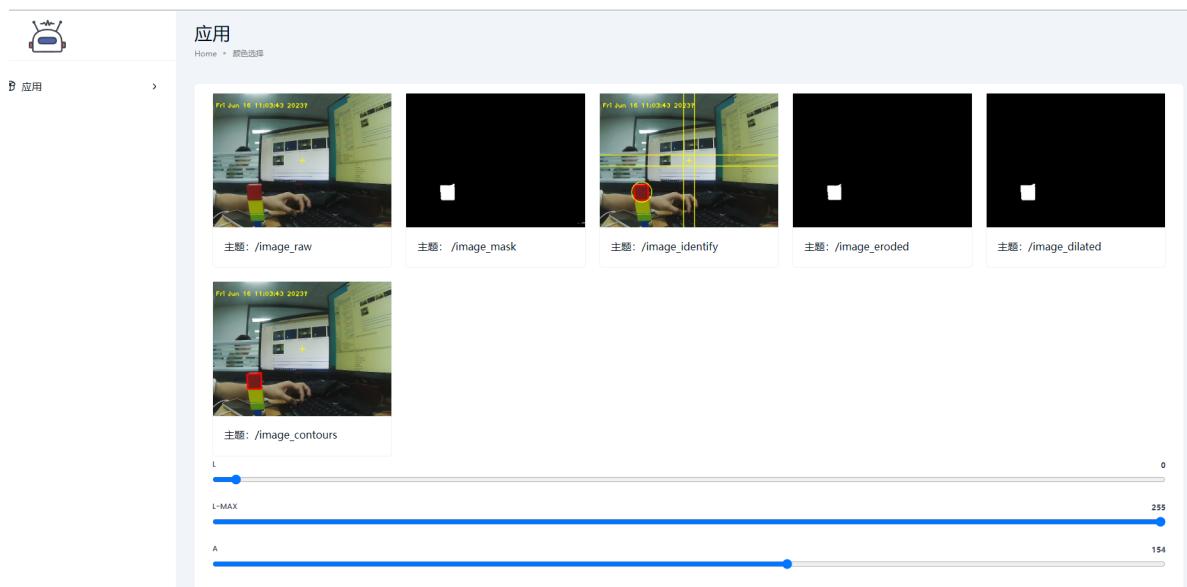
```
green: {"l":26, "a": 7, "b":170, "l_max": 143 , "a_max": 110, "b_max": 255}
```

```
blue: {"l":0, "a": 0, "b":0, "l_max": 255 , "a_max": 255, "b_max": 102}
```

Above are the LAB values for several colours, we can choose one to set. For example, let's set the colour red, as shown below, and move the slider bar.



In the figure we can see that the red square is recognised.



In the shell terminal we open a new terminal and in the terminal we enter the command

Note: The terminal here is the one connected to the robot dog, not the virtual machine.

```
cd cartographer_ws2/
```

```
source install/setup.bash
```

```
ros2 run voice_xgo_ctrl_run voice_xgo_ctrl_color_identify
```

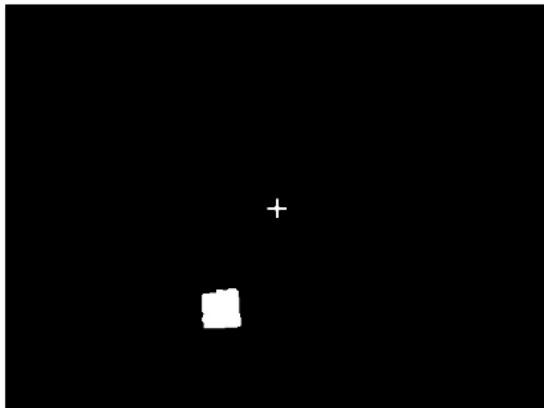
```
pi@yahboom:~$ cd cartographer_ws2/
pi@yahboom:~/cartographer_ws2$
pi@yahboom:~/cartographer_ws2$ source install/setup.bash
pi@yahboom:~/cartographer_ws2$
pi@yahboom:~/cartographer_ws2$ ros2 run voice_xgo_ctrl_run voice_xgo_ctrl_color_identify
Speech Serial Opened! Baudrate=115200
```

Then say to the robot dog, "Hi, Yahboom."

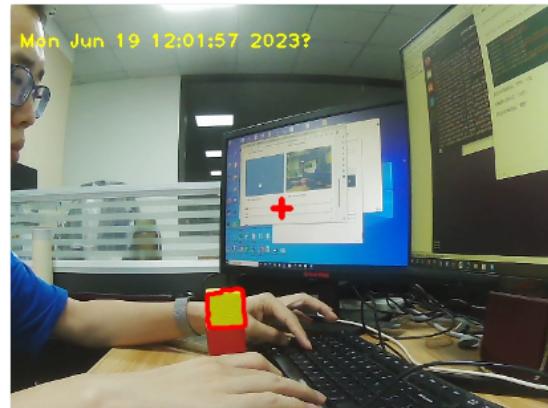
The robot dog will respond by saying, "Hi, I am here"

Then say to the robot dog, "yellow colour"

The yellow color will be recognized, as shown below



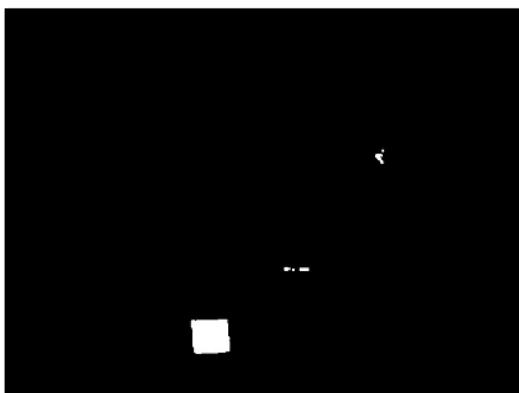
主题: /image_dilated



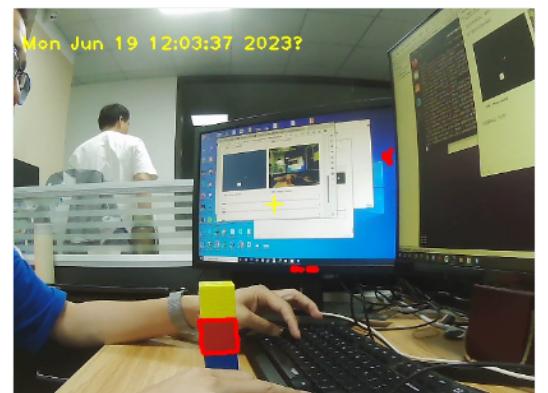
主题: /image_contours

Say "red colour" to the robot dog.

The red color will be recognized, as shown below.



主题: /image_dilated



主题: /image_contours