

4. Object recognition

1. Experimental purpose

Drive the car to recognize objects

2. Experimental path source code

Enter the car system, end the car program, enter "ip (ip is the car's ip): 8888" in the browser, enter the password "yahboom"



Password: yahboom

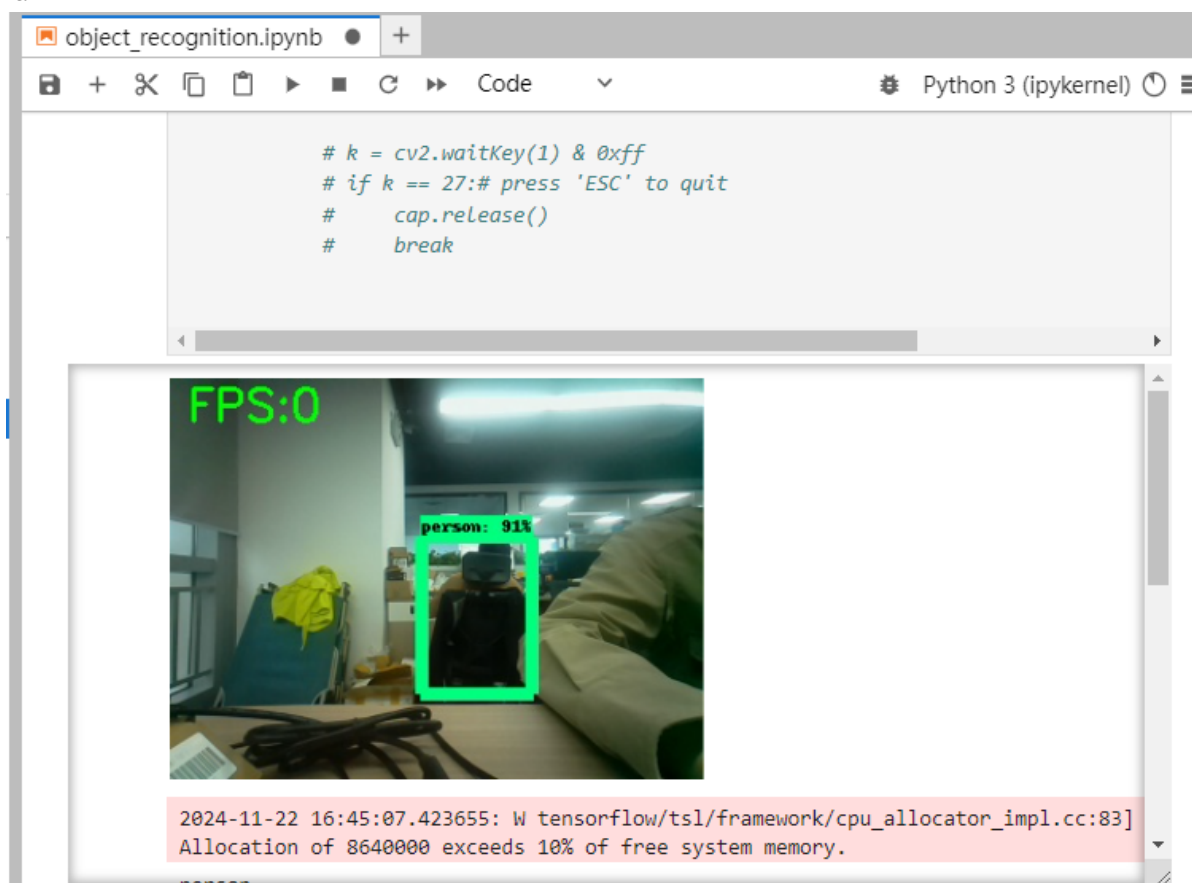
Log in

Then log in

Enter the path of **Rider-pi_class/5.AI Visual Recognition Course/4. Object recognition** and run **object_recognition.ipynb**.

3. Experimental phenomenon

After running the source code, you can see that the car will recognize the current target and select it.



4. Main source code analysis

```
# Main
t_start = time.time()
fps = 0
display(image_widget)
with detection_graph.as_default():
    with tf.compat.v1.Session(graph=detection_graph) as sess:
        while True:
            ret, frame = cap.read()
            #####
            image_np_expanded = np.expand_dims(frame, axis=0)
            image_tensor = detection_graph.get_tensor_by_name('image_tensor:0')
            detection_boxes =
detection_graph.get_tensor_by_name('detection_boxes:0')
            detection_scores =
detection_graph.get_tensor_by_name('detection_scores:0')
            detection_classes =
detection_graph.get_tensor_by_name('detection_classes:0')
            num_detections =
detection_graph.get_tensor_by_name('num_detections:0')

#         print('Running detection..')
        (boxes, scores, classes, num) = sess.run(
            [detection_boxes, detection_scores, detection_classes,
num_detections],
            feed_dict={image_tensor: image_np_expanded})

#         print('Done. Visualizing..')
        vis_utils.visualize_boxes_and_labels_on_image_array(
            frame,
            np.squeeze(boxes),
            np.squeeze(classes).astype(np.int32),
            np.squeeze(scores),
            category_index,
            use_normalized_coordinates=True,
            line_thickness=8)

        for i in range(0, 10):
            if scores[0][i] >= 0.5:
                print(category_index[int(classes[0][i])]['name'])
                objtype_str=category_index[int(classes[0][i])]['name']
            #####
            fps = fps + 1
            mfps = fps / (time.time() - t_start)
            cv2.putText(frame, "FPS:" + str(int(mfps)), (10,25),
cv2.FONT_HERSHEY_SIMPLEX, 0.9, (0,255,0), 2)
            image_widget.value = bgr8_to_jpeg(frame)

#Display on the LCD screen of the car
b,g,r = cv2.split(frame)
img = cv2.merge((r,g,b))
imgok = Image.fromarray(img)
mydisplay.ShowImage(imgok)
```

Through the source code, call the relevant AI recognition model. The car will recognize the current object and display the recognition results on the car screen and the power screen.

Because the memory of this car is only 2GB, this case will be stuck.