

8.7、Mask detection

8.7、Mask detection

8.7.1、Experimental goal

8.7.2、Preparation before experiment

8.7.3、The process of the experiment

8.7.4、Experimental effect

8.7.5、Experiment summary

8.7.1、Experimental goal

This lesson mainly learns the mask detection function, analyzes the screen captured by the camera, compares the model, analyzes whether to wear a mask, and prints out the status of wearing a mask

The reference code path for this experiment is: K210_Broad\05-AI\face_mask_detect.py

8.7.2、Preparation before experiment

Please import the model file to the memory card first, and then insert the memory card into the memory card slot of the K210 development board. For specific steps, please refer to:

[Appendix: Import model files to memory card](#)

8.7.3、The process of the experiment

The factory firmware of the module has integrated the AI vision algorithm module. If you have downloaded other firmware, please burn it back to the factory firmware and then conduct experiments.

1. Import related libraries, and initialize camera and LCD display.

```
import sensor, image, time, lcd
from maix import KPU

lcd.init()
sensor.reset()
sensor.set_pixformat(sensor.RGB565)
sensor.set_framesize(sensor.QVGA)
sensor.skip_frames(time = 1000)
clock = time.clock()
```

2. Initialize KPU related parameters, kpu needs to load the kmodel file, the model file path required for this experiment is: /sd/KPU/face_mask_detect/detect_5.kmodel, and use yolo2 to calculate whether it meets the model requirements. od_img is the image of the neural network, with a size of 320*256, which is used for subsequent storage of the camera image and passed to the KPU

for calculation.

```
od_img = image.Image(size=(320,256), copy_to_fb=False)

anchor = (0.156250, 0.222548, 0.361328, 0.489583, 0.781250, 0.983133, 1.621094,
1.964286, 3.574219, 3.94000)
kpu = KPU()
print("ready load model")
kpu.load_kmodel("/sd/KPU/face_mask_detect/detect_5.kmodel")
kpu.init_yolo2(anchor, anchor_num=5, img_w=320, img_h=240, net_w=320 , net_h=256
,layer_w=10 ,layer_h=8, threshold=0.7, nms_value=0.4, classes=2)
```

3. Create a while loop, transfer the image to the KPU for calculation, and use the yolo2 neural network algorithm to solve the calculation. If it is a face wearing a mask, it will be framed in green and "with mask" will be displayed. If it is a person not wearing a mask The face is framed in red and "without mask" is displayed.

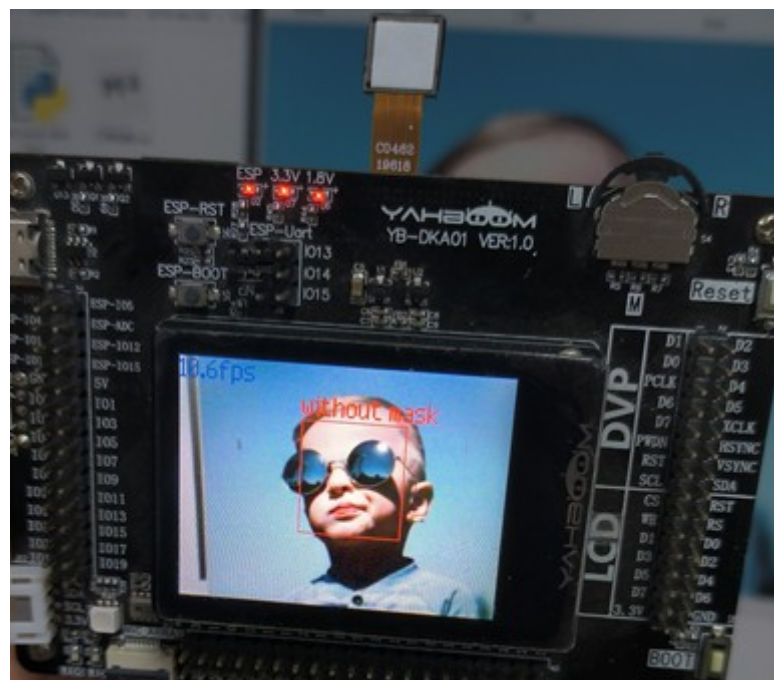
```
while True:
    clock.tick()
    img = sensor.snapshot()
    od_img.draw_image(img, 0,0)
    od_img.pix_to_ai()
    kpu.run_with_output(od_img)
    dect = kpu.regionlayer_yolo2()
    fps = clock.fps()
    if len(dect) > 0:
        print("dect:", dect)
        for l in dect :
            if l[4] :
                img.draw_rectangle(l[0],l[1],l[2],l[3], color=(0, 255, 0))
                img.draw_string(l[0],l[1]-24, "with mask", color=(0, 255, 0),
scale=2)
            else:
                img.draw_rectangle(l[0],l[1],l[2],l[3], color=(255, 0, 0))
                img.draw_string(l[0],l[1]-24, "without mask", color=(255, 0, 0),
scale=2)

    img.draw_string(0, 0, "%2.1ffps" %(fps), color=(0, 60, 128), scale=2.0)
    lcd.display(img)
```

8.7.4、 Experimental effect

Experimental effect Connect the K210 development board to the computer through the TYPE-C data cable, click the connect button in CanMV IDE, and click the run button after the connection is completed to run the routine code. You can also download the code as main.py to the K210 development board to run.

After the system initialization is complete, the LCD will display the camera image, and use the camera to capture faces. When a mask is worn, a green frame and "with mask" will be displayed. When a mask is not worn, a red frame and "without mask" will be displayed. At the same time, the serial terminal at the bottom of the IDE will print face-related information.



```
串行终端 | 图标 图标 ^
dect: [[187, 44, 102, 101, 0, 0.7043596]]
dect: [[187, 40, 102, 112, 0, 0.7640823]]
dect: [[185, 44, 102, 100, 0, 0.8318885]]
dect: [[184, 37, 102, 112, 0, 0.8166991]]
dect: [[182, 37, 103, 113, 0, 0.8869399]]
dect: [[182, 14, 102, 127, 0, 0.8089279]]
dect: [[183, 14, 102, 127, 0, 0.7507568]]
dect: [[187, 23, 91, 112, 0, 0.7318366]]
```

8.7.5、Experiment summary

Mask detection needs to use the memory card to load the model file, so you need to import the model file into the memory card in advance, and then insert the memory card into the memory card slot of the K210 development board. If the model file in the memory card cannot be read, it will error.

The current mask detection threshold is threshold=0.7. If you need to detect faces more accurately, you can adjust the threshold appropriately.