

Face Positioning

Orin board users can directly open a web page and enter IP address:8888 to access jupyter-lab and run directly. Jetson-Nano board users need to first enter the docker container, then enter the following command in docker:

```
cd
jupyter-lab --allow-root
```

Then open a web page and enter IP address:9999 to access jupyter-lab and run the following program.

The principle of the face positioning experiment is to determine the position information of the face relative to the camera by calculating the center point coordinates of the face in the camera image, thereby achieving face positioning. The experimental results show that it will continuously find the center point of the face, print the center point coordinates, and draw a box on the face.

Code path:

```
#Jetson-Nano users need to enter the docker container to view
~/dofbot_pro/dofbot_face_follow/scripts/Face_position.ipynb
```

1. Main Code

- Import header files

```
import cv2 as cv
import threading
from time import sleep
import ipywidgets as widgets
from IPython.display import display
from face_position import Face_Position
from dofbot_utils.robot_controller import Robot_Controller
from dofbot_utils.fps import FPS
from dofbot_utils.dofbot_config import *
```

- Create instances, initialize parameters

```
robot = Robot_Controller()
robot.move_init_pose()
fps = FPS()
face = Face_Position()
model = 'General'
```

- Main process

```
def camera():
    global model
    # 打开摄像头 Open camera
    capture = cv.VideoCapture(0, cv.CAP_V4L2)
    capture.set(3, 640)
```

```

capture.set(4, 480)
capture.set(5, 30)
while capture.isOpened():
    try:
        _, img = capture.read()
        fps.update_fps()
        img, pos = follow.follow_function(img)
        if model == 'Exit':
            cv.destroyAllWindows()
            capture.release()
            break
        fps.show_fps(img)
        imgbox.value = cv.imencode('.jpg', img)[1].tobytes()
    except KeyboardInterrupt: capture.release()

```

- Startup

```

display(controls_box,output)
threading.Thread(target=camera, ).start()

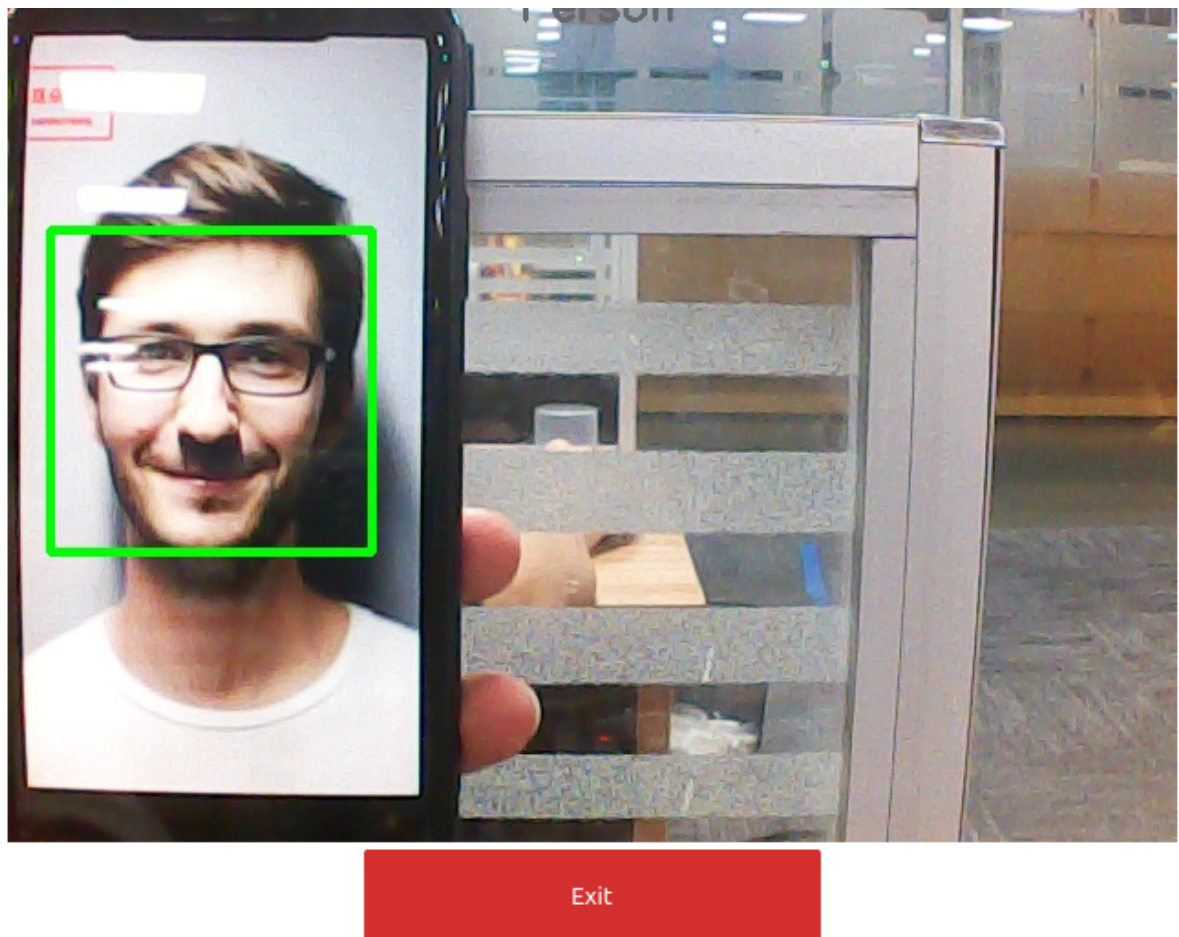
```

2. Run the Program

Click the run entire program button on the jupyterlab toolbar, then scroll to the bottom.



After the program starts, place a face image in the camera view. You can see a green box framing the face. As you move the face image, the box will also follow the face movement.



Open Log information, select Log Level as Info, you can see the printed face position coordinates.

Log: dofbot_ws/src/dofbot X	
+ Add Checkpoint ⌂ Clear Log Log Level: Info ▼	
4:57:52 PM	x=312, y=263 x=308, y=270 x=305, y=276 x=299, y=282 x=304, y=285 x=304, y=287 x=309, y=286 x=310, y=285 x=312, y=283 x=323, y=231

If you need to end the program, please click [Exit] to avoid affecting other programs from calling resources.