

Face tracking

1. Function Introduction

Based on the facial positioning function, combined with the robotic arm to achieve facial tracking function.

Code path: ~/jetcobot_ws/src/jetcobot_face_follow/face_tracking.ipynb

2. About code

- Import header file

```
import cv2 as cv
import threading
from time import sleep
import ipywidgets as widgets
from IPython.display import display
from face_follow import face_follow
```

- Create an instance and initialize parameters

```
# Create an instance
follow = face_follow()
# Initialization mode
model = 'General'
```

- Create Control

```
button_layout = widgets.Layout(width='250px', height='50px',
align_self='center')
output = widgets.Output()
# 退出控件 exit button
exit_button = widgets.Button(description='Exit', button_style='danger',
layout=button_layout)
# 图像控件 Image widget
imgbox = widgets.Image(format='jpg', height=480, width=640,
layout=widgets.Layout(align_self='center'))
# 空间布局 spatial distribution
controls_box = widgets.VBox([imgbox, exit_button],
layout=widgets.Layout(align_self='center'))
# ['auto', 'flex-start', 'flex-end', 'center', 'baseline', 'stretch', 'inherit',
'initial', 'unset']
```

- Mode switching

```
def exit_button_Callback(value):
    global model
    model = 'Exit'
    # with output: print(model)
    exit_button.on_click(exit_button_Callback)
```

- Main program

```
def camera():
    global model
    # 打开摄像头 Open camera
    capture = cv.VideoCapture(0)
    while capture.isOpened():
        try:
            _, img = capture.read()
            img = cv.resize(img, (640, 480))
            img = follow.follow_function(img)
            if model == 'Exit':
                cv.destroyAllWindows()
                capture.release()
                break
            imgbox.value = cv.imencode('.jpg', img)[1].tobytes()
        except KeyboardInterrupt: capture.release()
```

- Start

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

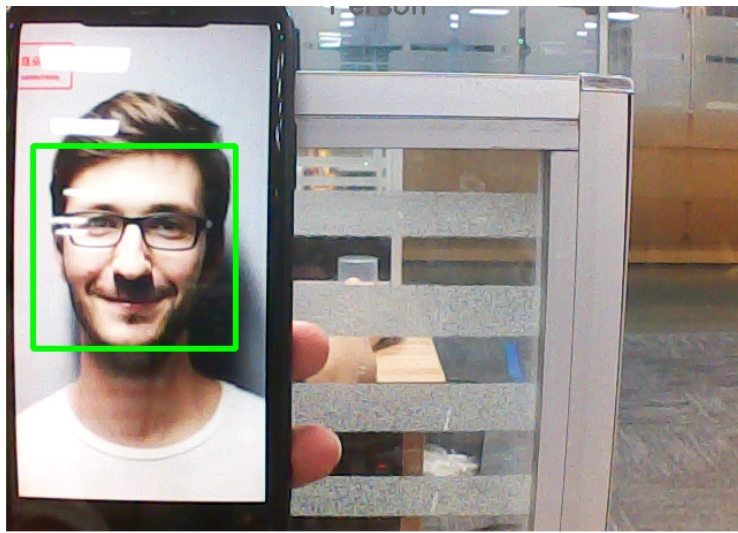
3. Run program

Click the run button on the jupyterlab toolbar, run the entire program, and then drag it to the bottom.



You can see the camera image, and when you place the face into the camera image, the robotic arm will move along with the face.

Note: When moving the face, the speed should not be too fast, otherwise the robotic arm may not be able to keep up due to the movement being too fast.



Exit