/home/pi/Yahboom_Project/Raspbot/3.AI Vision course/09.Face tracking/Face tracking.ipynb

```
import enum
import cv2

def bgr8_to_jpeg(value, quality=75):
    return bytes(cv2.imencode('.jpg', value)[1])
```

```
import cv2
import ipywidgets.widgets as widgets
import threading
import time
import sys

image_widget = widgets.Image(format='jpeg', width=320, height=240)
display(image_widget)
```

```
global face_x, face_y, face_w, face_h
face_x = face_y = face_w = face_h = 0
global target_valuex
target_valuex = 2048
global target_valuey
target_valuey = 2048
```

```
import PID
xservo_pid = PID.PositionalPID(1.1, 0.2, 0.8)
yservo_pid = PID.PositionalPID(0.8, 0.2, 0.8)
```

```
import YB_Pcb_Car
car = YB_Pcb_Car.YB_Pcb_Car()
car.Ctrl_Servo(1,90)
car.Ctrl_Servo(2,90)
```

```
import cv2
#face_haar = cv2.CascadeClassifier('haarcascade_profileface.xml')
#face_haar = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
face_haar = cv2.CascadeClassifier('123.xml')
```

```
image = cv2.VideoCapture(0)
image.set(3,320)
image.set(4,240)
image.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter.fourcc('M', 'J', 'P', 'G'))
image.set(cv2.CAP_PROP_BRIGHTNESS, 62)
```

```
image.set(cv2.CAP_PROP_CONTRAST, 63)
image.set(cv2.CAP_PROP_EXPOSURE, 4800)
```

```
while 1:
    ret, frame = image.read()
    gray_img = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_haar.detectMultiScale(gray_img, 1.1, 3)

    #xservo_pid = PID.PositionalPID(XServo_P.value, XServo_I.value, XServo_D.value)
    #yservo_pid = PID.PositionalPID(YServo_P.value, YServo_I.value, YServo_D.value)

    if len(faces) > 0:

        (face_x, face_y, face_w, face_h) = faces[0]
        #
cv2.rectangle(frame,(face_x+10,face_y),(face_x+face_w-10,face_y+face_h+20),(0,255,0),2)

cv2.rectangle(frame,(face_x,face_y),(face_x+face_w,face_y+face_h),(0,255,0),2)

        #Proportion-Integration-Differentiation
        xservo_pid.SystemOutput = face_x + face_w/2
        xservo_pid.SetStepSignal(150)
        xservo_pid.SetInertiaTime(0.01, 0.1)
        target_valuex = int(1500 + xservo_pid.SystemOutput)
        target_servox = int((target_valuex-500)/10)
        if target_servox > 180:
            target_servox = 180
        if target_servox < 0:
            target_servox = 0
        yservo_pid.SystemOutput = face_y + face_h/2
        yservo_pid.SetStepSignal(120)
        yservo_pid.SetInertiaTime(0.01, 0.1)
        target_valuey = int(1500 - yservo_pid.SystemOutput)
        target_servoy = int((target_valuey-500)/10)
        #print("target_servoy %d", target_servoy)
        if target_servoy > 180:
            target_servoy = 180
        if target_servoy < 0:
            target_servoy = 0
        #robot.Servo_control(target_valuex,target_valuey)

        car.Ctrl_Servo(1, target_servox)
```

```
            car.Ctrl_Servo(2, target_servoy)

        try:
            image_widget.value = bgr8_to_jpeg(frame)
        except:
            continue
```

```
image.release()
```