7. Face Detection

7.1 Introduction

This course mainly uses RDK-X3's CSI camera to get the camera feed, detect faces and frame them.

7.2 Core Content Analysis

Load the face Haar feature data.

```
face_haar = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
```

The image is passed into the face haar parser to extract the face position information.

```
def face_detect(image):
    gray_img = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    faces = face_haar.detectMultiScale(gray_img, 1.1, 3)
    for face_x,face_y,face_w,face_h in faces:
        return "1", (face_x,face_y,face_w,face_h)
    return None, (0,0,0,0)
```

When the start switch is turned on, it determines whether there is a human face in the returned image and frames the face if there is.

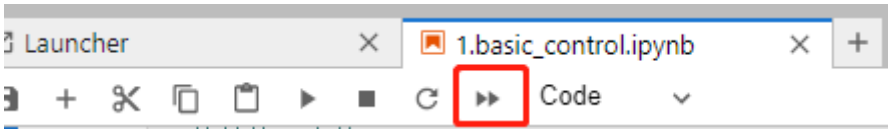
```
state, (face_x,face_y,face_w,face_h) = face_detect(frame)
if state != None:
    cv2.rectangle(frame, (face_x, face_y), (face_x+face_w, face_y+face_h),
(0,255,255), 1)
```

7.3 Play Operations

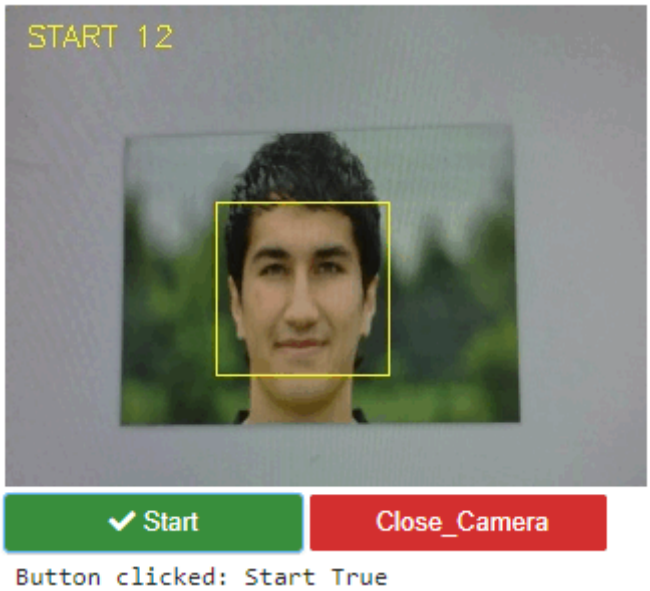
Open the jupyterLab client and find the code path:

```
/root/sunriseRobot/Samples/2_AI/07_face_detect/face_detect.ipynb
```

Click on Run All Cells and then pull down to the bottom to see the generated controls.



Click the Start start button below to turn on the Detect Face feature, which uses a box to frame out faces when they appear in the camera frame.



Finally click the Close_Camera button to close the camera.