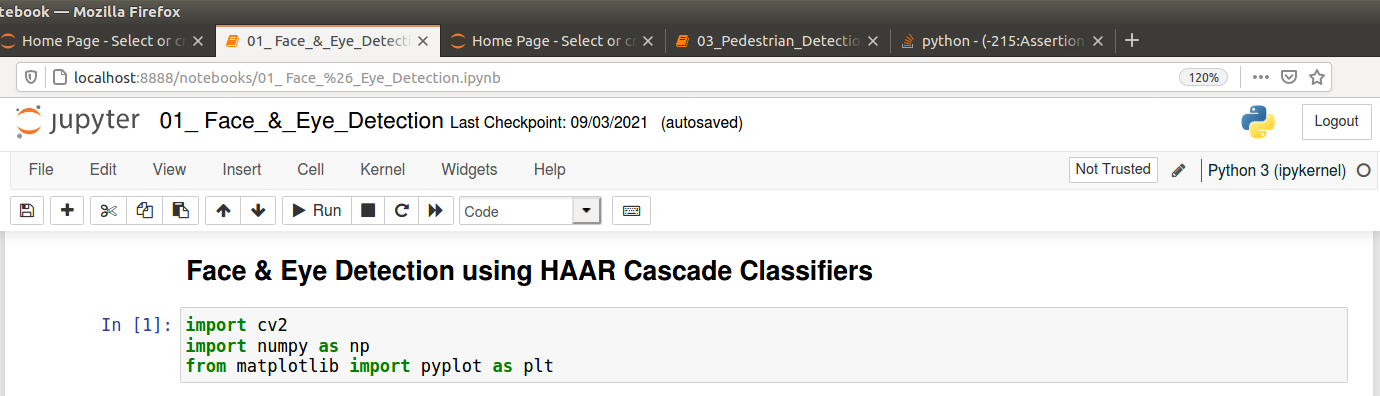
Face & Eye Detection using HAAR Cascade Classifiers

* First download OpenCV then download project file from github and run the code in jupyter notebook…

1. import cv2

import numpy as np

from matplotlib import pyplot as plt



1. # Load our image then convert it to grayscale

image = cv2.imread('../../images/Trump.jpg')

gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

plt.imshow(cv2.cvtColor(image, cv2.COLOR\_BGR2RGB))

plt.title('Input Image')

cv2.imwrite('input\_image.jpg', image)

plt.show()



1. # We point OpenCV's CascadeClassifier function to where our

# classifier (XML file format) is stored

face\_classifier = cv2.CascadeClassifier('../../Haarcascades/haarcascade\_frontalface\_default.xml')

# Our classifier returns the ROI of the detected face as a tuple

# It stores the top left coordinate and the bottom right coordiantes

faces = face\_classifier.detectMultiScale(gray, 1.3, 5)

# When no faces detected, face\_classifier returns and empty tuple

if faces is ():

print("No faces found")

# We iterate through our faces array and draw a rectangle

# over each face in faces

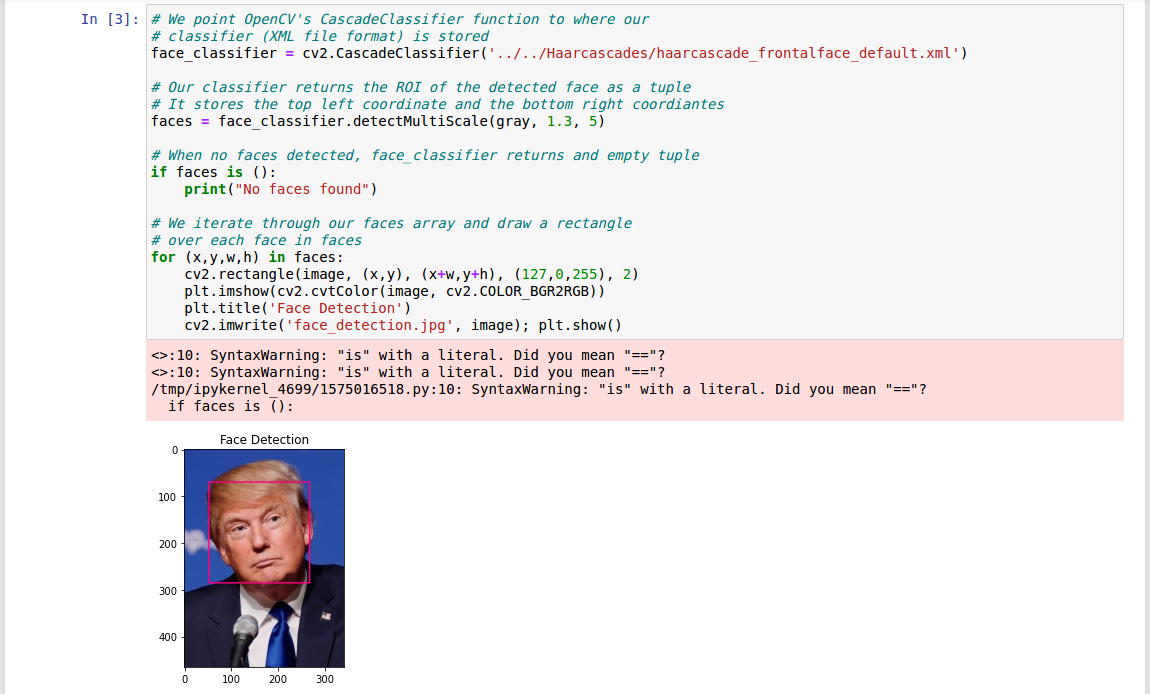
for (x,y,w,h) in faces:

cv2.rectangle(image, (x,y), (x+w,y+h), (127,0,255), 2)

plt.imshow(cv2.cvtColor(image, cv2.COLOR\_BGR2RGB))

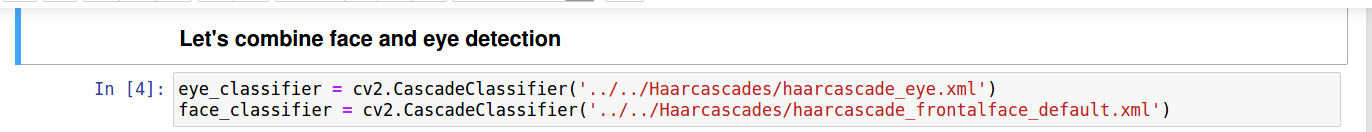
plt.title('Face Detection')

cv2.imwrite('face\_detection.jpg', image); plt.show()



4. eye\_classifier = cv2.CascadeClassifier('../../Haarcascades/haarcascade\_eye.xml')

face\_classifier = cv2.CascadeClassifier('../../Haarcascades/haarcascade\_frontalface\_default.xml')



5. img = cv2.imread('../../images/Trump.jpg')

gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

faces = face\_classifier.detectMultiScale(gray, 1.3, 5)

# When no faces detected, face\_classifier returns and empty tuple

if faces is ():

print("No Face Found")

for (x,y,w,h) in faces:

cv2.rectangle(img,(x,y),(x+w,y+h),(127,0,255),2)

plt.imshow(cv2.cvtColor(img, cv2.COLOR\_BGR2RGB))

plt.title('img'); plt.show()

cv2.imwrite('img.jpg', image)

roi\_gray = gray[y:y+h, x:x+w]

roi\_color = img[y:y+h, x:x+w]

eyes = eye\_classifier.detectMultiScale(roi\_gray)

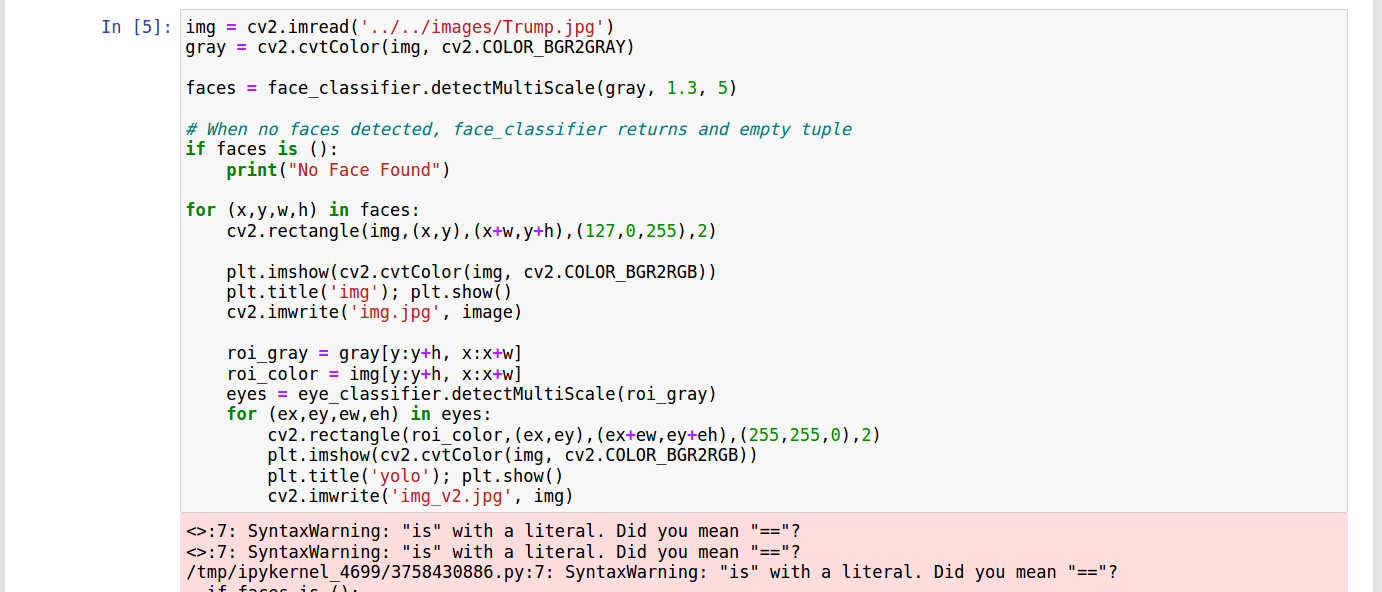
for (ex,ey,ew,eh) in eyes:

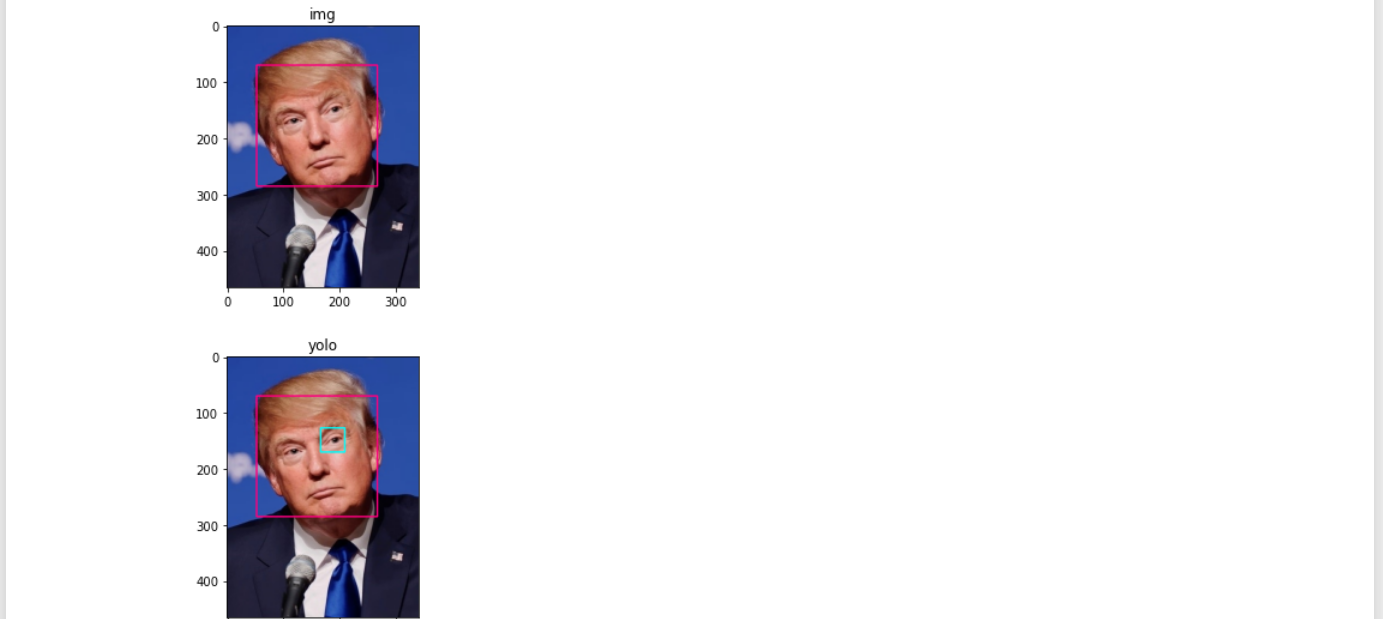
cv2.rectangle(roi\_color,(ex,ey),(ex+ew,ey+eh),(255,255,0),2)

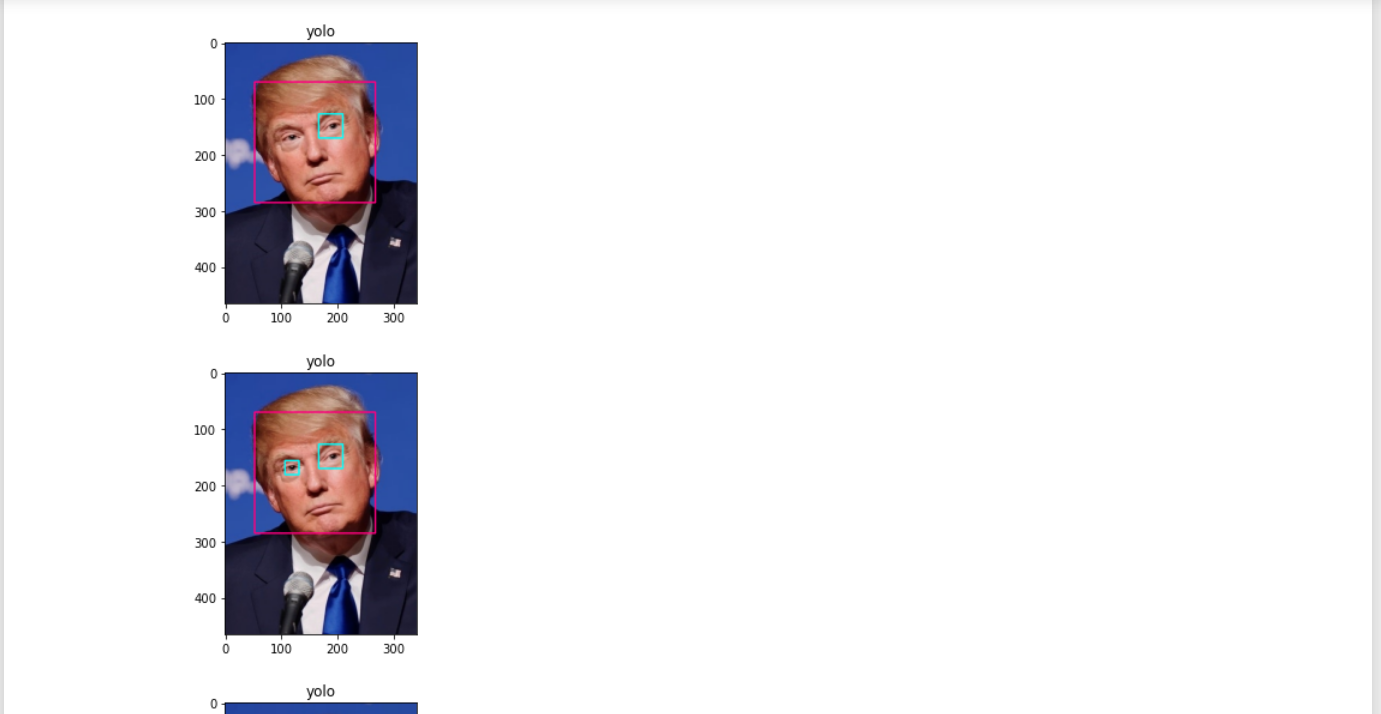
plt.imshow(cv2.cvtColor(img, cv2.COLOR\_BGR2RGB))

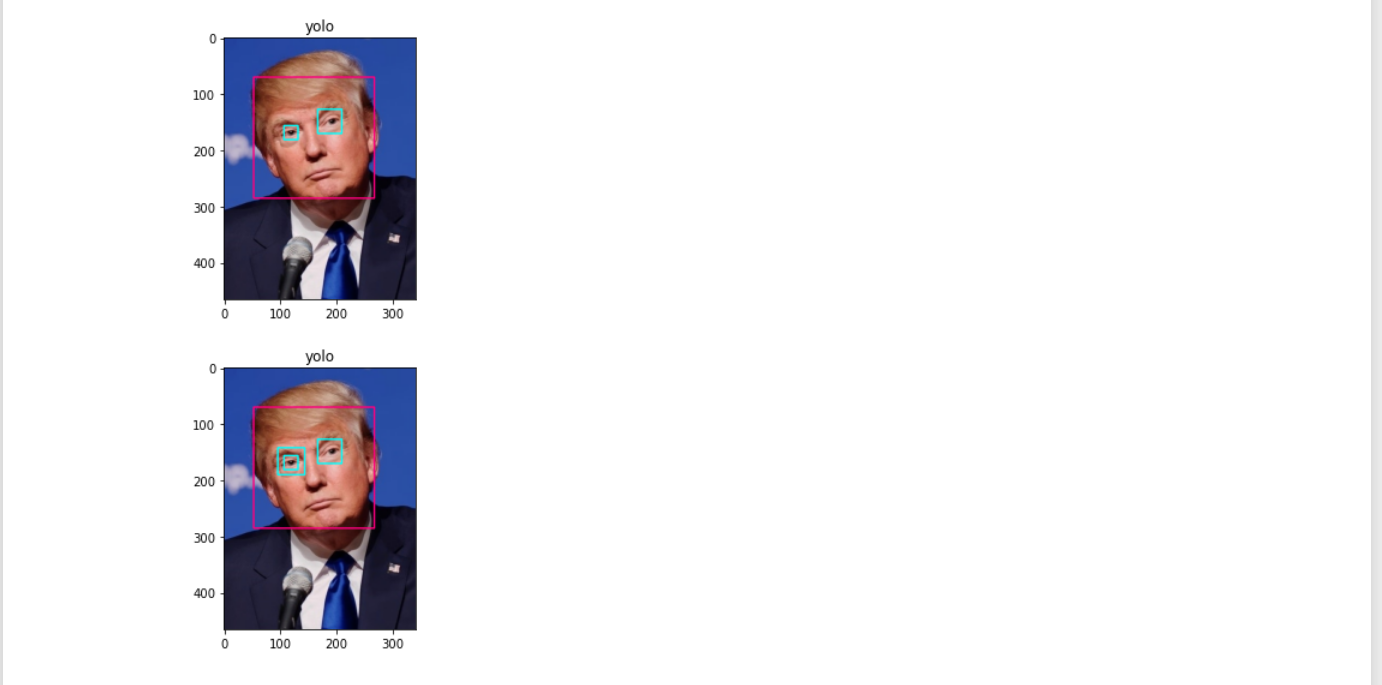
plt.title('yolo'); plt.show()

cv2.imwrite('img\_v2.jpg', img)



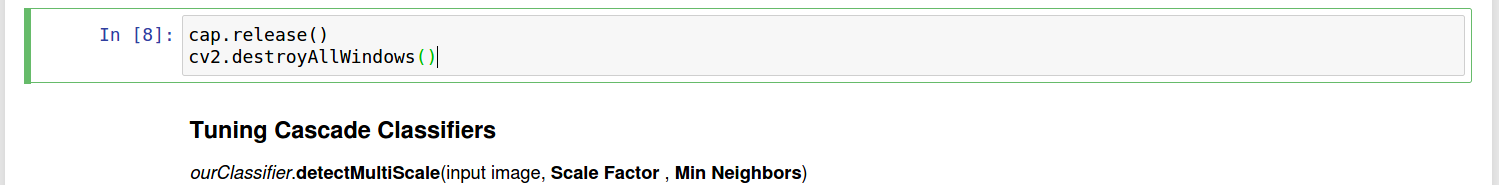


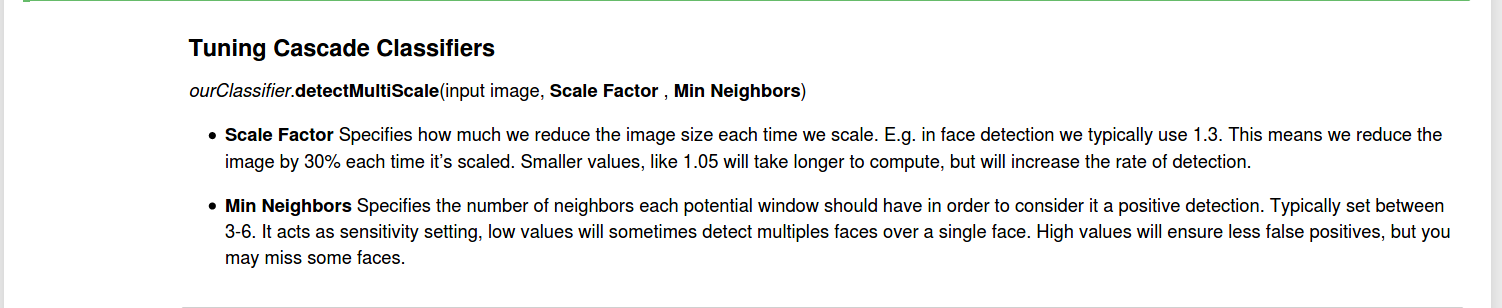




6. cap.release()

cv2.destroyAllWindows()





* Refernce: <https://github.com/rchavezj/OpenCV_Projects>
* Name: Ramah Hashem Madi.