

```
In [ ]: #Pre-trained models in opencv
>also known as haar(rectangle) cascade classifier.
>all models are trained using AdaBoost algorithm.
>all models are saved into xml format.
```

```
In [1]: import cv2
```

```
In [2]: model=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
```

```
In [15]: img=cv2.imread("f:/images/players/abc.jpg")
gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
faces=model.detectMultiScale(gray)
for x,y,w,h in faces: #[[ ],[ ],[ ]]
    cv2.rectangle(img,(x,y),(x+w,y+h),(0,0,255),2)
cv2.imshow("img",img)
cv2.waitKey()
cv2.destroyAllWindows()
```

```
In [28]: img=cv2.imread("f:/images/players/abc.jpg")
gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
faces=model.detectMultiScale(gray,minSize=(80,80),maxSize=(150,150),minNeighbors=5,scale
for x,y,w,h in faces:
    cv2.rectangle(img,(x,y),(x+w,y+h),(0,0,255),2)
cv2.imshow("img",img)
cv2.waitKey()
cv2.destroyAllWindows()
```

```
In [29]: import cv2
vdo=cv2.VideoCapture(0)
model=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
while True:
    isImg,img=vdo.read()
    if isImg==False:
        break
    gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
    faces=model.detectMultiScale(gray)
    for x,y,w,h in faces:
        cv2.rectangle(img,(x,y),(x+w,y+h),(0,0,255),2)
    cv2.imshow("img",img)
    key=cv2.waitKey(50)
    if key==ord('c'):
        break
cv2.destroyAllWindows()
vdo.release()
```

```
In [36]: img=cv2.imread("f:/images/players/virat.jpg")
gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
model=cv2.CascadeClassifier("haarcascade_eye.xml")
eyes=model.detectMultiScale(gray)
for x,y,w,h in eyes:
    cv2.rectangle(img,(x,y),(x+w,y+h),(255,255,255),1)
cv2.imshow("img",img)
cv2.waitKey()
cv2.destroyAllWindows()
```

```
In [41]: img=cv2.imread("f:/images/players/hardik.jpg")
gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
model_face=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
model_eye=cv2.CascadeClassifier("haarcascade_eye.xml")
faces=model_face.detectMultiScale(gray)
eyes=model_eye.detectMultiScale(gray)
```

```

for x,y,w,h in eyes:
    cv2.rectangle(img, (x,y), (x+w,y+h), (255,255,255), 1)

for x,y,w,h in faces:
    cv2.rectangle(img, (x,y), (x+w,y+h), (255,0,255), 2)

cv2.imshow("img",img)
cv2.waitKey()
cv2.destroyAllWindows()

```

```

In [42]: import cv2
vdo=cv2.VideoCapture(0)
model_face=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
model_eye=cv2.CascadeClassifier("haarcascade_eye.xml")
while True:
    isImg,img=vdo.read()
    if isImg==False:
        break
    gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
    faces=model_face.detectMultiScale(gray)
    eyes=model_eye.detectMultiScale(gray)
    for x,y,w,h in eyes:
        cv2.rectangle(img, (x,y), (x+w,y+h), (255,255,255), 1)

    for x,y,w,h in faces:
        cv2.rectangle(img, (x,y), (x+w,y+h), (255,0,255), 2)

    cv2.imshow("img",img)
    key=cv2.waitKey(50)
    if key==ord('c'):
        break
cv2.destroyAllWindows()
vdo.release()

```

In [ ]: