

# GÖRÜNTÜDEKİ YÜZLERİN TESPİTİ

Drive bağlanma:

```
from google.colab import drive
drive.mount('/content/gdrive')
```

Mounted at /content/gdrive

Kütüphaneleri yükleme:

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
```

Resmi okuma:

```
resim = cv2.imread('/content/gdrive/MyDrive/kodluyoruz/yuz.jpeg')
```

Resmi ekrana basma:



Resmimiz renkli olsaydı, siyah beyaza çevirmemiz gerekiyordu. Çünkü Cascade'ler siyah beyaz resme uygulanır. Resimde bazı yüzler bize doğrudan bakıyorken bazıları açılı bakıyor. Bazılarının gözleri tam olarak seçilebilirken, bazıları görünmeyecek kadar kısık ya da karanlık.

Amacımız bu 8 yüzü ve bu yüzlere ait gözleri tespit etmek.

Uygulamada yüz ve beden özelliklerini sınıflandırma amacı ile oluşturulmuş haar cascade sınıflandırıcıları kullanma:

Yüz için:

```
[5] yuz_cascade = cv2.CascadeClassifier("/content/gdrive/MyDrive/kodluyoruz/haarcascade_frontalface_default.xml")
```

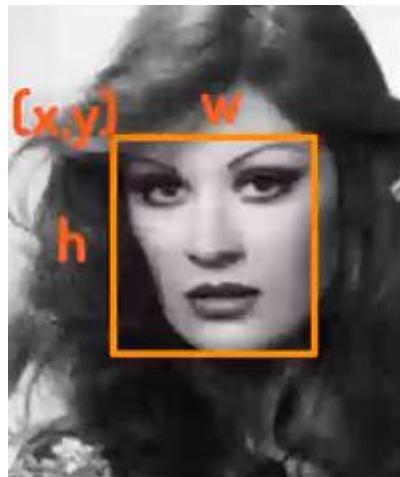
Göz için:

```
goz_cascade = cv2.CascadeClassifier("/content/gdrive/MyDrive/kodluyoruz/haarcascade_eye.xml")
```

Yüz cascade sınıflandırıcı ile resim üzerindeki yüzleri algılayarak koordinatlarını kaydetme: Tespit edilen yüzlerin etrafına çizilen dikdörtgenlerin koordinatlar belirleme ve yüzler birden fazla olduğu için for döngüsü kullanma.

```
[7] yuzler = yuz_cascade.detectMultiScale(resim, scaleFactor=1.075, minNeighbors=5, minSize=(15,15))
```

Yüzlerin tespiti için yuz\_cascade sınıflandırıcıyı detectMultiScale fonksiyonu ile çalıştırıldı. Bu sınıflandırıcı tüm resimde kare kare gezerek yüz arıyor. Bulduğunda da detectMultiScale fonksiyonu sayesinde etrafına çizilen karenin koordinatlarını yüzler isimli değişkende saklıyor. detectMultiScale fonksiyonu çalıştığında 4 tane değer geri döndürüyor. x, y, w ve h değerleri. Bu değerler bulunan yüzün etrafına çizmek istenilen dikdörtgenin koordinatlarına erişmeyi sağlıyor. x ve y dikdörtgenin sol üst başlangıç köşesinin koordinatları, w ve h ise dikdörtgenin yüksekliği ve genişliği.



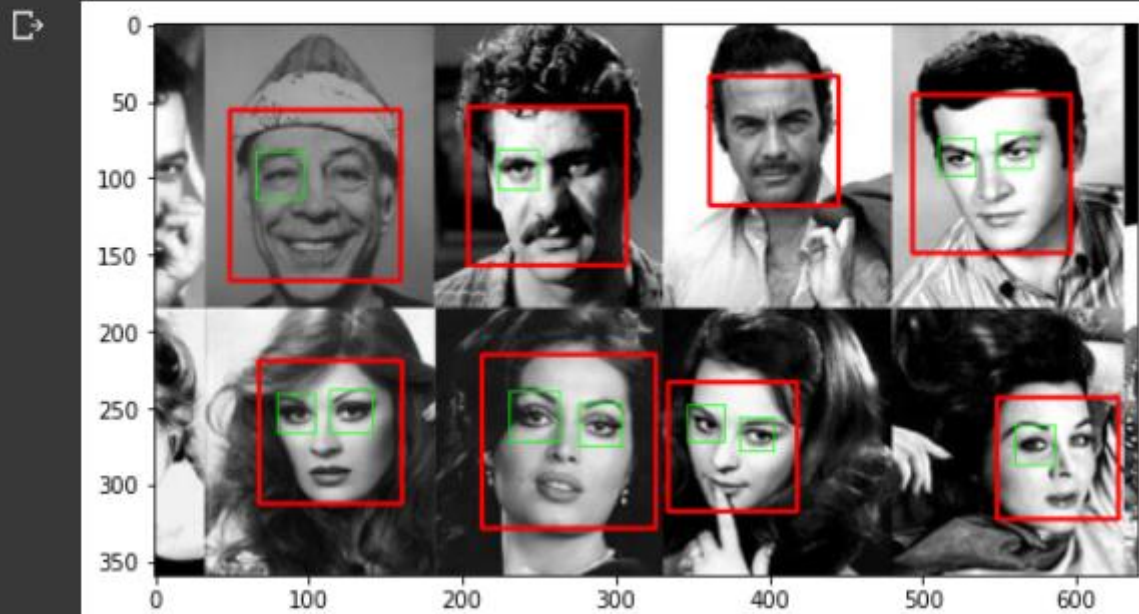
Yüz tespiti yaptığımız dikdörtgenlerin içerisinde göz bulma: yüz bulmada yapılan işlemler göz bulma içinde yapılır.

```
[8] for(x,y,w,h) in yuzler:
    cv2.rectangle(resim, (x,y),(x+w,y+h), (255,0,0),2 )
    roi = resim[y:y+h, x:x+w]
    gozler = goz_cascade.detectMultiScale(roi)
    for(ex,ey,ew,eh) in gozler:
        cv2.rectangle(roi, (ex,ey), (ex+ew, ey+eh), (0,255,0),1)

print("Yüz Bulundu! Bulunan Yüz Sayısı = ",format(len(yuzler)))
```

Yüz Bulundu! Bulunan Yüz Sayısı = 8

```
plt.figure(figsize=(8,12))
plt.imshow(resim)
plt.show()
```



## KAYNAKÇA

Bilgeiş “Herkes için Yapay Zekâ II” eğitimi.