

BALIKESİR ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ



**Görüntü İşleme Yöntemi İle Suçlu Tespiti ve
Uyarı Sistemi**

BİTİRME PROJESİ
RAPOR-1

Dr. Öğr. Üyesi: Gültekin KUVAT

Ad Soyad: Gizem SOLMAZ

Numara: 201613709028

Bilgisayar Mühendisliği Tasarımı

05.11.2021

ÖZET

Kaçak suçluların ilgili ekipler tarafından bulunmaları uzun sürebiliyor. Hedefim suçluların bulundukları şehri ya da ülkeyi terk etme durumlarında ulaşımı sağlayabilmek için tren garları, otobüs terminalleri, havaalanları veya iskeleler, aynı şekilde kaçak suçlunun gidebilme ihtimalinin yüksek olduğu market zincirler, büfeler, petrol ofisleri, otel gibi bir çok yerin kameralarından suçlunun tespitiyle birlikte ilgili birimlere bildiri gönderilmesidir.

GİRİŞ

Projemi yapma amacım kaçak suçluların günümüz şartlarında bile haftalar, aylar gibi uzun zaman zarflarında bulunmasına hatta kimi zaman hiç bulunamama durumlarına son vermektir. Kaçan bir suçlunun bulunduğu şehri hatta ülkeyi terk edebilmek için otobüs, uçak gibi yolculuk biletlerine erişim sağlayabileceği mekanların kameralarıyla suçlu tespiti yapılabilir. Suçlu tespit edildiği takdirde ise ilgili birimlere bildiri gönderilir. Suçlu kaçmak yerine bulunduğu yerde saklanmak istediği takdirde ise en temel ihtiyaçlarını karşılamak için yani yemek, barınma gibi ihtiyaçlarını karşılayabilmek için market zincirlerde, büfe, geceleri alışverişe el verişli olan petrol ofisleri, barınmak için ise lüks oteller yerine daha orta seviyeli dikkat çekmeyecek otel kameralarından da suçlunun tespitini rahatça sağlayıp, tespit edildiği takdirde ilgili birimlerimize bildiri gönderilip kaçak suçlunun yer tespiti yapılır ve yakalanması sağlanır.

PROJEM İÇİN İZLEYECEĞİM YOL

Yüz tespit için etkili olabileğini düşündüğüm 2 yol buldum. Bunlardan birincisi **haar cascades** bir diğeri ise **face_recognition**.

Haar cascades ve face_recognition görüntüde ki yüzlerin tespiti için kullanılabilir fakat haar cascades kullanarak kimlik tespiti yani yüzün kime ait olduğunu bulma işlemini gerçekleştiremeyeceğimden dolayı yoluma face_recognition ile devam edeceğim. Peki neden face_recognition kütüphanesi neden tercih ediyorum. Face_recognition görüntüde ki yüzleri tespit etme ve yüzün kime ait olduğunu bulmayı sağlayan bir kütüphanedir. Dlib geliştirilerek ortaya çıkmıştır. Kameradan gelen görüntü ya da dosya içerisinde ki bir yüzü tespit eder ve yüzün önemli noktalarını yani kaşların konumu, göz aralığı, dudağın konumu gibi bilgilere erişerek yüz kime ait olduğunu bulur.

Çalıştığım makinenin terminal kısmına “pip install face-recognition” yazarak kütüphanemi ekliyorum. Bundan sonrasında ise çalışacağım sayfada “import face-recognition” yazarak projeme dahil ediyorum.

Daha sonrasında dosya içerisinde 2 farklı resim karşılaştırması yaparak yazdığım kodun doğruluğunu test edeceğim. Resimlerde ki yüzleri tespit edip yüz özelliklerini kıyaslayarak True ya da False değeri döndürerek tespiti sağlayacağım. Sonraki aşamada bu işlemi kameraya aktarıp doğru sonuç elde etmeye çalışacağım.

İLK AŞAMA İÇİ YAZMIŞ OLDUĞUM KOD PARÇALARI

```
simple_facer.py
1  import cv2 as cv
2  import face_recognition
```

face-recognition' çalıştığım sayfaya ekledim.

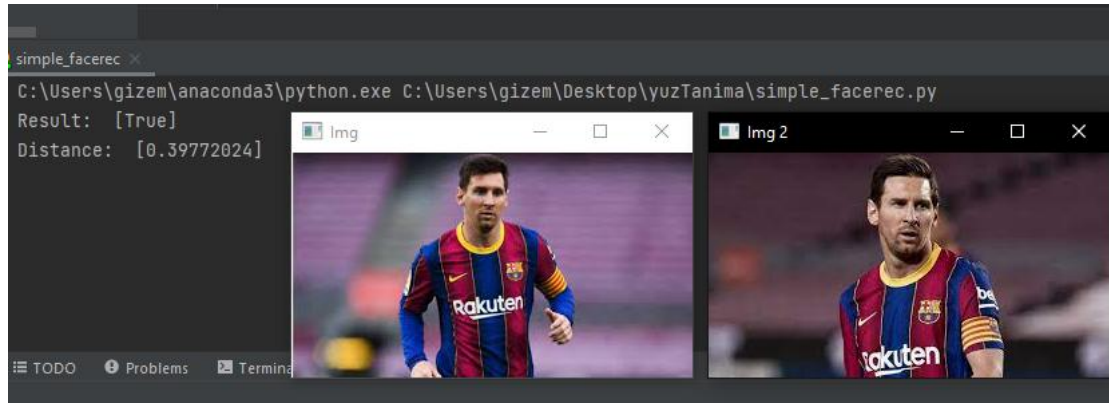
```
resim1 = cv.imread("messi.jpg")
rbg_resim1 = cv.cvtColor(resim1, cv.COLOR_BGR2RGB)
img_encoding = face_recognition.face_encodings(rbg_resim1)[0]

resim2 = cv.imread("images/messi.jpg")
rbg_resim2 = cv.cvtColor(resim2, cv.COLOR_BGR2RGB)
img_encoding2 = face_recognition.face_encodings(rbg_resim2)[0]
```

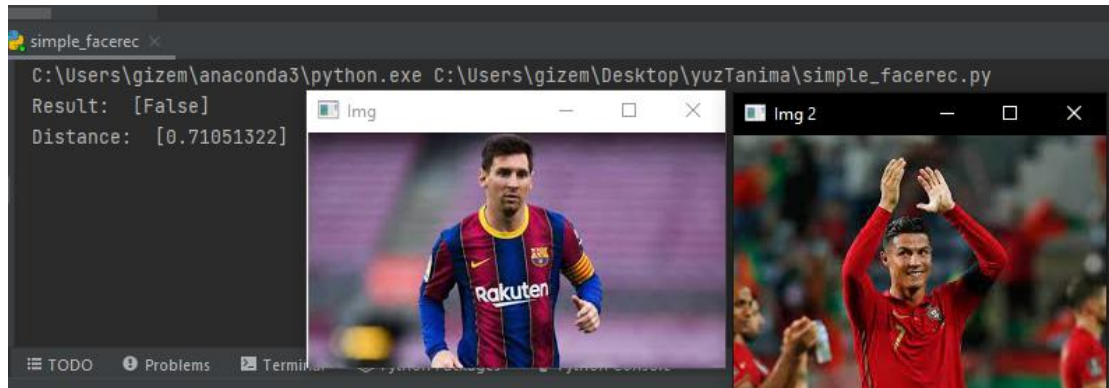
Çalıştığım dosya içerisine tanımlamış olduğum 2 resmi tanımladım, bu resimlerde ki yüzlerin ayrı ayrı özelliklerini tespit etmiş oldum.

```
sonuc=face_recognition.compare_faces([img_encoding],img_encoding2)
pay=face_recognition.face_distance([img_encoding],img_encoding2)
print("Result: ", sonuc)
print("Distance: ",pay)
```

Tespit edilen özellikleri birbirleriyle kıyaslayarak benzerlik oranı çıkardım. Daha sonrasında print ile benzerlik olup olmadığını True ya da False değeri ile ve benzerliğin oranını konsol ekranına yazdırdım.



Seçmiş olduğum iki resim arasında ki benzerlik True değeri olarak dönmüş oldu.



Kişiler farklı olduğu zaman ise False değerine ulaşmış oluyorum.

KAYNAKÇA

https://google.github.io/mediapipe/solutions/face_detection.html