**POLİS İÇİN GÖRÜNTÜ İŞLEME TABANLI YARDIMCI ASİSTAN**

**Giriş:**

Bu proje polislerin GBT kontrolleri sırasında daha hızlı işlem yapabilmeleri için görüntü işleme yardımıyla bu sorunu çözebilmek. Görüntü işleme için kullanılacak olan kamera raspberry Pi bağlı olup alınan görüntülerin burada işlenip sonuç elde ederek polise gerekli bilgileri aktarabilecektir.

Projede raspbery e bağlı bir kamera olacaktır. Kameranın kullanım yeri ise toplu taşıma araçları, yoğun caddeler ve güvenliğin önem kazandığı herhangi bir yerde kullanılmak üzere amaçlanabilir. Bu kameralar insan yüzlerine odaklanıp buradaki görüntüyü bağlı olduğu raspberrye aktaracaktır. Bu aktarım cadde sokaklarda güvenlik kameralarından da olabilir GBT kontrolü yapacak olan herhangi bir polis üzerindeki mini kameradan da olabilir.Raspberrynin veri tabanında olan belli başlı isimleri Raspberryye aktarılan görüntülerle karşılaştırlıp bir suç teşkil edip etmediği tespit edilir. Sonuçlanan bu bilgi polise aktarılır.

**Gerekli Donanım Bileşenleri:**

**1.** 1 adet Rasperry pi Model B .

**2.** 1 adet WebCamera.

3. 1 adet bellek kartı.

4. Telefon raspberry kontrol etmek için .



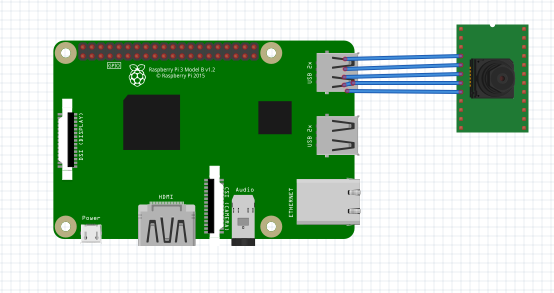
**Gerekli Yazılım Bileşenleri**

1. Python(<https://www.python.org/> )
2. openCV (<https://opencv.org/> )
3. Tensorflow (<https://www.tensorflow.org/> )

**Kullanılan Bileşenlerin Özellikleri:**

1. Rasperry pi 3;[Raspberry Pi](https://www.robotistan.com/raspberry-pi-3-model-b-plus), kredi kartı büyüklüğünde bir bilgisayardır.Günümüze kadar birçok modeli ortaya çıkmıştır. Bizim bu brojede kullandığımız model B dir. Ürünü şuradaki adresten temin edebilirsiniz. (<https://www.direnc.net/> )
2. Web Camera; Bu ürün şuradan temin ettik vs. ([www.urunadi.com/webcamera.htm](http://www.urunadi.com/webcamera.htm))

**Şematik Çizimi:**



Resim 1 : Şematik Çizimi

**Yapım Aşamaları**

**Raspbian kurulumu :**

* Raspbian kurulumu için [**https://www.raspberrypi.org/downloads/raspbian/**](https://www.raspberrypi.org/downloads/raspbian/) adresini ziyaret ediyoruz.
* İndirdiğimiz raspbian dağıtımını bir yere açıyoruz.
* Daha sonra [buradan](https://launchpad.net/win32-image-writer) sağ kısımdan Win32DiskImager’ı (binary) indiriyoruz
* Açılan sayfadan imajı seçip sağ kısımdan SD Kartın bölümünü seçiyoruz ve write’a basıyoruz (Not: Kartın içi sıfırlanıyor).
* Bir süre bekledikten sonra karta yükleme tamamlanıyor, ondan sonra kartı sökebiliriz.
* **Güç kaynağı** olarak 5Volt 1 Amper istemekte Raspberry Pi.

**openCV kurulumu:**

OpenCV’yi  [**http://opencv.org/downloads.html**](https://opencv.org/releases.html) adresine giriyoruz ve indirmek istediğimiz sürümün altındaki **OpenCVfor Windows** linkine tıklıyoruz.İndirme bağlantısı sourceforge.net sitesine yönlendirecek ve indirme işlemi başlayacak.İndirdiğinizde sıkıştırılmış olarak gelecektir, çalıştırdığınızda OpenCV dosyalarını çıkartmak için bir dizin isteyecektir, burada çıkartılmasını istediğiniz dosya dizini yolunu yazarak **Extract** butonuna tıklayın. Dosyaları çıkarttığınız dizinde OpenCV klasörü içerisinde **build** ve **sources**diye 2 adet klasör bulunmaktadır. Build klasörü içerisinde Windows platformu için derlenmiş olarak sistem native kütüphaneler ve programlama dilleri için kütüphaneler bulunmaktadır.  Sources klasöründe ise OpenCV kaynak kodları ve örnek uygulamalar yer almaktadır. Buradaki kaynak kodlar ile OpenCV’yi tekrardan derleyebilirsiniz.Java ile OpenCV uygulaması geliştirmek için **build** içerisindeki **java** klasöründe yer alan jar dosyasını ve kullanacağımız işletim sistemi mimarisine göre OpenCV Windows sistem kütüphanelerini kullanacağız.

**Tensorflow kurulumu:**

Deeplearning hizmeti sunar.

Tensorflow Windows’ta 2 yükleme yöntemi sunuyor. Birincisi **pip** ile yükleme, ikincisi **Anaconda** içinde yükleme.

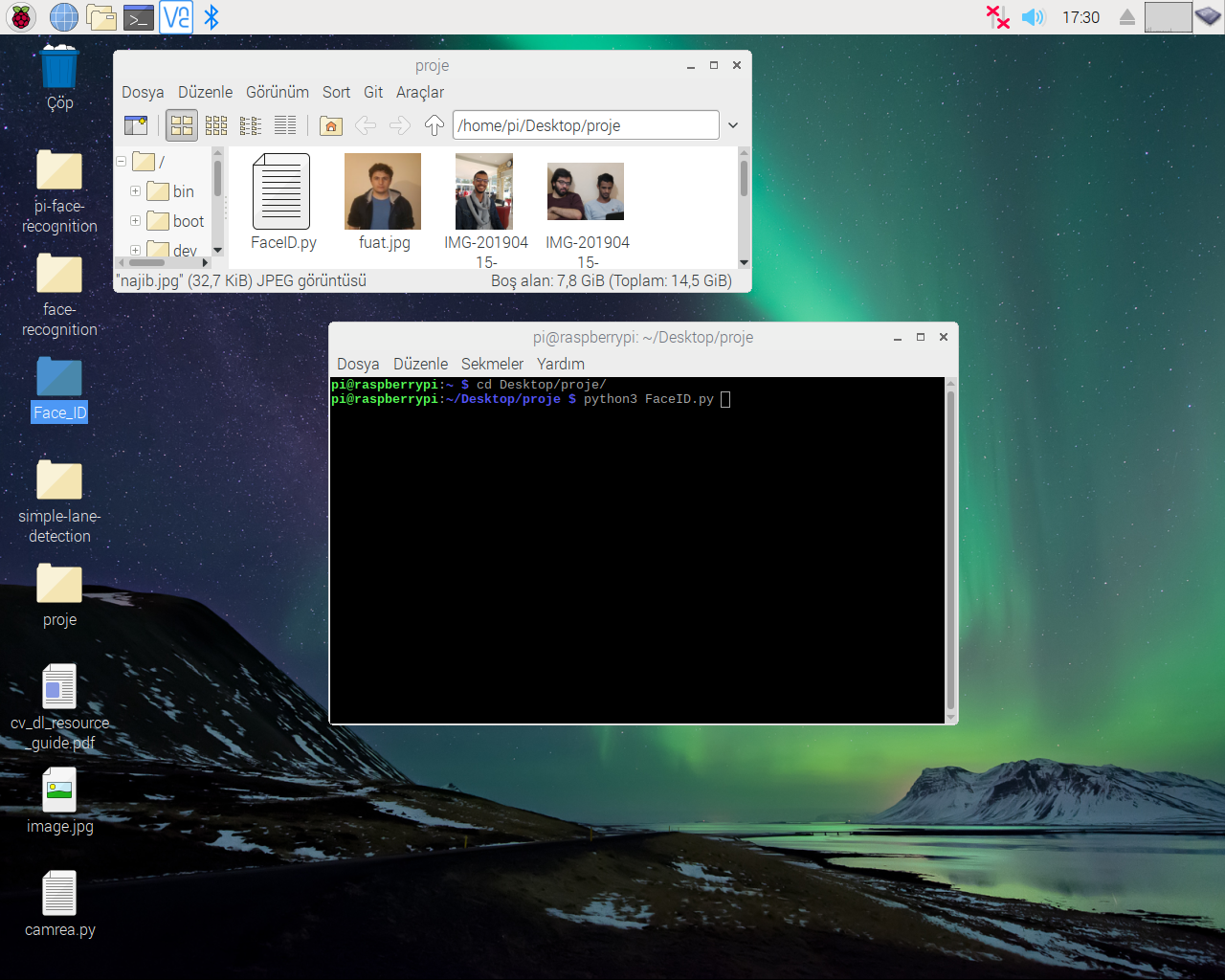
Tensorflowpip ile yüklendiğinde sanal bir ortam oluşturulmadan yüklenir, bu nedenle diğer Python modülleriyle karışıp diğer işlerinizin çalışmasını engelleyebilirler. Bu yüzden Anaconda ile yüklemek daha güvenli bir yöntem.

Ancak, Tensorflow Windows’ta şu an için Python 3.6 versiyonunu desteklemiyor. Bu yüzden Tensorflow’da sanal bir ortam oluşturmak, bunu da **Python 3.5** versiyonuyla yapmak gerekiyor.

NVIDIA ekran kartı varsa Tensorflow’unGPU’lu versiyonu yüklenebilir, yoksa sadece CPU üzerinde çalışan versiyon yüklenebilir.

**Yapım Aşamaları:**

1. Aşağıdaki cd Desktop/proje2/ komut yardımıyla projenin yeri ulaştık . Sonra python3 FaceID.py komutuyla Proje çalıştırdık.

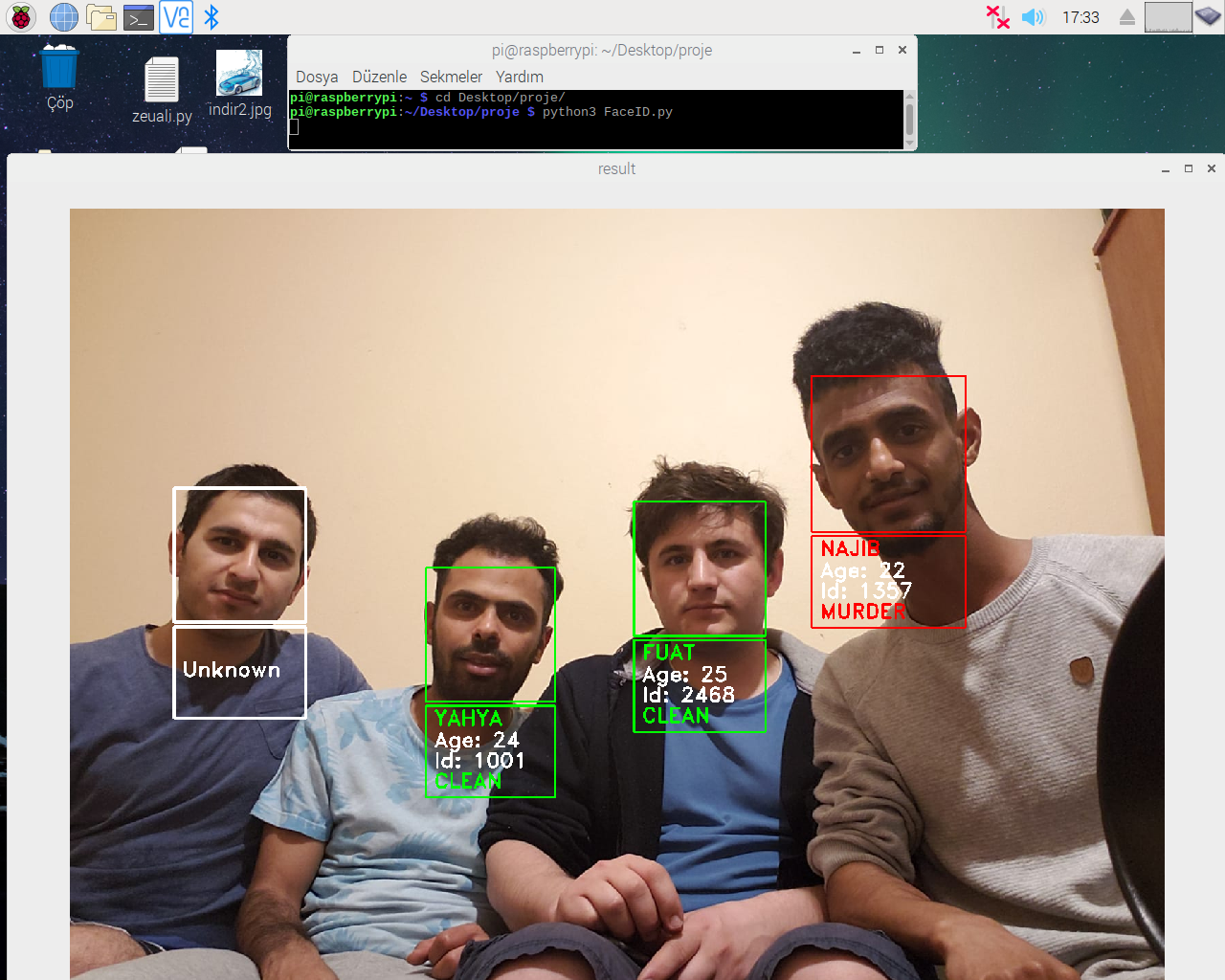


Proje çalıştığı zaman kamera çalışıyor ve aşağıdaki görüntü gibi bir sonuç elde edilmektedir.

Kırmızı ise suçlu kişidir.

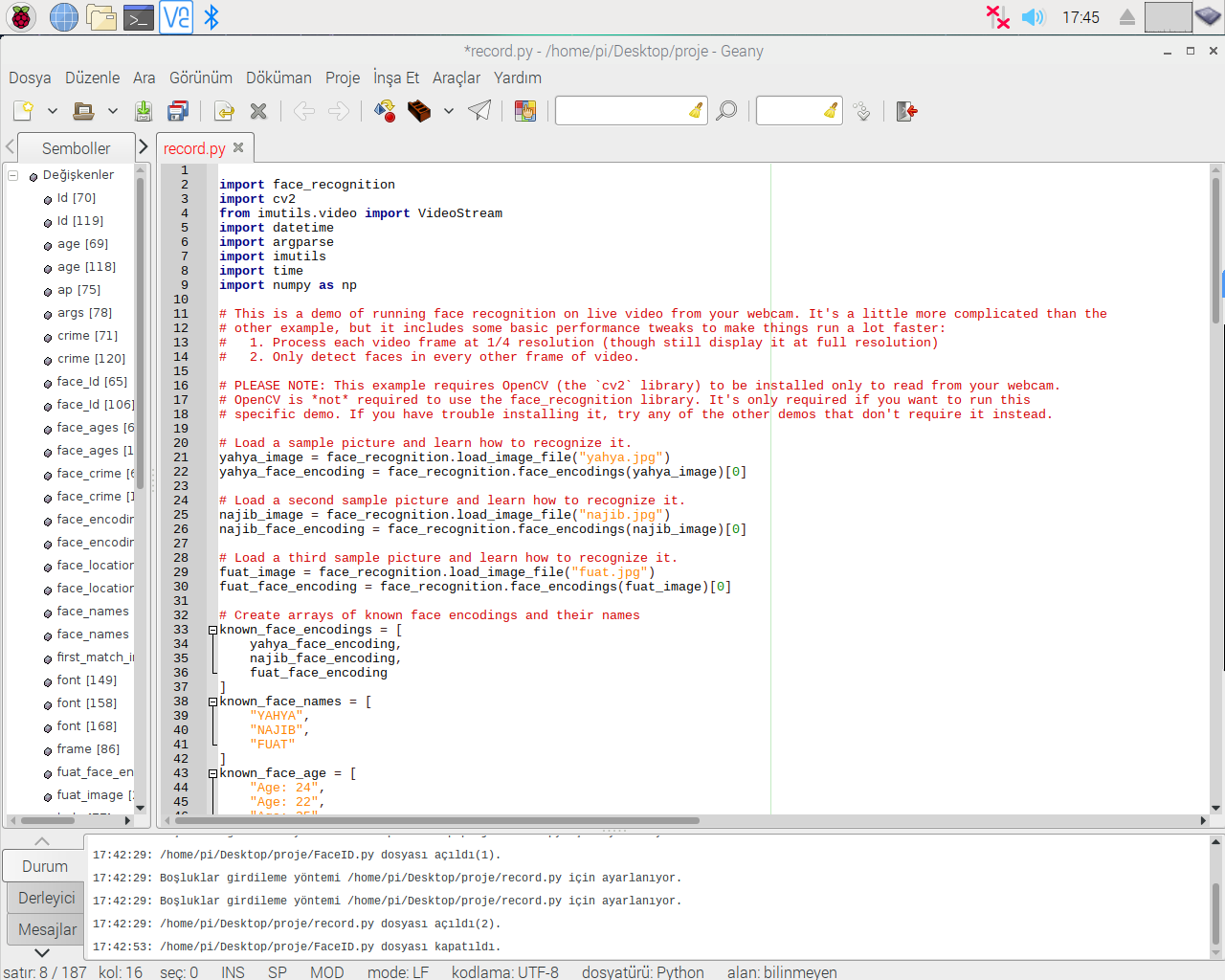
Yeşil ise temiz kişi demek.

Beyaz ise kayıtlı olmayan kişi.

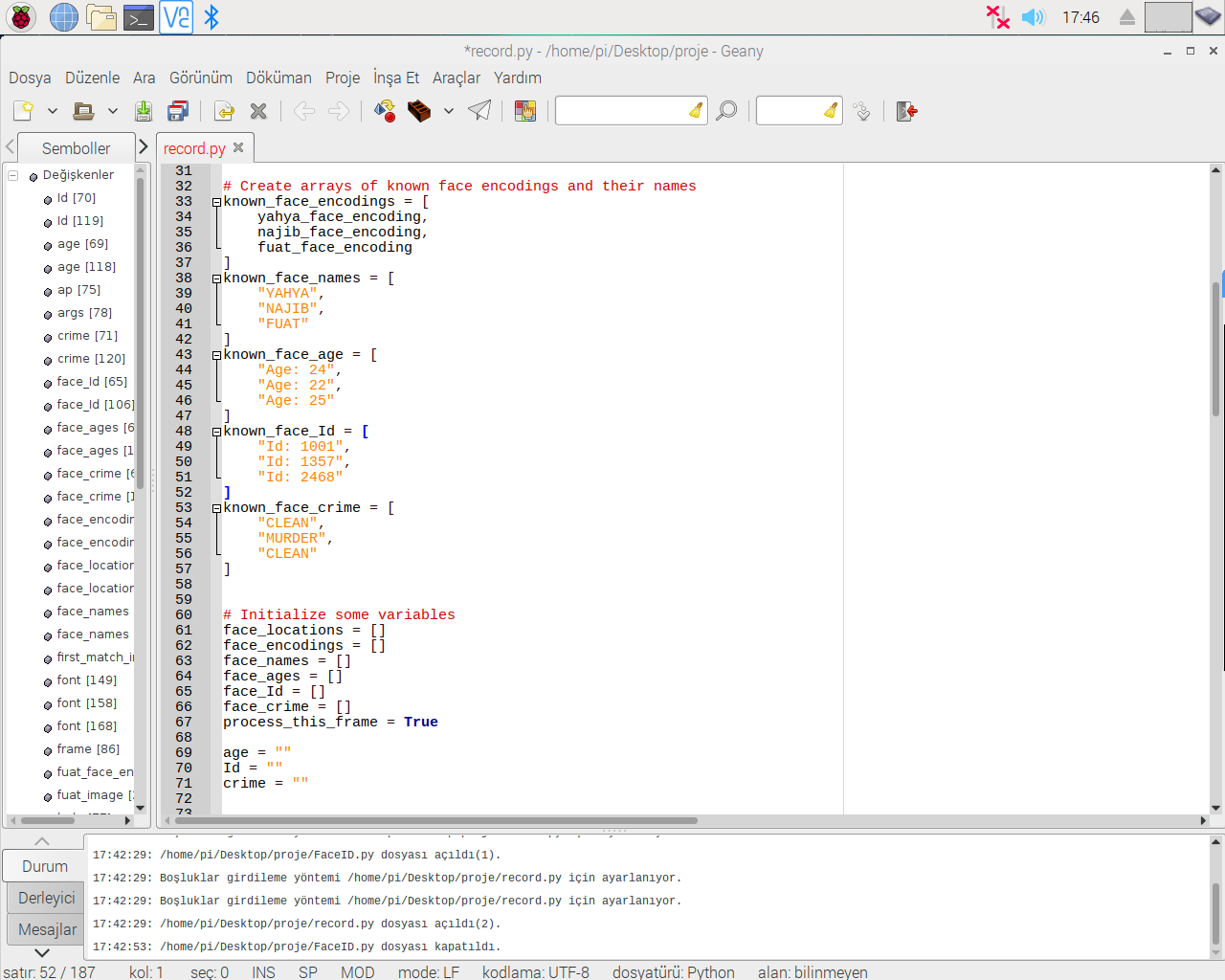


**projenin kodları**

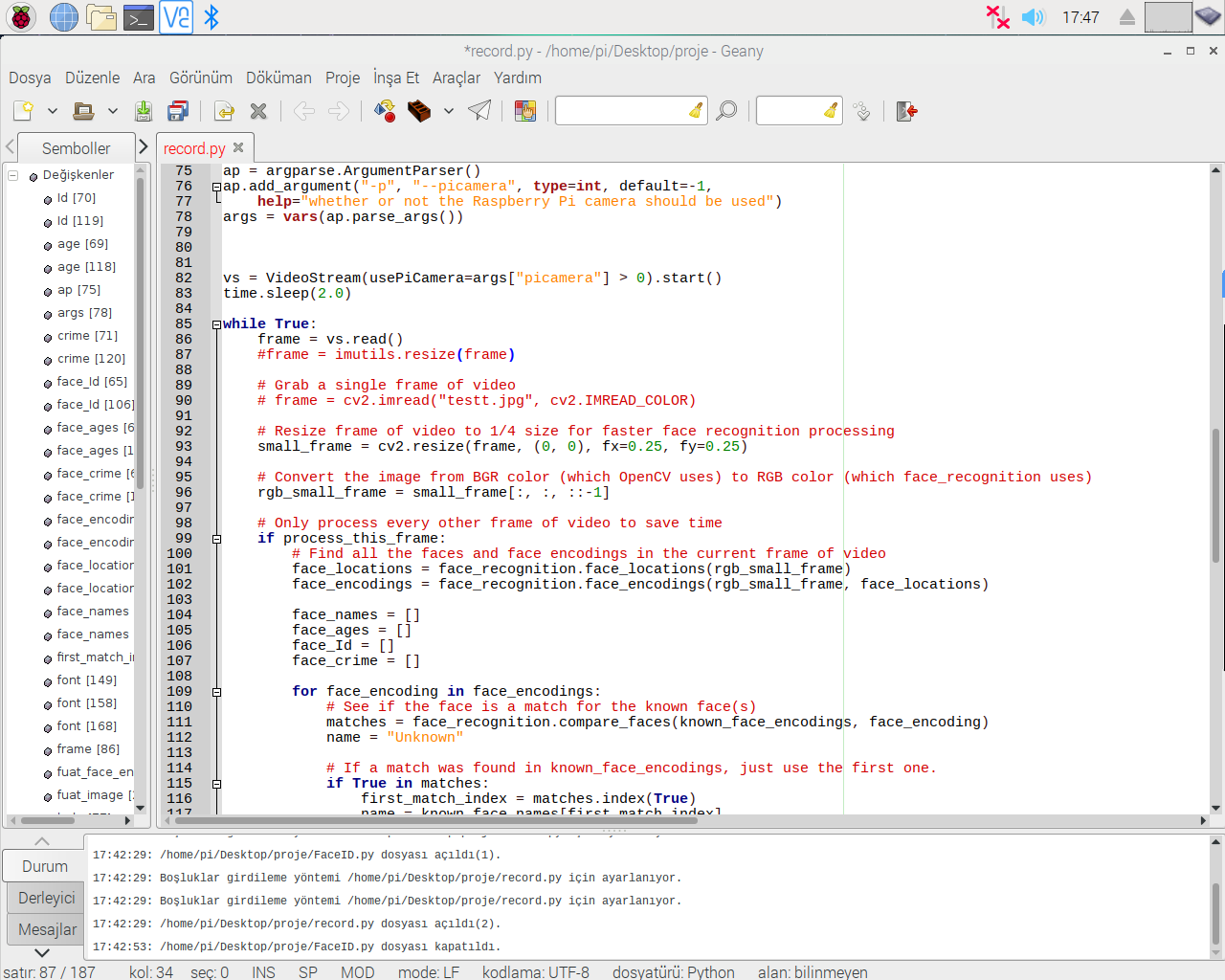
Kullanılan kütüphaneleri çağırdık .

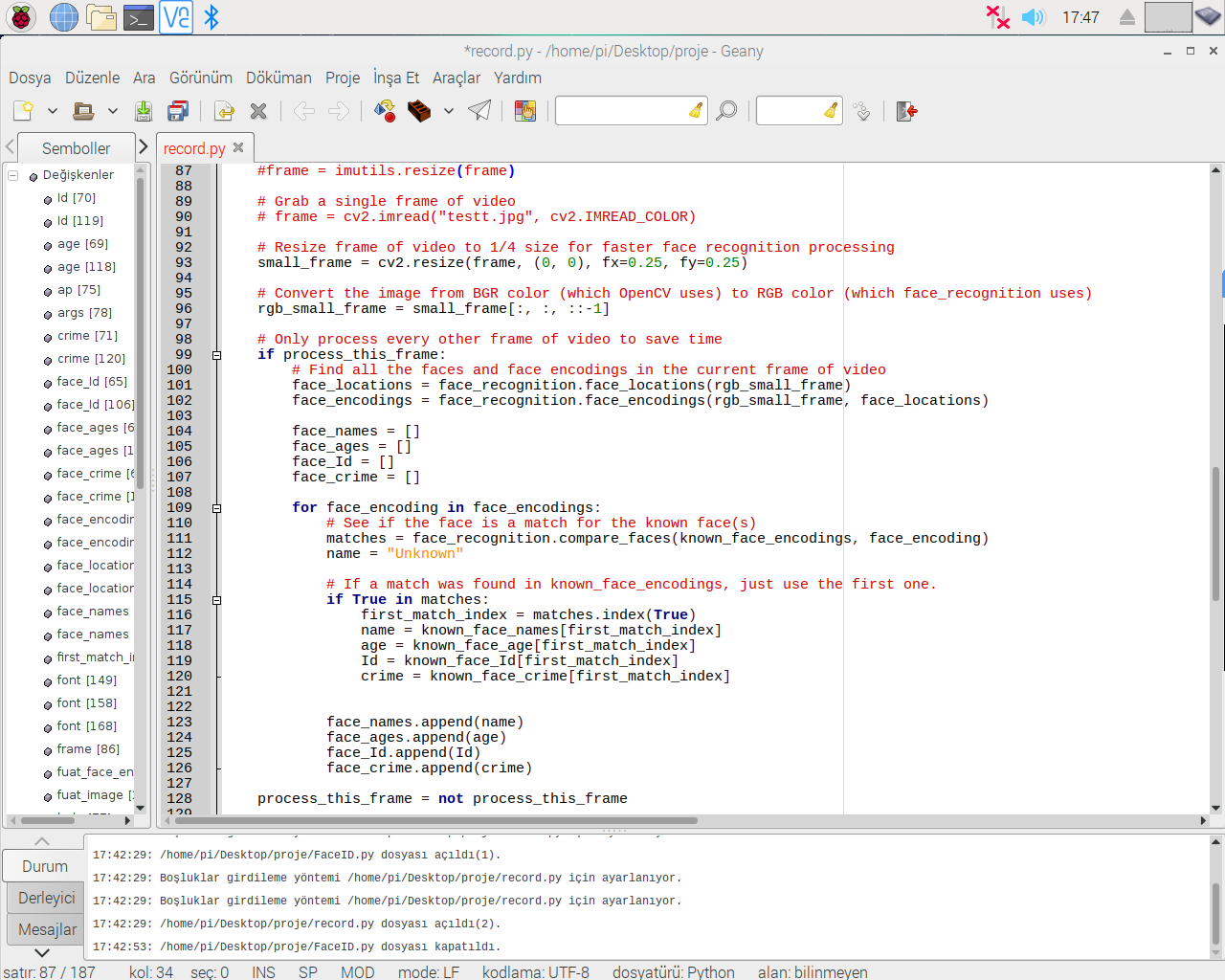


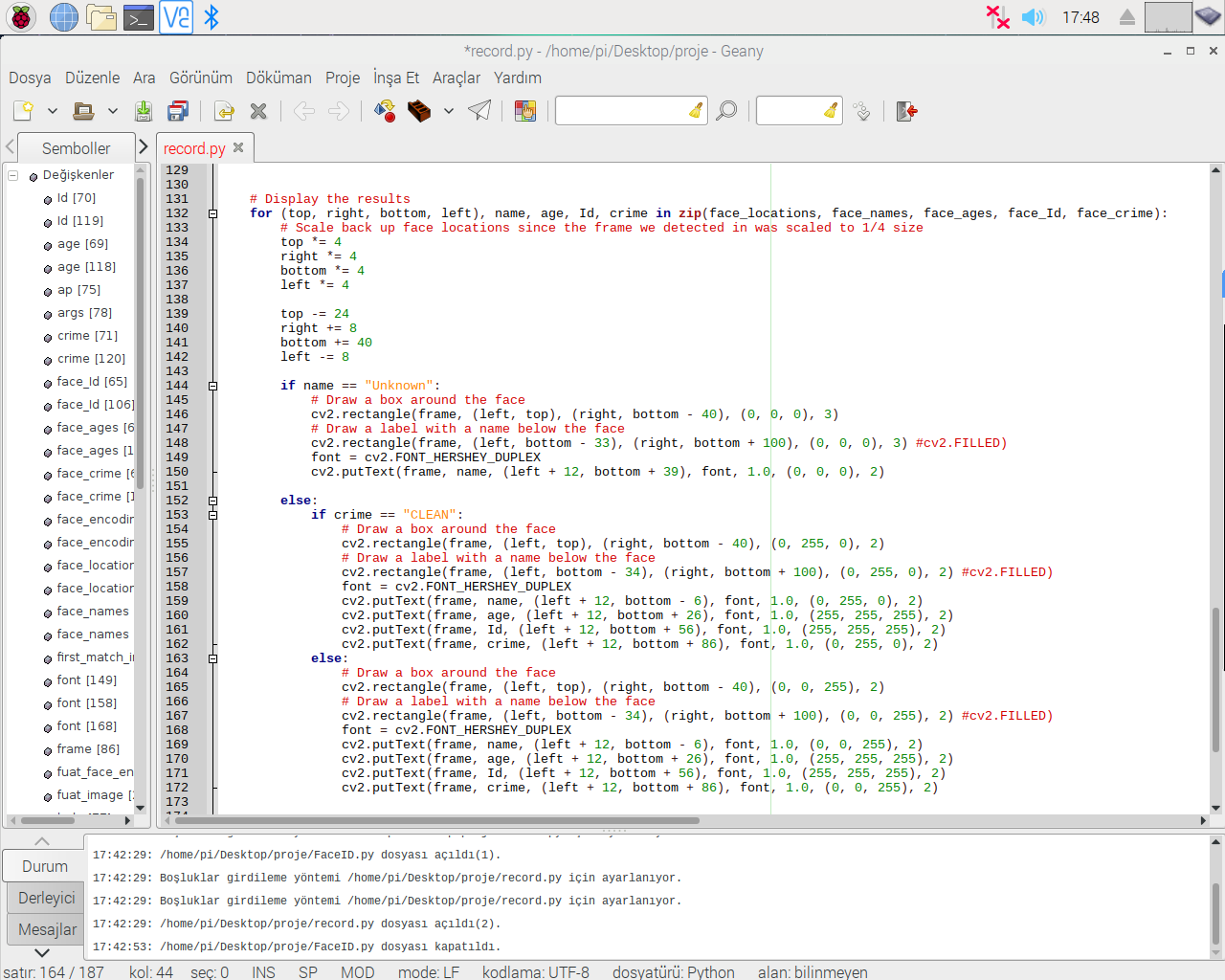
GBT genel bilgeler tabanıya kişilerinin bilgilerini kaydediyoruz.

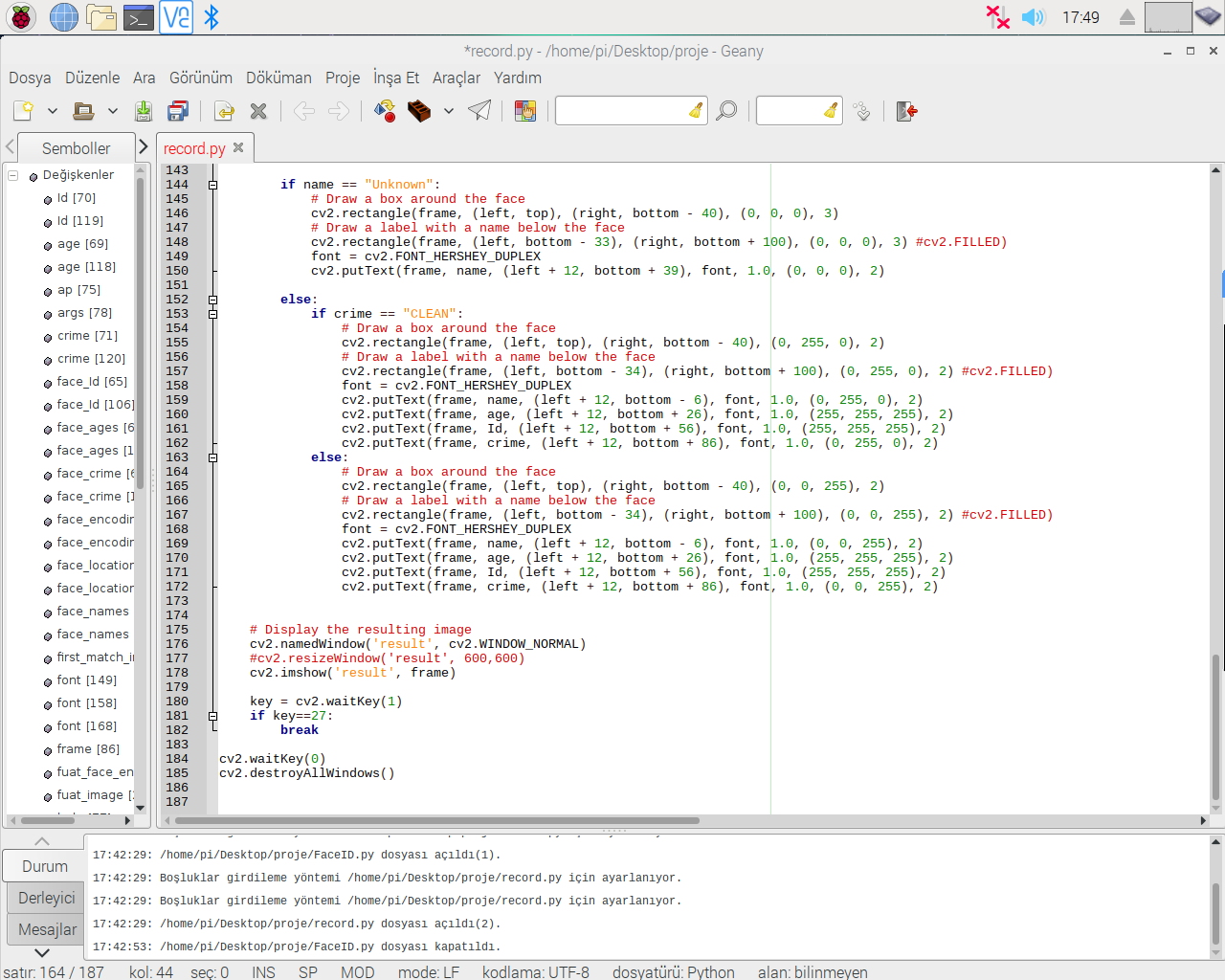


Görüntü işlemi ile yüz tanıma işlemi yapılacaktır.









**Kaynak Kodu :**

Buradaki proje resimlerine, videolarına (kısa bir video koyunuz) ve kaynak koduna <https://github.com/hyuce> adresinden erişilebilir.

**Nasıl Kullanılır**

Bu projenin çalıştırılması oldukça basittir. İlk olarak raspberry bağlı kamera ile görüntü elde edilir. Kamera ile alınan bu görüntülerraspberryninveritabanındaki görüntülerle belli bir algoritamaya göre karşılaştırılır ve bir sonuç elde edilir. Bu sonuç kullanıcıya bir arayüz aracılığıyla sunulur.

**Proje Resimleri :**

****

**Öneriler**

Projemizde kameranın aldığı görüntüler raspberryyesunulumaktadır bunun bir adım ötdesi olarak fotoğraf yerine video şeklinde olan canlı görüntüler raspberrye aktarılır böylece çok daha fazla insan taranmış olur

Video Linki:

YoutubeVideo Kaydını <https://www.youtube.com/watch?v=GQrv7FmMQB&feature=youtu.be> adresinden görüntüleyebilirsiniz.