***RASPBERRY PI 3 İLE OPEN CV VE PYTHON PROGRAMLAMA DİLİNİ KULLANARAK YÜZ TANIMA SİSTEMİ***

***MUHSİN AYDIN-460…***

***MEHMET ALİ SONBAHAR-460…***

***FURKAN KOÇ-460115017***

***FURKAN ŞANLI-460…***

***RAİF TELLİ-460…***

***PROJE SAHİPLERİ:***

******



***Muhsin AYDIN Mehmet Ali SONBAHAR***



***Furkan KOÇ Furkan ŞANLI Raif TELLİ***



***RASPBERRY PI 3 NEDİR?***

**Raspberry Pi kredi kartı boyutunda bir bilgisayardır. Televizyonunuza bağlayıp görüntü alabilir, bir klavye bağlayabilirsiniz. Yetenekli küçük bir bilgisayar diye adlandırdığımız Raspberry Pi ile normal masaüstü bilgisayarlarda yaptığınız işleri örneğin, sözcük işlemciler ve hesap programları (Word, Excel) ile çalışabilir çeşitli oyunlar oynayabilirsiniz. Ayrıca yüksek çözünürlüklü HD videolar oynatabilirsiniz. Ayrıca tüm dünyada çocukların alıp kullanabileceği, basit programlama yapabilecekleri hatta deneylerinde kullanabileceği uygun fiyatlı bir bilgisayar gibi düşünebilirsiniz.**

***RASPBERRY PI 3 TEKNİK ÖZELLİKLERİ:***

* **64-bit quad-core ARMV8 işlemci**
* **1.2GHz**
* **1GB RAM**
* **Dahili WiFi - BCM43143**
* **Bluetooth 4.1 (Bluetooth Low Energy - BLE)**
* **40 Adet GPIO**
* **4 Adet USB 2**
* **4 uçlu Stereo çıkışı ve Composite video çıkışı**
* **Full HDMI**
* **Raspberry Pi Kamera bağlanstını için CSI  kamera portu**
* **Raspberry Pi 7" dokunmatik ekran için DSI ekran portu**
* **Micro SD soketi**
* **Güncellenmiş güç katı (2,5A'e kadar destekliyor.)**
* **Güç ve aksiyon ledi.**

***RASPBERRY PI 3 İÇİN İŞLETİM SİSTEMİ***

**Raspbian (Debian tabanlı) - Raspberry Pi resmi işletim sistemi**

**Raspberry Pi için hazırlanan Raspbian'ın son dağıtımı ise, Raspbian Jessie adını taşıyor. Resmi olarak her Raspberry Pi sürümü için güncellemeler servis ediliyor. Raspbian'ın Jessie kod adlı sürümü öncesi kullandığı sürüm adı ise, Raspbian Wheezy sürümüydü.**

***RASPBERRY PI 3 İLE YÜZ TANIMA SİSTEMİ***

**Yüz tanıma sistemi günümüzde bir çok büyük şirket tarafından birçok farklı uygulama ve programlarda çeşitli amaçlarla kullanılmaktadır. Bizim bu projedeki en temel amacımız kameradan alınan görüntüyü işleyebilmektir.**

**Open CV bu amaç için en kullanışlı yöntem olmakla birlikte görüntü tanımlamada bizlere çok imkan sağlamaktadır. Tanımlanan görüntü üzerinde renklerin ve yüzün seçilmesi işlemi PYTHON programlama dili ile kolay bir biçimde gerçekleştirilebilmektedir. Biz bu projeyi yaparken en çok zorlandığımız kısım stabil işletim sistemi bulma kısmıydı. İlk olarak kurduğumuz Ubuntu Mate işletim sisteminden yeterli randımanı alamadığımız için Raspbian Pixel işletim sistemine geçme kararı aldık. Umarız proje bizden sonra devam etmeyi düşünen arkadaşlara yardımcı olur.**

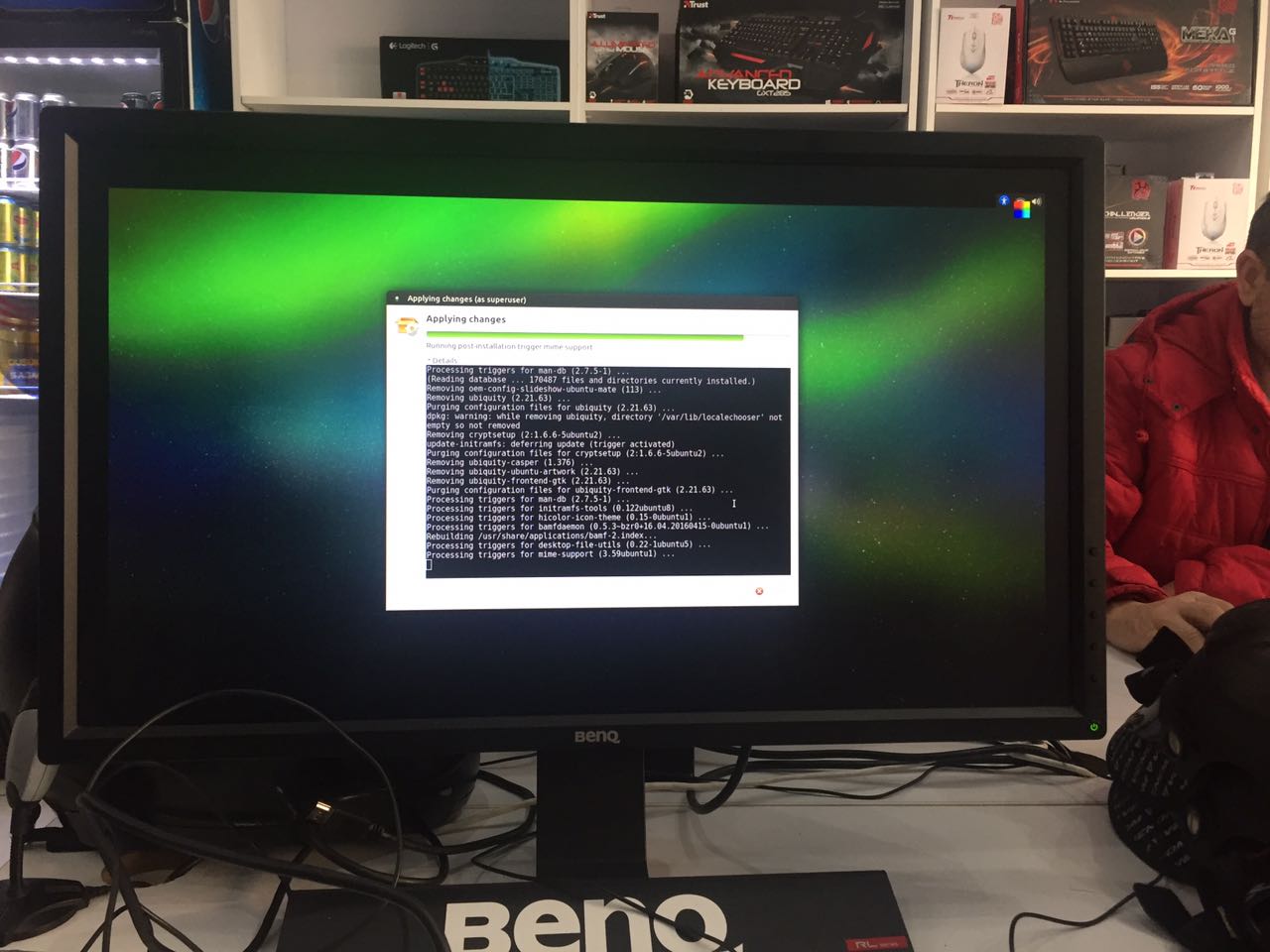
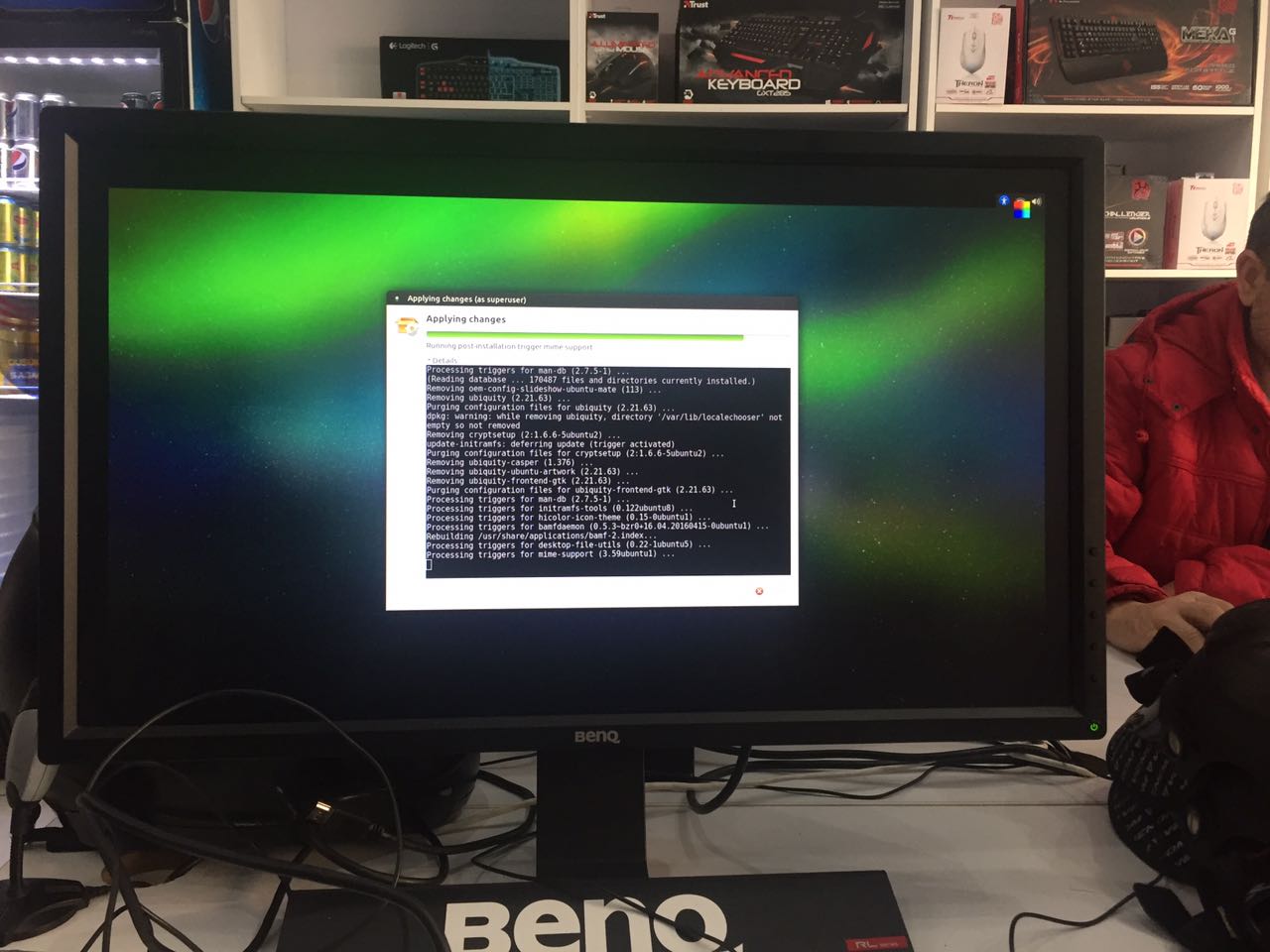
***KULLANILAN MALZEMELER:***

**-Raspberry Pi 3**

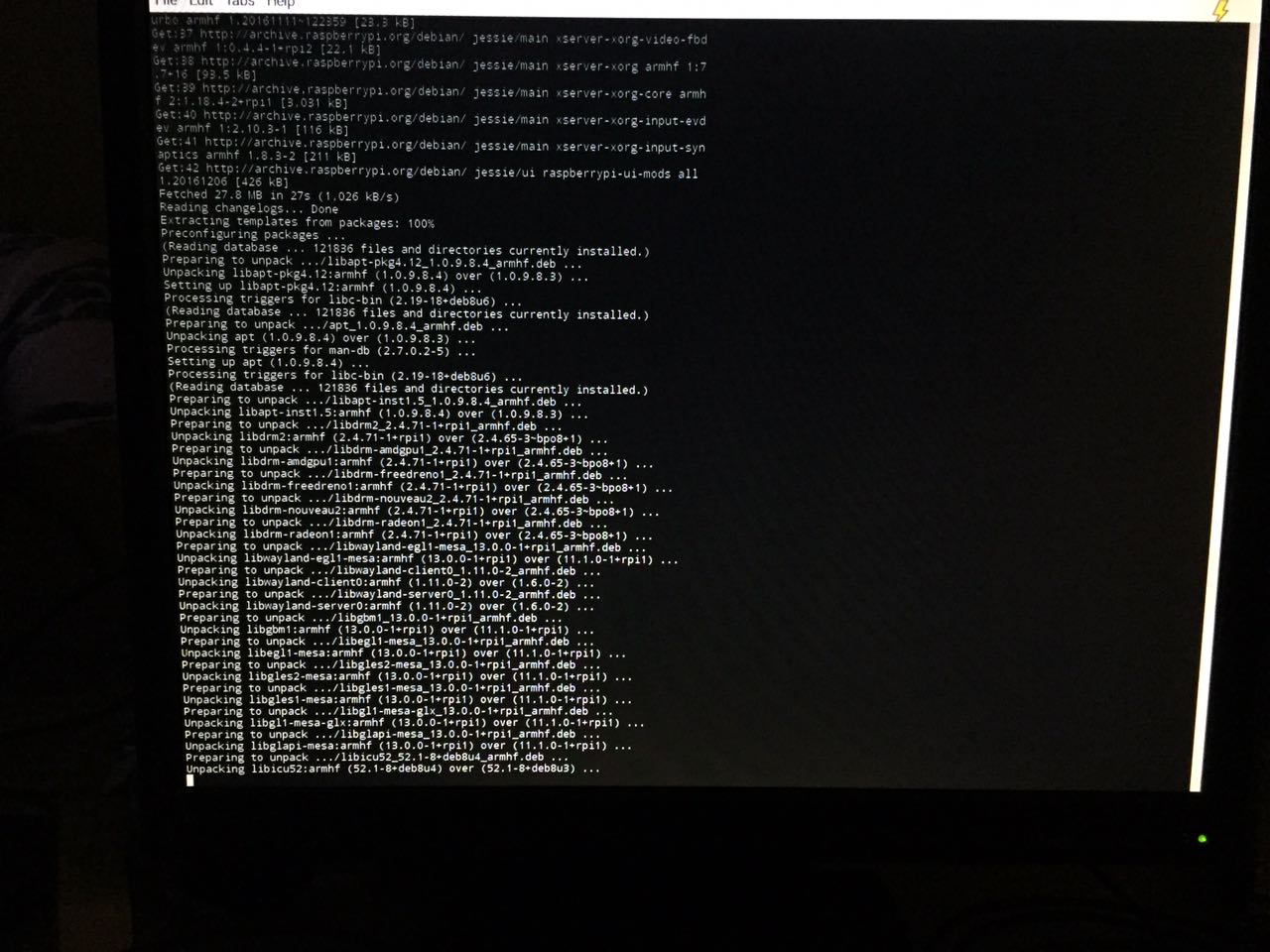
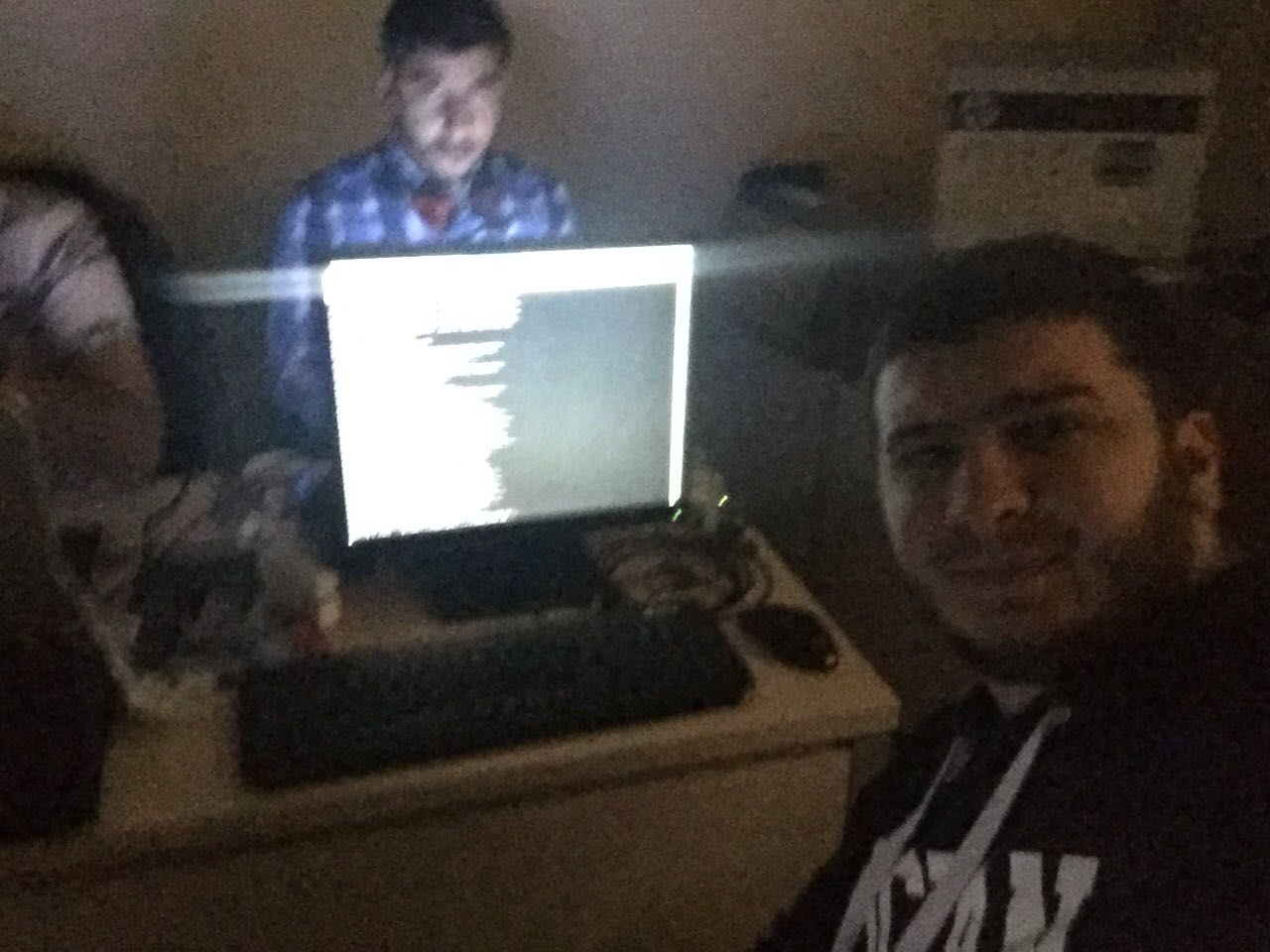
**-Usb Kamera**

***YÜZ TANIMA SİSTEMİ YAPIM AŞAMALARI***

**İlk önce Ubuntu Mate işletim sistemi kurarak başladık.**



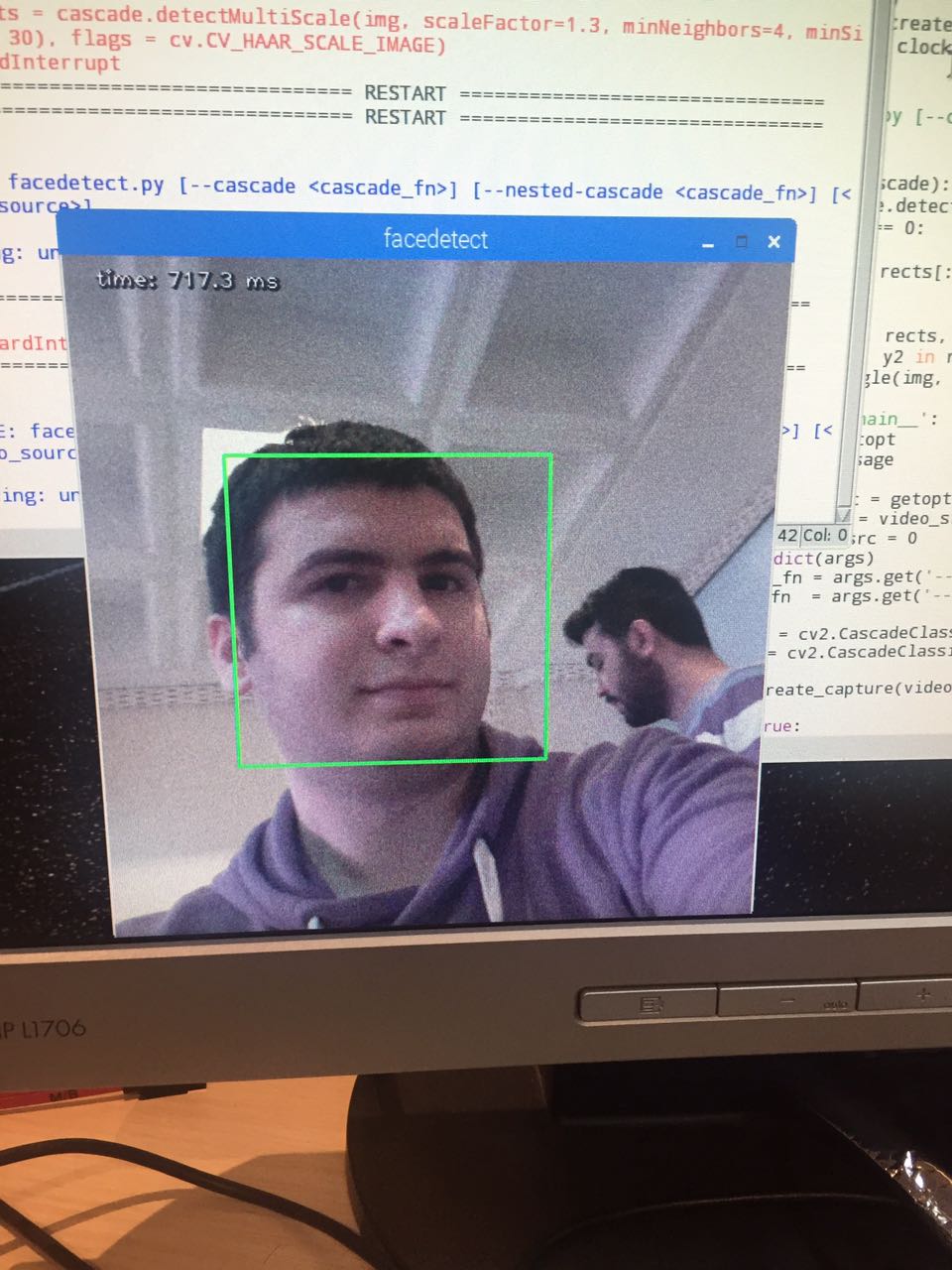
**Ubuntu Mate’den verim alamayınca farklı bir işletim sistemi olan Raspian Pixel ile yola devam etme kararı aldık.**



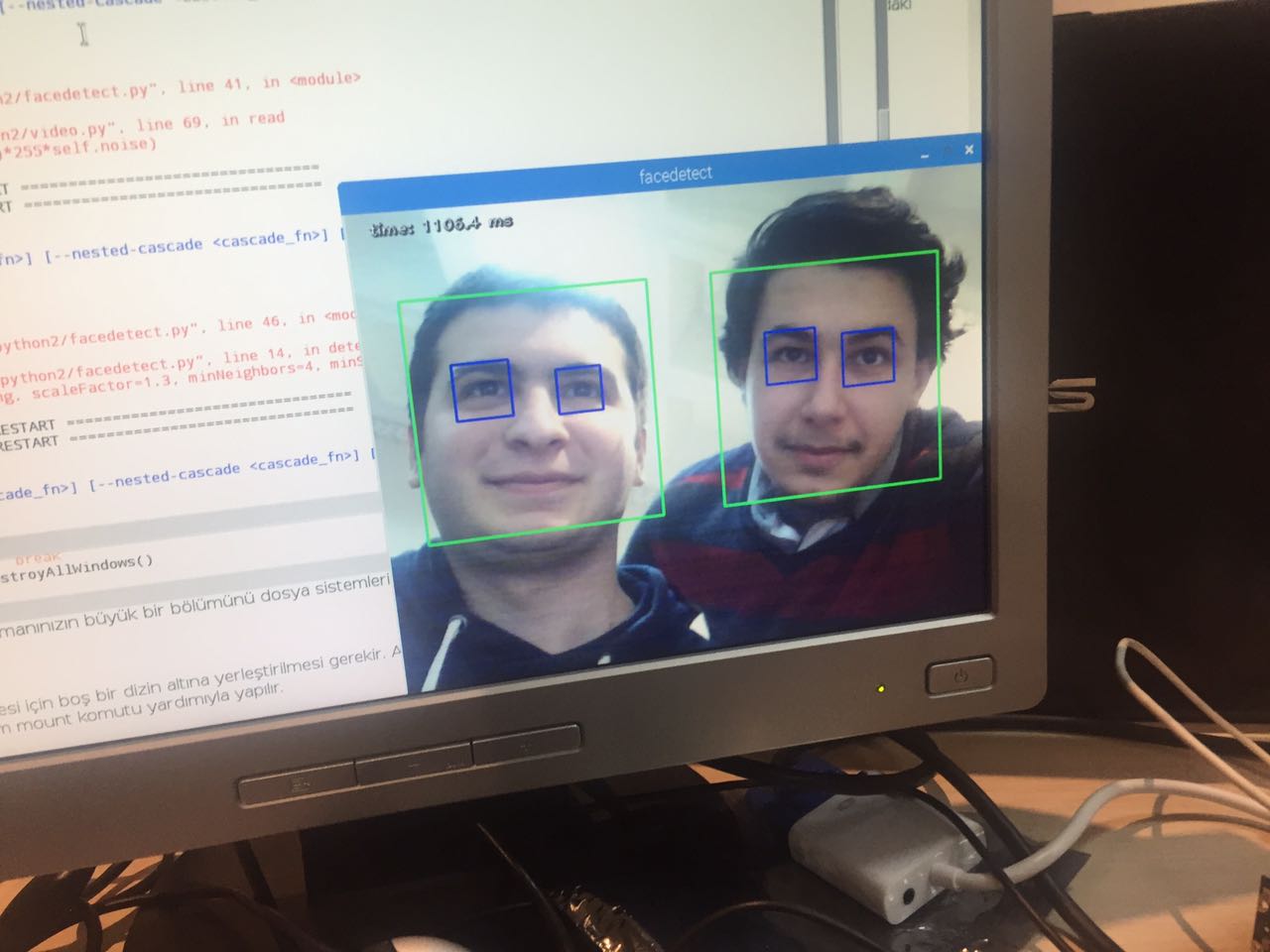
**Raspbian Pixel’i kurduktan sonra usb kameradan görüntü aldık.**



**Görüntü aldıktan sonrada resimden yüzü algılamayı gerçekleştirdik.**



**Daha sonrada uzun uğraşlar sonucunda kameradan aldığımız görüntüyle eş zamanlı olarak yüz tanıma işlemini gerçekleştirdik.**



***KAYNAK KODLARIMIZ***

**Exec.py**

#!/usr/bin/env python

# -\*- coding: utf-8 -\*-

import requests

import json

import sys

import os

import Ana\_Menu

from Ana\_Menu import secim

import Add\_Person\_Face

from Add\_Person\_Face import add\_person\_face

import Create\_Person

from Create\_Person import create\_person

import Create\_PersonGroup

from Create\_PersonGroup import create\_persongroup

import Delete\_Person

from Delete\_Person import delete\_person

import Delete\_PersonGroup

from Delete\_PersonGroup import delete\_persongroup

import Face\_Detect

from Face\_Detect import face\_detect

import Face\_Group\_Train

from Face\_Group\_Train import face\_grouptrain

import Face\_Identify

from Face\_Identify import face\_identify

import Delete\_PersonGroup

from Delete\_PersonGroup import delete\_persongroup

import CaptureImages

from CaptureImages import capture

import Keys

from Keys import keyz

import DetectImage

from DetectImage import detect

def exeter():

import Ana\_Menu

from Ana\_Menu import secim

import Keys

from Keys import keyz

secim()

secim = secim.secim

pin = ""

prompt = '> '

if secim == 1:

print "Grup Oluşturuluyor.."

create\_persongroup()

print "\n\nİşlem tamamlandı, anasayfa için \'A\', çıkış için \'Q\'"

pin = raw\_input(prompt)

if pin == "q" or pin == "Q":

return None

if pin == "a" or pin == "A":

exeter()

elif secim == 2:

print "Kişi Oluşturuluyor.."

create\_person()

print keyz.pID

print "\n\nİşlem tamamlandı, anasayfa için \'A\', çıkış için \'Q\'"

pin = raw\_input(prompt)

if pin == "q" or pin == "Q":

return None

if pin == "a" or pin == "A":

exeter()

elif secim == 3:

capture()

print "Kişi Yüzü Ekleniyor.."

add\_person\_face()

print "\n\nİşlem tamamlandı, anasayfa için \'A\', çıkış için \'Q\'"

pin = raw\_input(prompt)

if pin == "q" or pin == "Q":

return None

if pin == "a" or pin == "A":

exeter()

elif secim == 4:

os.chdir("./FaceImages/")

resimIsmi = ""

print "Resim adını giriniz: {}".format(resimIsmi)

resimIsmi = raw\_input(prompt)

keyz.captureName = "./FaceImages/{}.png".format(resimIsmi)

keyz.captureImgName = resimIsmi

add\_person\_face()

print "\n\nİşlem tamamlandı, anasayfa için \'A\', çıkış için \'Q\'"

pin = raw\_input(prompt)

if pin == "q" or pin == "Q":

return None

if pin == "a" or pin == "A":

exeter()

elif secim == 5:

detect()

print "Yüz Tespit Ediliyor.."

face\_detect()

face\_grouptrain()

keyz.faceID = face\_detect.fID

print keyz.faceID

print "\n"

face\_identify()

if keyz.confidence >= 0.5:

print "{} resmindeki kişi ile {} resmindeki kişi %{} oranla aynıdır.".format(keyz.captureImgName,keyz.detectImgName,keyz.confidence)

if keyz.confidence == 0:

print "{} resmindeki kişi ile {} resmindeki kişi, aynı kişi değildir.".format(keyz.captureImgName,keyz.detectImgName)

raw\_input(prompt)

print "\n\nİşlem tamamlandı, anasayfa için \'A\', çıkış için \'Q\'"

pin = raw\_input(prompt)

if pin == "q" or pin == "Q":

return None

if pin == "a" or pin == "A":

exeter()

elif secim == 6:

print "Grup Tanımlanıyor.."

face\_identify()

print "\n\nİşlem tamamlandı, anasayfa için \'A\', çıkış için \'Q\'"

pin = raw\_input(prompt)

if pin == "q" or pin == "Q":

return None

if pin == "a" or pin == "A":

exeter()

elif secim == 7:

delete\_persongroup()

print "\n\nİşlem tamamlandı, anasayfa için \'A\', çıkış için \'Q\'"

pin = raw\_input(prompt)

if pin == "q" or pin == "Q":

return None

if pin == "a" or pin == "A":

exeter()

exeter()

**Create\_Person.py**

# Install the Python Requests library:

# `pip install requests`

import requests

import json

def create\_person():

# Create Person

# POST https://api.projectoxford.ai/face/v1.0/persongroups/group1/persons

import Keys

from Keys import keyz

try:

response = requests.post(

url="https://api.projectoxford.ai/face/v1.0/persongroups/group1/persons",

headers={

"Ocp-Apim-Subscription-Key": "8007d3f4ca0041ae8d0c708fd5c1b7d5",

"Content-Type": "application/json",

},

data=json.dumps({

"name": "person01"

})

)

Q = '{content}'.format(content = response.content)

Parse\_pID = json.loads(Q)

create\_person.pID = Parse\_pID['personId']

keyz.pID = create\_person.pID

print('Response HTTP Status Code: {status\_code}'.format(

status\_code=response.status\_code))

print('Response HTTP Response Body: {content}'.format(

content=response.content))

except requests.exceptions.RequestException:

print('HTTP Request failed')

**Add\_Person\_Face.py**

#!/usr/bin/env python

# -\*- coding: utf-8 -\*-

# Install the Python Requests library:

# `pip install requests`

import requests

import json

import os

def add\_person\_face():

# Add Person Face

# POST https://api.projectoxford.ai/face/v1.0/persongroups/group1/persons/3fc6716d-8d6b-463b-99cb-c7381b68b888/persistedFaces

import Keys

from Keys import keyz

try:

os.chdir("../")

data = open(keyz.captureName, 'rb').read()

response = requests.post(

url="https://api.projectoxford.ai/face/v1.0/persongroups/group1/persons/{}/persistedFaces".format(keyz.pID),

data=data,

headers={

"Ocp-Apim-Subscription-Key": "8007d3f4ca0041ae8d0c708fd5c1b7d5",

"Content-Type": "application/octet-stream",

},

)

print keyz.captureName

print('Response HTTP Status Code: {status\_code}'.format(

status\_code=response.status\_code))

print('Response HTTP Response Body: {content}'.format(

content=response.content))

except requests.exceptions.RequestException:

print('HTTP Request failed')

***PROGRAMIN TAMAMININ OLDUĞU Face REC.zip DOSYASI CD NİN İÇİNDE MEVCUTTUR.***