



T.C.
SELÇUK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

RASPBERRY PI İLE GERÇEK ZAMANLI
YÜZ TANIMA VE KONTROL SİSTEMİ

Eevir RASHID

YÜKSEK LİSANS

Bilgisayar Mühendisliği Anabilim Dalı

MAYIS-2018
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

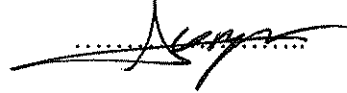
Eevir Abdullqader RASHID tarafından hazırlanan "Raspberry Pi İle Gerçek Zamanlı Yüz Tanıma ve Kontrol Sistemi" adlı tez çalışması 30/05/2018 tarihinde aşağıdaki jüri tarafından oy birliği ile Selçuk Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Danışman

Prof. Dr. Cemil SUNGUR



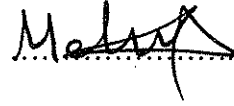
Başkan

Dr. Öğr. Üyesi Ömer Kaan BAYKAN



Üye

Dr. Öğr. Üyesi Mehmet HACİBEYOĞLU



Yukarıdaki sonucu onaylarım.

Prof. Dr. Mustafa YILMAZ
FBE Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.



Eevir RASHID

30 - 05 - 2018

ÖZET

YÜKSEK LİSANS

RASPBERRY PI İLE GERÇEK ZAMANLI YÜZ TANIMA VE KONTROL SİSTEMİ

Eevir RASHID

**Selçuk Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı**

Danışman: Prof. Dr. Cemil SUNGUR

2018, 74 Sayfa

Jüri

Prof.Dr. Cemil SUNGUR

Dr. Öğr. Üyesi Ömer Kaan BAYKAN

Dr. Öğr. Üyesi Mehmet HACİBEYOĞLU

Günümüzde artan terör, hırsızlık, gasp olayları gibi toplum dinamiklerini bozan etkenlerde kişisel kimliklerin hızlı ve otomatik bir şekilde belirlenmesi önemli hale gelmiştir. Kişisel özelliklerin tanımlanmasında yüz, parmak izi, damar tanıma sistemleri gibi kişiye özel özelliklerin belirlenebildiği biyometrik tanıma sistemleri ön plana çıkmaktadır. Bu sistemler sayesinde gümrük kapıları, bankalar, resmi daireler gibi yoğun kalabalığın olduğu ve güvenliğin önplanda tutulduğu kurumlarda şüpheli kişilere hızlı müdahaleler yapılabilmektedir. Bu tez çalışmasında Raspberry Pi tabanlı bir kamera sistemi kullanılarak, yüz tanımanın yapıldığı kişisel bir güvenlik sistemi tasarlanmıştır. Raspberry Pi kamera, hafıza kartı, monitör, klavye ve benzeri çevre birimlerinin bağlanabildiği mikroişlemci tabanlı bir mini bilgisayardır. Bu çevre birimlerinin pratik bir şekilde kullanımına imkân verdiği için Raspberry Pi platformu kullanılmıştır. Platform üzerinde veritabanı olarak kullanılan Microsoft Azure sistemine birçok yüz verisi kaydedilerek, Raspberry Pi kamerası ile alınan yüzlerin karşılaştırılması yapılmıştır. Eşleşen yüzlerin doğruluğu bazı istatistiksel metotlar kullanılarak, kişinin beyaz listede olup olmadığı belirlenmiştir. İstatistiksel karşılaştırma sonucuna göre eşleşen özellikler %50 ve üzerinde ise kapının açılması diğer durumlarda kapının kapalı kaldığı bir elektromekanik sistem tasarlanmıştır. Tasarlanan bu sistemde, belirlenen alana giriş yapmak isteyen kişilerin kimliğini belirleyip bu sonuca göre kapı kilidini açıp-kapanan bir mekanizmanın yazılım ve donanımı gerçekleştirilmiştir. Geliştirilen bu sistemde yapılan deneysel çalışmalarda büyük oranda başarı sağlanmıştır.

Anahtar Kelimeler: Raspberry Pi 3, Yüz Tanıma, Erişim kontrolü, Veri tabanı, Microsoft Azure,

ABSTRACT

MS THESIS

REAL TIME FACE RECOGNITION AND CONTROL SYSTEM WITH RASPBERRY PI

Eevir RASHID

**The Graduate School of Natural and Applied Science of Selçuk University
The Degree of Doctor of Philosophy in Computer Engineering**

Advisor: Prof. Dr. Cemil SUNGUR

Year, 74 Pages

Jury

Prof. Dr. Cemil SUNGUR

Asst. Prof. Dr. Ömer Kaan BAYKAN

Asst. Prof. Dr. Mehmet HACIBEYOĞLU

Nowadays, it has become important to determine the personal identities quickly and automatically in the factors that disrupt the social dynamics such as increasing terror, theft and extortion. Biometric identification systems, such as face, fingerprint, vein recognition systems, etc., can be used to identify personal characteristics. Thanks to these systems, it is possible to quickly intervene in suspicious persons where there is a dense crowd, such as customs gates, banks, government offices, and institutions where security is held in the foreground. In this thesis study, a security system was designed using a Raspberry Pi based camera system to enable personal face recognition. Raspberry Pi, is a mini computer microprocessor-based to which a camera, memory card, monitor, keyboard and similar peripherals can be connected. The Raspberry Pi platform was used because this environment allows the units to be used in a practical way. On the platform, hundreds of data were saved to the Microsoft Azure system used as the database, and the Raspberry Pi compares the faces were taken by the camera with the data that saved earlier in the database. The accuracy of matching faces was determined using some statistical methods to determine whether the person was in the white list or not. According to the statistical comparison result, an electromechanical system is designed which can be turned on and off according to the achievement of 50% and above. This system, which is designed to detect unknown persons and to use a mechanism that can be turned on and off according to this result, offers a solution with a high performance with face recognition using a mini computer. This system, which is designed to detect unknown persons and to use a mechanism that can be turned on and off the doors' lock according to the obtained result and offers a solution with a high performance using face recognition and a mini computer.

Keywords: Raspberry Pi, Face Recognition, Access Control, Database, Microsoft Azure

ÖNSÖZ

Tez çalışmamı yapabilmem için her zaman bilgi birikimlerini benimle paylaşarak bana yol gösterici olan saygı değer hocam Prof. Dr. Cemil Sungur' a teşekkür ederim. Tüm bu çalışmalar süresince bana yardımcı olan, varlıkları ile destek ve moral veren aileme ve özellikle canım babama ve ağabeyim Abdullah RASHED' e, beni her zaman ve her konuda destekledikleri için teşekkürlerimi sunuyorum.

Eevir RASHID
KONYA-2018

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR	ix
ŞEKİL VE ÇİZELGELER	xi
1. GİRİŞ	1
2. KAYNAK ARAŞTIRMASI	3
3. MATERYAL VE YÖNTEM.....	8
3.1 Raspberry Pi 3 Platformu	8
3.1.1. SD Kart.....	10
3.1.2. Yazılım Desteği.....	10
3.1.3. Windows 10 Core IoT	12
3.2. Microsoft Azure nedir?	12
3.3. Microsoft Life Cam HD-3000.....	13
3.4. Elektrik Kapı Kilidi.....	14
3.5. 5V 2-Kanallı Röle Modeli.....	15
3.6. Logitech C270 Kamera	16
3.7. Sistem Mimarisi	17
3.8. Genel Kontrol Sistemi ve Algoritmasının Akış Şeması.....	19
3.9. Yüz görüntülerinin alınma yöntemleri	21
3.10. Sistem Tanımı.....	21
3.10.1 Kullanılan Yazılım	21
3.10.2 İnsanlarda Yüz Tanıma	22
3.10.3 Yüz Tanıma Sistemleri.....	22
3.11. Yüz Tanıma İçin Farklı Teknikleri.....	23
3.11.1 Farklı Yaklaşım	23
3.11.2 Bütünsel Yaklaşım	23
3.11.3 Özellik Temelli Yaklaşım	24
3.11.4 Hibrid Yaklaşım	24
3.11.5 Özyüzler Yöntemi	24
3.11.6 Temel Bileşen Analizi (PCA)	25
3.11.7 Bağımsız Bileşen Analizi (ICA)	26
3.11.8 Doğrusal Ayırıcı Analizi (LDA)	26
3.11.9 Evrimsel Takip (EP).....	27
3.11.10 Kernel Yöntemleri.....	27
3.11.11 İz Dönüşümü	27
3.11.12 Destek Vektör Makinesi (SVM)	27
3.11.13 Elastik Demet Grafik Eşleştirme (EBGM)	28

3.11.14	Aktif Görünüm Modeli (AAM)	29
3.11.15	3B Dönüştürülebilir Model	30
3.11.16	3B Yüz Tanıma	30
3.11.17	Bayes Çerçeve	31
3.11.18	Gizli Markov Modelleri (HMM).....	31
3.11.19	Yükseltme ve Topluluk Çözümler	32
3.11.20	En Yakın Özellik Hattı Yöntemi	32
3.12	Tasarım ve Uygulama	33
3.13	Sistemin Süreçleri:.....	34
3.14	Metot.....	35
3.14.1	Donanım	35
3.14.2	Yazılım	36
3.14.3	Yüz Tanıma Uygulamalarının Çalışması	37
3.14.5	Microsoft Bilişsel Hizmetler	39
3.14.6	Kapı Kilidinin Açılması:	41
3.15	Karşılama.....	41
3.16	Sistemin Değiştirilmesi	41
4.	SONUÇLAR VE TARTIŞMALAR	47
4.1	Donanım	47
4.2	Yazılım	47
4.2.1	Tanıma Modu	47
4.3	Deneyler ve Sonuç	49
5.	SONUÇ VE ÖNERİLER.....	55
5.1	Sonuç.....	55
5.2	Gelecek İş ve Tavsiye	56
KAYNAKLAR	57	
EKLER	63	
ÖZGEÇMİŞ	74	

SİMGELER VE KISALTMALAR

Kısaltmalar

2D	:Two Dimensional	(İki Boyutlu)
3D	:Three Dimensional	(Üç Boyutlu)
AC	:Alternating Current	(Alternatif Akım)
AMM	:Active Appearance Model	(Aktif Görünüm Modeli)
API	:Application Program Interface	(Uygulama Programlama Arayüzü)
ATM	:Automated Teller Machine	(Otomatik Vezne)
AVR	:Automatic Voltage Regulating Relays	(Otomatik Voltaj Regülasyon Rölesi)
CAGR	:Compound annual growth rate	(Yıllık Bileşik Büyüme Oranı)
CNN	:Convolutional Neural Network	(Konvülsiyon Sinir Ağları)
CPU	:Central Processor Unit	(Merkezi İşlemci Birimi)
CSI	:Camera Serial Interface	(Kamera Seri Arabirimi)
CSS	:Cascading Style Sheets	(Basamaklı Stil Şablonu)
DC	:Direct Current	(Doğru akım)
DCT	:Differential Current Transformer	(Diferansiyel Akım Trafosu)
DPM	:Data Protection Manager	(Veri Koruma Yöneticisi)
DSI	:Display Serial Interface	(Seri Arabirimi Göster)
EBGM	:Elastic Bunch Graphic Mapping	(Elastik Bunch Grafik Haritalama)
FES	:Fixed Earth Station	(Sabit Yer İstasyonu)
FL	:Feature Line	(Özellik Hattı)
GB	:Giga Byte	(Gigabayt)
GPIO	:General Purpose Input Output	(Genel Amaçlı Giriş Çıkışı)
GPRS	:Global Position Radio System	(Küresel Konum Radyo Sistemi)
GPRS	:General Packet Radio Services	(Genel Paket Radyo Hizmetleri)
GPU	:Graphics Processing Unit	(Grafik İşleme Ünitesi)
GSM	:Global System for Mobile Communications	(Mobil iletişim için küresel sistem)
GUI	:Graphical User Interface	(Grafiksel kullanıcı arayüzü)
HD	:High Definition	(Yüksek çözünürlük)
HDMI	:High-Definition Multimedia Interface	(Yüksek Çözünürlüklü Multimedya Arayüzü)
HMM	:Hidden Markov Models	(Gizli Markov Modelleri)
HTML	:HyperText Markup Language	(HyperText İşaretleme Dili)
IBM	:International Business Machines	(Uluslararası İş Makinaları)
ICA	:Independent Component Analysis	(Bağımsız Bileşen Analizi)
ID	:Identification	(Tanımlama)
IEEE	:Institute Of Electrical And Electronics Engineers	(Elektrik-Elektronik Mühendisleri Enstitüsü)
IoT	:Internet Of Thing	(İnternetin Şeyleri)
IP	:Internet Protocol	(İnternet protokolü)
ISA	:Industry Standard Architecture	(Endüstri Standart Mimarisi)
LDA	:Linear Discriminant Analysis	(Doğrusal Ayırıcı Analizi)
LED	:Light Emitting Diode	(Işık Yayan Diyot)
Mb	:Mega Byte	(Mega Byte)
MI-MO	:Memory Input-Memory Output	(Bellek Girişi-Bellek Çıkışı)
MODEM	:Modulator-Demodulator,	(Modülatör-Demodülatör)
NFL	:Nearest Feature Line	(En Yakın Özellik Hattı)
NOOBS	:New Out Of the Box Software	(Kutusundan Yeni Çıkmış Yazılım)
OEM	:Original Equipment Manufacturer's	(Orijinal ekipman üreticileri)

OpenCV	:Open Source Computer Vision	(Açık Kaynak Bilgisayar Görmesi)
PC	:Personal Computer	(Kişisel bilgisayar)
PCA	:Principal Component Analysis	(Temel bileşenler Analizi)
PCB	:Printed Circuit Board	(Baskılı devre kartı)
PDF	:Probability Density Function	(Olasılık Yoğunluğu Fonksiyonu)
PIC	:Peripheral Interface Controller	(Çevresel Arabirim Kontrol Cihazı)
PIR	:Passive Infrared Sensor	(Pasif Kızılötesi Sensör)
RAM	:Random Access Memory	(Rastgele Erişilir Bellek)
RDP	:Remote Desktop Protocol	(Uzak Masaüstü Protokolü)
S_b	:Between-class scatter matrix	(Sınıf arasında dağılım matrisi)
SD	:Storage Device	(Depolama aygıtı)
SDK	:Software Development Kit	(Yazılım Geliştirme Kiti)
S_w	: Within-class scatter matrix	(Sınıf içi dağılım matrisi)
SQL	:Structured Query Language	(Yapılandırılmış Sorgu Dili)
TPFMs	:Topology Protection Feature Maps	(Topoloji Koruma Özelliği Haritaları)
UI	:User Interface	(Kullanıcı arayüzü)
USB	:Universal Data Bus	(Evrensel Veri Yolu)
VB	:Visual Basic	(Visual Basic)
WAP	:Wireless Protected Access	(Kablosuz Korunmalı Erişim)
WEP	:Wireless Encryption Protocol	(Kablosuz Şifreleme Protokolü)
WIFI	:Wireless Fidelity	(Kablosuz Bağlılık)
WPS	:Workplace Safety	(İşyeri güvenliği)
XAML	:Stands For Extensible Application Markup Language	(Genişletilebilir Uygulama İşaretleme Dili İçin Standlar)
XSLT	:Extensible Stylesheet Language Transformation	(Genişletilebilir Stil Sayfası Dili Dönüşümü)

ŞEKİL VE ÇİZELGELER

Şekil 3.1	Raspberry Pi 3 model B (Diener, 2016)	9
Şekil 3.2	Raspberry Pi 3 özellik dağılımlarının kart üzerinde gösterilmesi	10
Şekil 3.3	Raspberry Pi içinde kullanabilen programlar	11
Şekil 3.4	Raspberry Pi Win 10 IoT.....	12
Şekil 3.5	Microsoft Life CAM HD-3000 (Microsoft, 2016)	13
Şekil 3.6	Elektrik kapı kilidi.....	14
Şekil 3.7	5V 2-Kanallı röle modülü (Essabre-90xx-rpi, 2018)	15
Şekil 3.8	Rölenin pin tanımı (Essabre-90xx-rpi,2018).....	16
Şekil 3.9	Logitech C270 kamera (Biosis, 2018).....	17
Şekil 3.10	Sistemin genel çalışma şekli.....	18
Şekil 3.11	Raspberry Pi 3 ve çevre donanım birimleri ile tasarlanan sistem mimarisi...18	
Şekil 3.12	Sistem açıklaması	19
Şekil 3.13	Sistemin akış şeması.....	20
Şekil 3.14	Bazı yüz tanıma yöntemleri (Mehrdad, 2013).....	24
Şekil 3.15	İki boyutlu noktaların temel bileşenleri (PC). Birinci ana bileşen, ortalama kare hatası anlamında 2B'den 1B'ye kadar optimal doğrusal boyut küçültme sağlar (Ferdous ve Ahmed, 2008)	26
Şekil 3.16	Konvolusyon sonucu (Ferdous ve Ahmed, 2008)	28
Şekil 3.17	Etiketli grafik (Ferdous ve Ahmed, 2008).....	29
Şekil 3.18	Eğitim görüntüsünün şekil ve normalleştirilmiş şekilli dokuya ayrılması (Ferdous ve Ahmed, 2008)	30
Şekil 3.19	Yüz tanıma görüntüsü için HMM (Ferdous ve Ahmed, 2008)	31
Şekil 0.20	Blok çıkarma formu (Ferdous ve Ahmed, 2008).....	32
Şekil 3.21	x1 ve x2 prototip özellik noktalarının genellenmesi (Turk ve Pentland, 1991)	33
Şekil 3.22	Sistemin kullanıcı arayüzü	34
Şekil 3.23	Donanım bağlantıları	36
Şekil 3.24	Belirli noktalar (Microsoft.Azure, 2018)	38
Şekil 3.25	Gönderilen resimden json olarak geri dönen sonuç	40
Şekil 3.26	MBM yapılandırması	42
Şekil 3.27	Uzaktan bağlantı için cihazın ayarları	42
Şekil 3.28	Programın ana ekranı ve kişi eklemek için misafir ekle butonunun belirtilmesi	43
Şekil 3.29	Fotoğraf çekme ekranı	44
Şekil 0.30	Fotoğraf çekme sonunda gelen ekran.....	42
Şekil 3.31	Yeni kişi eklendikten sonra ana sayfanın görünüşü	45
Şekil 3.32	Veritabanındaki kişilerin görüntüsü	45
Şekil 3.33	Zil butonu	46
Şekil 4.1	Bilinen ziyaretçiler için basılan zilin çıkışını göstermektedir.	48
Şekil 4.2	Bilinmeyen ziyaretçiler için basılan zil çıkışını göstermektedir	48
Şekil 4.3	Kullanıcıya giden uyarı mesajı.....	48
Şekil 4.4	Logitech kamera ile maksimum mesafe 3m.....	49
Şekil 4.5	8 metreye kadar mesafe Microsoft LifeCam HD-3000.....	50
Şekil 4.6	Yüz 90° ve mesafe 0,5 m ile yüz tanıma	51
Şekil 4.7	90° açısı ve 1 m mesafe ile yüz tanıma	51
Şekil 4.8	90° açısı ve 1,5 m mesafesi ile yüz tanıma.....	52
Şekil 4.9	90° açı ve mesafe 2 m ile yüz tanıma	52

Çizelge 3.1 Microsoft Life CAM HD-3000 Özellikleri	14
Çizelge 3.2 Elektrik kapı kilidi özellikleri.....	14
Çizelge 4.1 Deney 1.Yüz tanıma doğruluğu.....	49
Çizelge 4.2 Deney 2. Yüz tanıma doğruluğu.....	52
Çizelge 4.3 Deney 3.Yüz tanıma doğruluğu.....	53
Çizelge 4.4 Deney 4.Yüz tanıma doğruluğu.....	53
Çizelge 4.5 Deney 5.Yüz tanıma doğruluğu.....	53
Çizelge 4.6 Deney 6.Yüz tanıma doğruluğu.....	53
Çizelge 4.7 Deney 7.Yüz tanıma doğruluğu.....	53
Çizelge 4.8 Deney 8.Yüz tanıma doğruluğu.....	53
Çizelge 4.9 Deney 9.Yüz tanıma doğruluğu.....	54
Çizelge 4.10 Deney 10.Yüz tanıma doğruluğu.....	54
Çizelge 4.11 Deney 11.Yüz tanıma doğruluğu.....	54

1. GİRİŞ

Günümüzde suçların çeşitliliği ve özellikleri çok farklı hale gelmiştir. Modern dünyada teknolojinin gelişmesi ile beraber suçların niteliği de artmıştır. Hergün çok sayıda hırsızlık, gasp, terör olayları, kaçırma gibi birçok olay meydana gelmektedir. Dolayısıyla insanların ve kurumların gündelik hayatta güvenliklerini sağlaması önemli bir hale gelmiştir. Bu amaçla kişisel veya kurumsal güvenliğin sağlanması için birçok teknoloji geliştirilmiştir ve bu sahadaki çalışmalar halen devam etmektedir. Güvenlik sistemleri içerisinde kişinin ayırt edilebilmesi için biyometrik tanıma sistemleri öne çıkmaktadır. Bu sistemler yüz, parmak izi, avuç içi, avuç içi damar, retina vb. kişiye has özelliklerin tanınabilmesi için geliştirilmiş algoritmaya dayalı yazılımlardır. Bu özelliklerin tanınabilmesi için görüntü işleme, video işleme, özellik çıkarma, özellik azaltma ve makine öğrenmesi teknikleri beraber kullanılmaktadır. Kişiye has özellikler ihtiyaç duyulan algoritma ile çıkarıldıktan sonra bir yapay zekâ yöntemi ile eğitim işlemine tabi tutulur. Eğitilen sistem aynı özelliğin tekrar sisteme sorulması ile veritabanında eşleşen özellik kişinin kim olduğuna dair gerekli verileri geri döndürür. Böylece sistem ya alarm verir ya da istenen hedefe göre bir çıktı verir. Alarm sistemleri ile şüpheli kişiler yakalanabilir veya soygun gibi suçlar önceden tahmin edilebilir. Diğer kullanım alanları ise kişinin kimliği tanımlandıktan sonra kasanın açılması, kapının açılması, para çekme, izleme vb. hedefe özel çıktılar elektronik sistemler ile gerçekleştirilebilir. Bu alanlarda kullanılması ile yazılım ve elektronik tabanlı biyometrik tanıma sistemleri günlük hayatımızın bir parçası haline gelmiştir.

Literatürde güvenlik sistemleri ile ilgili birçok çalışma yapılmıştır. GSM/GPRS tabanlı kablosuz ev güvenlik sistemi (Lee ve ark., 2007); (Zhao ve Ye, 2008), mikrodenetleyici kullanarak otomatik anahtarlama manyetik kapı kilit sistemine dayanan yüz tanıma (Hassan ve ark., 2012); (Assaf ve ark., 2012), sensör tabanlı ev otomasyonu ve güvenlik sistemi (Ibrahim ve Zin, 2011) gibi elektronik ve yazılımın bir arada olduğu güvenlik amaçlı gömülü sistem araştırmaları sıklıkla yapılmaktadır. Bu tez çalışmasında Raspberry Pi sistemi ve kamerası ile bir yüz tanıma sistemi tasarlanarak elektronik bir kapı sisteminin kontrolü yapılmıştır. Yüz tanıma sisteminin veritabanı belli mesafe ve açılarda alınan görüntülerden oluşmaktadır. Bu veritabanı tez kapsamında oluşturulmuştur. Microsoft Azure bulut platformuna kaydedilen yüz resimleri Raspberry Pi kamerası ile alınan görüntüler ile karşılaştırılmıştır. Görüntüler veritabanı ile karşılaştırıldıktan sonra %50 ve üzeri başarı gösteren eşleştirmeler ile

elektro-mekanik kapı açılmaktadır. Eğer sistem %50 altı eşleştirme gösteriyorsa şüpheli olarak tespit edilen görüntü Gmail üzerinden sistemi kontrol eden kişiye gönderilmektedir ve elektro-mekanik kapı sistemi açılmamaktadır. Aynı zamanda %50 ve üzeri eşleştirme veren görüntülerde kişiye ismiyle hitap edilen hoş geldin sesli mesajı verilmektedir.

Bu tez çalışmasında geliştirilen sistem ile kişi tanımlamanın gerekli olduğu güvenlik sistemleri bulut üzerinde kontrol edilebilir. Eşleşen görüntüler ile sadece elektro-mekanik kapı sistemleri değil diğer birçok elektro-mekanik sistemlerde ihtiyaç duyulan alanlarda kontrol edilebilir. Geliştirilen sistemler ev, üniversite, hastane, banka gibi yerlere şüpheli veya istenmeyen kişilerin girmesinin engellenmesi gerektiği durumlarda otomatik tanımlama ile pratik bir kullanım ortamı sağlayabilir. Bu nedenle bu tez çalışması yüz tanıma, bulut, Raspberry Pi, Microsoft Azure tabanlı bir otomatik biyometrik tanıma çerçevesi sunduğu için önem arz etmektedir. Bu sistemin diğer sistemlerden farkı daha hızlı, daha hassas çalışmasıdır.

2. KAYNAK ARAŞTIRMASI

Yüz tanıma teknikleri, ilk olarak Francis Galton tarafından 1888'de önerilmiştir. Galton önemli yüz özelliklerini ya da kilit- noktaları saptamaya odaklanan çok sayıda çalışma yapmıştır. Tipik kilit noktalar arasında göz köşeleri, ağız köşeleri, burun ucu ve çene kenarı bulunmuştur. Bu yüz kilit-noktaları arasındaki göreceli mesafeler ölçülerek, her bir yüzü tanımlamak için bir özellik vektörü oluşturulabilmiştir. Bilinmeyen bir yüzün özellik vektörünü bilinen yüzlerin veri tabanındaki özellik vektörleri ile karşılaştırarak en yakın eşleşme belirlenir. Bu teknik yüz yönü ya da açısındaki değişikliklere son derece duyarlıdır ve kontrollü deneylerde bile yüksek başarı gösterememektedir (Kosugi, 1995).

Yüz tanıma için sinir ağlarının kullanılması, literatürde yer alan diğer bir popüler yaklaşımdır. (Arun ve ark., 2014) tarafından yapay sinir ağları tabanlı yüz tanıma sınıfına öncülük eden özdenetimli haritalara (SOM) ve konvolüsyonel sinir ağlarına (CNN) dayanan bir yüz tanıma sistemi geliştirilmiştir.

Yüz tanımda kullanılan SOM'un çıkış bilgileri, Evrimsel sinir ağını (CNN) beslemiştir. CNN (Arun ve ark., 2014), bir dereceye kadar translasyon ve rotasyon bağımsızlık sergilerler ve kullanımları yine biyolojik gözlemler tarafından motive edilir. Bir CNN, seri bağlı üç boyutlu sinir ağlarından oluşmuştur. Her bir ağ çeşitli nöron düzlemlerden oluşur ve her nöron sadece 6 komşusuyla bağlantılıdır. Lawrence ve arkadaşlarının sistemi, üç katmanlı bir CNN kullanmıştır. Birinci ve ikinci katmanlar 20 düzlemsel üç boyutlu sinir ağı iken, üçüncü tabaka birbirine bağlı 40 birim sinir ağıdır. Her tabakada, her bir nöron, önceki tabakadaki 3x3 pencereye karşılık gelmiştir. Buna ek olarak, her bir nöral ağ arasında bir alt örnekleme süreci meydana gelir; burada uyarım haritası, hem yatay hem de dikey yönlerde iki faktörden oluşan bir alt örnekleme ile alt-örneklenmiştir. Bu sistem 400 görüntüden oluşan bir veritabanında% 94'lük bir tanıma oranına ulaşan bir performans sergilemiştir. Nispeten küçük standartlaştırılmamış bir veritabanının kullanılması, bu sistemin performansının diğerleri ile karşılaştırılmasını zorlaştırır.

Yapay sinir ağlarının karşılaştığı en büyük problem, problemin yüksek boyutu ile başa çıkamamasının bir sonucu olmuştur. Örneğin, 128×128 piksel büyüklüğündeki küçük bir görüntü bile, işlem için 16,384 girişli nöron içeren bir sinir ağı gerektirmiştir. Yüzün bu temsilinde çok miktarda fazlalık vardır ve bu da sinir ağının eğitiminde çok yavaş bir yakınsamaya neden olmuştur. Ayrıca, böyle bir ağı eğitmek ve sağlam

performans sağlamak için son derece büyük bir eğitim seti gerekir (>> 16,384). Bu, bireyin sadece bir veya iki imgesinin mevcut olabileceği gerçek dünya uygulamalarında genellikle mümkün değildir (McKenna ve Gong, 1997).

Cottrell ve Fleming (McKenna ve Gong, 1997) sınıflandırma ağı olarak adlandırılan ikinci bir geri yayılım ağını sisteme ekleyerek bu problemin üstesinden gelmeye çalışmaktadır. Programlarında, orijinal otomatik ilişkilendirme ağı yalnızca sınıflandırma ağını eğitmek için kullanılmıştır. Sınıflandırma ağı eğitildikten sonra karşılaştırmanın belirlenmesinden öncelikle sorumludur. Cottrell ve Fleming, en iyi koşullarda bu tasarımın bir sinir ağının bir özyüz özellik ağından daha iyi olmadığı sonucuna varmışlardır (McKenna ve Gong, 1997).

Yüz tanıma uygulamalarındaki sinir ağları ile ilgili temel problemler, yüzün yapı özelliklerinin açık bir şekilde kullanılmadığı için ortaya çıkmıştır. Ağın eğitiminde karşılaşılan zorluklar da eklendiğinde, yüz tanıma problemine uygun olmayan sinir ağlarının doğrudan uygulanması sağlanmıştır.

Görüntü işlemede, korelasyon şablon eşleştirmesinde çok yaygın kullanılan bir tekniktir. Bu, yüz özelliklerini tespit etmek ve yüz tanıma için Kosugi (Moghaddam ve Pentland, 1995) tarafından uygulanmıştır. Çalışmada ilk olarak, bilinmeyen yüzün gri tonlamalı görüntüsünü katmanlar halinde sıralayarak bir görüntü vektörü oluşturulmuştur. Bu görüntü vektörü ile veri tabanında kayıtlı her bir bilinen yüz vektörü arasındaki korelasyonun hesaplanması için görüntü vektörü ile veritabanındaki her bir yüz karşılaştırılmıştır. En büyük korelasyonu sağlayan veri tabanının verisi en yakın eşleşme olarak kabul edilmiştir. Korelasyon, ölçeklemeye ve gürültüye karşı yüksek duyarlılığa sahiptir ve kötü aydınlatılmış koşullarda iyi performans gösterememektedir. (Moghaddam ve Pentland, 1995)

Otomatik yüz tanıma problemine uygulanan tekniklerin bir kısmında tanıma için hangi yüz özelliklerinin önemli olduğu konusunda rastgele kararlar verilmiştir. Örneğin, korelasyona dayalı teknikler, bir yüz görüntüsünün tüm piksellerinin eşit derecede önemli olduğunu varsaymıştır. Diğer teknikler bazı yüz özelliklerini diğerlerine göre önemli varsayarlar. Önerilen yöntem yöntem, yüz kilit noktaları arasındaki mesafe ile her bir yüzün benzersiz bir şekilde tanımlanabileceği varsayımı üzerine geliştirilmiştir. Bunun doğruluğu tartışmalıdır. (Moghaddam ve Pentland, 1995)

Yüz tanımda yüz özelliklerinin önemine göre ölçeklendirilmesi veya normale indirgenmesi, özyüz (eigenfaces) tekniğinin temel dayanağıdır. Bu yöntemde, öz görüntü (eigen image) olarak adlandırılan vektörlerin ortogonal temel kümesindeki yüz

görüntüleri arasındaki değişimi yakalamaya çalışılmıştır. Özyüzler bu nedenle, yüzler arasındaki en önemli varyasyonları gösteren görüntü vektörleridir.

Yüzlerin, görüntü uzayda basit bir bağlantılı bölge oluşturduğu varsayımı altında (tüm olası görüntüleri, yüz ve yüz harici içeren alan), herhangi bir yüzü özyüzlerin doğrusal bir kombinasyonu olarak temsil edebilmektedir. Her bir yüz bu nedenle, bu yüzü oluşturmak için gerekli olan her bir özyüzün oranlarını içeren bir ağırlık vektörü ile temsil edilebilmiştir. Bilinmeyen bir yüzün ağırlık vektörü, bilinen yüzlerin bir veritabanındaki ağırlık vektörleri ile karşılaştırılarak bir eşleşme belirlenebilir.

Korelasyona dayalı tekniğin aksine özyüz kullanılan teknikte, gürültüden, zayıf aydınlatma koşullarından ve kısmi bozukluklardan etkilenmez. Özyüzler, ölçek, rotasyon ve ifadedeki küçük değişikliklere nispeten duyarsızdır. Ayrıca, çoklu çözünürlüklü piramitlerin ve çoklu öz boşlukların kullanımıyla, özyüzler ölçek, düzlem içi ve düzlem dışı rotasyondaki büyük değişikliklerle başa çıkmak için uyarlanabilir (Moghaddam ve Pentland, 1995).

Özyüz kullanılan tekniklerde, değişen aydınlatma koşullarında %96 doğru sınıflandırma üretmiştir. Özyüzler de büyük ölçekli (10000 kişinin üzerinde) veritabanları ile doğruluğunu koruduğun gösterilmiştir. Özyüzlerin geliştirilmesinden bu yana birçok ileri teknik önerilmiştir. Bunlar, artmış yoğunluk modelleme Öz-özellikler, Modüler öz-uzaylar ve Bayes benzerlik ölçüsü içerir. Tüm bu teknikler, özyüz tekniği ile ortaya konan temel çalışmalara dayanarak özyüzlere uzantılar olarak düşünülebilir (Yang ve Huang, 1994).

Öz nitelikler, ifade, gizleme veya bozukma gibi değişiklikler sırasında daha fazla doğruluk sağlamıştır. Modüler öz uzay (modular eigen space), düzlem dışı rotasyondaki büyük değişikliklere karşı doğruluk sağlamıştır.

Geliştirilmiş yoğunluk modellemesi ve Bayes benzerlik ölçüsü daha yüksek hesaplama karmaşıklığı daha gelişmiş tanıma doğruluğu sunmaktadır. Örneğin geliştirilmiş yoğunluk modelleme ile tanıma doğruluğunu %8'e kadar artırabilir, ancak bu, hesaplama yükünün iki katına çıkar. Benzer şekilde, Bayes benzerlik ölçüsü tanıma doğruluğunu %7 oranında artırabilir. (Yang ve Huang, 1994). Gerçek zamanlı sistem ve zaman kısıtlamalarında yüksek hesaplama verimliliğinin gerekliliği nedeniyle, yoğunluk modellemesi ve Bayes benzerlik ölçüsünün gerçek zamanlı uygulamalarda kullanılması uygun olmamaktadır.

Hassan ve arkadaşları (2012), tarafından, kapı kilit güvenlik sistemi için manyetik kilidi açmak / kapatmak amacıyla giriş / çıkış kontrolörü olarak yüz tanıma sistemi ve çevresel Arabirim Kontrol Cihazı (PIC) temelli Grafik Kullanıcı Arayüzü (GUI) geliştirilmiştir (Hassan ve ark., 2012). Geliştirme, MATLAB R2009a' da yerleşik GUI arayüzü ile mikro denetleyici ile yapılmıştır. Güvenlik sistemi için kapı kilidi otomatik anahtarlamalı manyetik kilitle gerçekleştirilmiştir. Ayrıca, giriş alanına insan varlığında otomatik olarak bir sensörle görüntü yakalama özelliği ekleyerek tam otomatik yüz tanıma sistemi haline getirilebilmesi için sistem geliştirileceği konusunda öneri getirilmiştir.

Padmapriya ve arkadaşları (2012) tarafından cilt renginin bilgisine dayanan gelişmiş yüz algılama ve tanıma yöntemi geliştirilmiştir. Yüzü algılamak için AdaBoost algoritması kullanılmıştır. Temel bileşen analizi (PCA) algoritması kullanılarak, mevcut yüzün ana bileşenlerini, önceden oluşturulmuş bir yüz veritabanındaki verilerle karşılaştırarak belirli bir yüz eşleştirilmesi yapılması önerilmiştir (Padmapriya ve KalaJames, 2012).

Prasad ve arkadaşları (2014) mobil cihazlar için Raspberry Pi ve PIR sensörünü kullanan akıllı gözetim izleme sistemi tasarlanmış ve uygulanmıştır. Raspberry Pi uzaktan algılama ve izleme için hareket dedektörleri ve anlık video akışı ve kayıt için video kamera kullanılmıştır. Sistem video kaydetme, görüntü yakalama yapar, ardından bu verileri akıllı bir telefona aktarır. Her iki tarafta güvenilirlik ve gizlilik sunması avantaj sağlamıştır (Prasad ve ark., 2014).

Zhao ve ark. (2004), yapay sinir ağı kullanarak çoklu özelliklere dayalı yeni bir yüz tanıma yöntemi önermişlerdir. Yüz tanımayı gerçekleştirmek için sinir ağlarını kullanılmıştır. Birçok denemeden sonra, çoklu özelliklere dayalı yüz tanıma yöntemi, tek özellik kullananlara göre daha yüksek bir sınıflandırma doğruluğuna sahip olduğu gösterilmiştir (Zhao ve ark., 2004).

Bir yüz algılama ve tanıma sisteminde, aydınlatma ve yüz ifadeleri, oklüzyonlar ve yüzdeki pozlar gibi her bir bireyin tanımlanması ve doğru sınıflandırılmasını etkileyen çok sayıda dış etkenle uğraşmak zorundadır (Shaheen ve ark., 2016).

Bir yüz algılama ve tanıma sisteminde, aydınlatma ve yüz ifadeleri, oklüzyonlar ve yüzün açılı duruşu gibi her bir dış etken algılama ve doğru sınıflamayı zorlaştırır. (Syafeeza ve ark., 2014). Ayrıca, erişim kontrol sistemleri, suçlu tanımlama veya banka kontrolleri gibi biyometrik sistemleri kullanan önemli uygulamalar için bu sistemlerin yüksek doğruluk ile çalışması beklenir (Dunstone ve Yager, 2008)

Guevara ve ark. (2008) tarafından yüz algılama için ardışık sınıflandırıcı ve haar tabanlı filtreler kullanılmıştır. Ahmad ve ark. Evrişimsel Sinir Ağları (CNN) kullanarak yüz algılamada %100 başarı sağlamıştır (Syafeeza ve ark., 2014). Artırılmış gerçeklik tabanlı eğitim veritabanı oluşturan Martinez ve Benaverta çalışmalarında %99,5 doğruluğa ulaşmıştır. Lawrence ve ark. (1997), Sharma ve diğ. (2016) ve Li ve diğ. (2015) çalışmalarında yüz algılama algoritmaları evrişimsel sinir ağları ile geliştirilmiş, yüzlerin tamamen profildeki pozisyonlarla veya ifadenin değişmesi halinde dahi tanımlanması sağlandı (Arun ve ark., 2014).

Yüz sınıflandırma alanında, aydınlatma, yüz ifadesi ve yüzün bakış yönü değiştiği durumlarda bile sınıflama yapılan uygulama (Liu ve ark., 2017) tarafından geliştirilmiştir. Diğer bir çalışmada da cinsiyet, ırk ve yüz ifadesinin sınıflandırılması Lyons ve ark (1999) tarafında gerçekleştirilmiştir

Biyometri alanında, (Fox ve ark., 2007) tarafından yapılan çalışmada kişilerin tanımlanması için ses ve yüz tanıma bilgilerini bir arada kullanan kimliklendirme sistemleri geliştirilmiştir. (Filković ve ark., 2016) ve (Konen ve Schulze-kruger, 1995) tarafından yapılan çalışmalarda istenen kişiyi birden fazla kamera ile takip eden sistem geliştirilmiştir. Kohen ve ark. tarafından yüz görüntülerini yarı otomatik konsol üzerinden alınarak evrişimsel sinir ağı ile değerlendirildiğinde %96 doğrulukla ve 3,5 sn ortalama sürede yüz tanıma yapan yüz algılamalı erişim kontrol sistemi geliştirilmiştir.

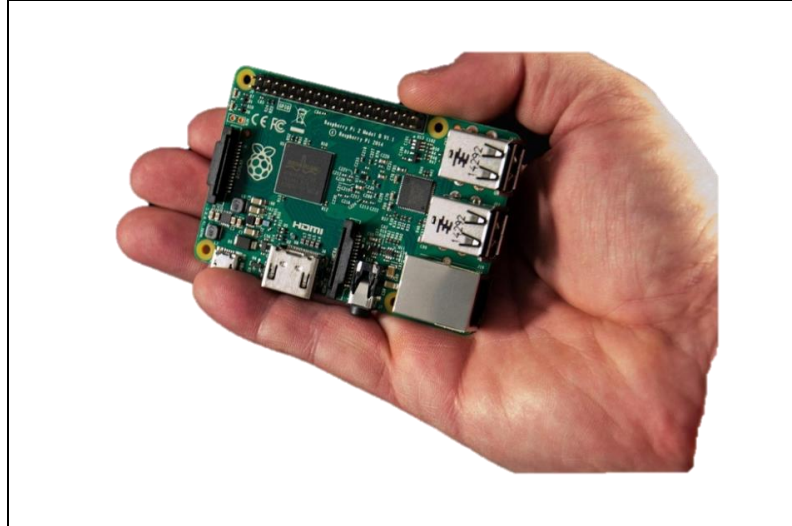
(Useche ve ark., 2018) çalışmalarında bir videodan yüzlerin algılandığı ve izlendiği bir yüz tanıma erişim kontrol sistemi geliştirilmiş, kullanıcılar evrişimsel sinir ağlarını kullanarak gerçek zamanlı olarak tanımlanmış ve %96 doğruluk oranının elde edilmiştir.

3. MATERYAL VE YÖNTEM

Bu tez çalışması yazılım ve donanım olmak üzere iki kısımdan oluşmaktadır. Yazılım bölümünde Visual studio C# üzerinde yüz tanımda kullanılan algoritma, karşılaştırma ve performans analizleri için kodlama yapılmıştır. Visual Studio üzerinden kamera görüntüsünün alınabilmesi ve karşılaştırma sonuçlarının gösterilmesi için bir arayüz tasarlanmıştır. Microsoft Azure bulut sisteminde veritabanı oluşturularak kamera üzerinden alınan görüntüler bu veritabanı ile karşılaştırılmıştır. Tüm kodlama sistemi Raspberry PI3 platformu üzerine gömülerek çevre birimler ile iletişim protokolleri sağlanmıştır. Donanım kısmı ise Raspberry Pi 3 ve kapı kiliti platformu olmak üzere iki kısımdan oluşmaktadır. Raspberry Pi 3 platformu SD kart, kamera, hoparlör, klavye, mouse, ekran ve zil kısmından oluşmaktadır. Kapı kilit platformu ise Raspberry Pi 3 platformuna entegre edilen 12V 3A güç kaynağı, dişi DC güç adaptörü, 5V elektromekanik kapı kilidi, anahtarlama için 5V iki kanal röle sisteminden oluşmaktadır. Gelecek bölümlerde bu sistemler ve sistem parçaları detaylı olarak tanıtılmıştır.

3.1 Raspberry Pi 3 Platformu

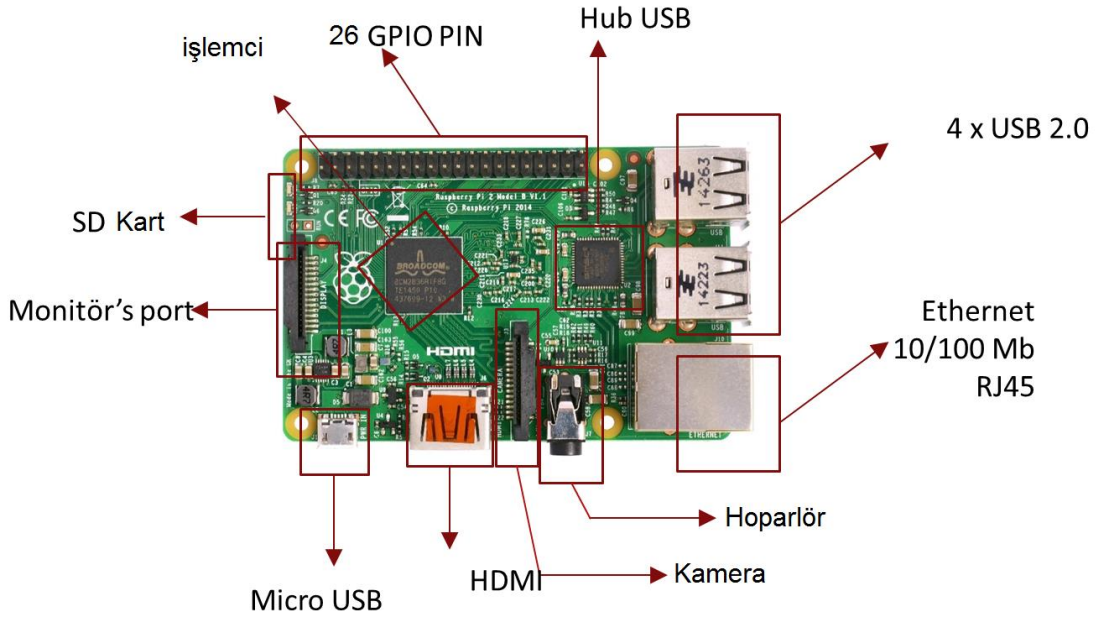
Raspberry Pi gömülü sistem platformu İngiltere’ de Raspberry Pi vakfı tarafından geliştirilen, asıl amacı okullarda bilgisayar temelli eğitimi geliştirmek ve çocuklara bilgisayarı öğretmek olan mini bir bilgisayardır. Raspberry Pi kredi kartı büyüklüğünde olan içinde çeşitli Linux ve Android dağıtımları kurulabilen ARM mimarisine sahip fansız bir mini bilgisayardır. Kart üzerinde bulunan 26-pin GPIO bağlantı ucu sayesinde istenilen donanımlar Raspberry'e bağlanabilir. Kart üzerindeki pinler 8 GPIO, I2C, SPI, UART kısımlarından oluşmaktadır. Pinler üzerinden 5V ve 3.3V güç alınabilir. Raspberry Pi üzerinde kullanılabilen I/O pinlerinin lojik seviyesi 3.3V'dur. Çevre birimlerinin bağlanabildiği USB çıkışları ve ethernet çıkışı bulunmaktadır. Şekil 3.1’ de Raspberry Pi 3 modeli gösterilmektedir (Diener, 2016).



Şekil 3.1 Raspberry Pi 3 model B (Diener, 2016)

Raspberry Pi 3 özellikleri ve çevre birimi çıkışlarının detayları aşağıda verilmiştir (Avcı, 2016). Bu özelliklerin kart üzerinde dağılımı Şekil 3.2’ de işaretler ile gösterilmiştir.

- 1-Broadcom BCM2837 SoC
- 2-1.2 GHz 4 çekirdekli 64-bit ARM Cortex-A53 işlemci
- 3-2 çekirdekli Videocore IV® Multimedia işlemcisi
- 4-1 GB LPDDR2 bellek
- 5-Dahili 802.11b/g/n destekli WiFi
- 6-Bluetooth 4.1, low-energy destekli
- 7-10/100 Mbit destekli Ethernet portu
- 8-HDMI portu (HDMI 1.4 destekli)
- 9-Kompozit video ve ses çıkışı için 3.5mm TRRS (4 uçlu) konektör
- 10-4 adet USB2.0 portu
- 11-40 adet GPIO pini, önceki Raspberry Pi modelleri ile uyumlu
- 12-WiFi/Bluetooth için dahili çip anten
- 13-CSI (kamera) ve DSI (ekran) konektörleri
- 14-Mikro SD kart yuvası
- 15-Tüm Raspberry Pi uyumlu Linux dağıtımlarını ve Windows 10 IoT Core işletim sistemleri
- 16-Boyutlar: 85 x 56 x 17 mm



Şekil 3.2 Raspberry Pi 3 özellik dağılımlarının kart üzerinde gösterilmesi

3.1.1. SD Kart

Raspberry Pi, işletim sisteminin ilk aşamada çalışması için bir SD karta ihtiyaç duymaktadır. Sonraki aşamalarda tüm sistem, USB'nin diske yönlendirilmesi durumunda dahi ilk açılışı SD karttan yapmaktadır. Bu nedenle cihazın kullanımı için bir SD kart gerekmektedir. Tüm işletim sistemi ve okuma / yazma işlemleri göz önüne alındığında SD kartın en az 4GB veya sınıf 10 olması ile bu sayede hız ve alanla ilgili sorunların önüne geçilebilmektedir (Kılıçdağı, 2014).

3.1.2. Yazılım Desteği

Raspberry Pi içerisine çeşitli yazılımlar yüklenebilir. Bu yüklenebilecek yazılımlar şu şekilde sıralanabilir:

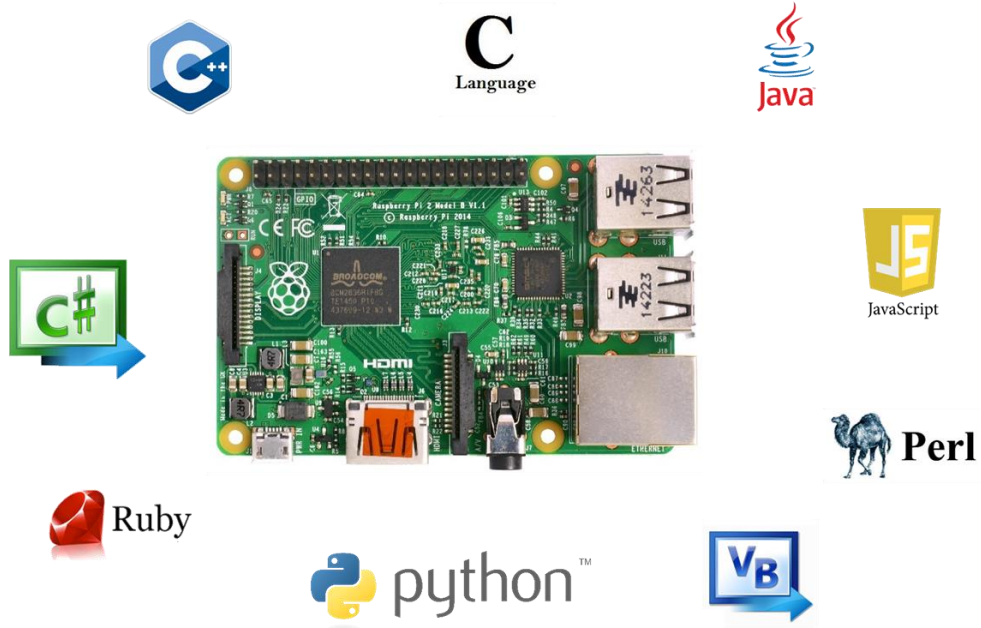
- Python
- C language
- Java
- C#
- Perl
- Ruby

- VB
- JavaScript
- C++ (Şekil 3.3).

Raspberry Pi çalıştırmak için kullanılan işletim sistemleri şu şekildedir:

- Rasbian
- Windows 10 Core IOT
- Noobs

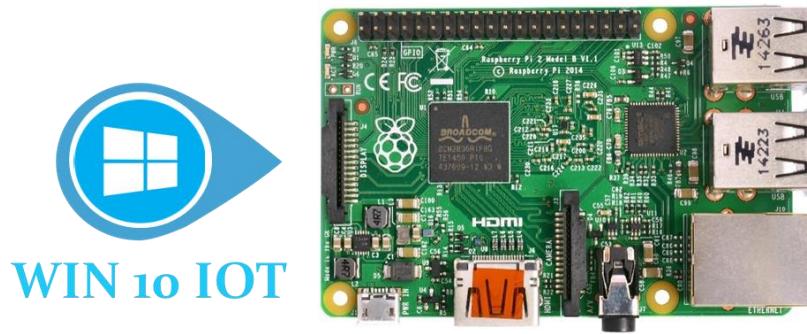
Rasbian kapsamında eğitim, programlama ve genel kullanım için birçok yazılım bulunmaktadır. Python, Çizik, Sonic Pi, Java, Matematiksel vb. programlama dilleri sistem içinde mevcuttur. ZIP arşivinde bulunan Rasbian boyutu 4GB' ın üzerindedir (Raspberrypi, 2015).



Şekil 3.3 Raspberry Pi içinde kullanabilen programlar

3.1.3. Windows 10 Core IoT

Windows 10 Core IoT, Windows 10'un en küçük sürümüdür ve her sürümde daha önceki sürümlere destek sağlayabilecek şekilde tasarlanmıştır. Windows Embedded, Microsoft' un gömülü sistemlerinde kullanılmak üzere tasarlanmış olduğu bir işletim sistemidir. Windows 10 Core IoT, modern uygulama altyapısını IoT cihazlara taşımakta, süreci hızlandıran bir işletim sistemi olmaktadır (Petrini ve Ionescu, 2016).



Şekil 3.4 Raspberry Pi Win 10 IoT

3.2. Microsoft Azure nedir?

Azure, Bilgi Teknolojileri (BT) uzmanlarının ve geliştiricilerin veri oluşturmak, dağıtmak ve yönetmek için kullandığı kapsamlı bir bulut hizmetleri koleksiyonudur. Tümleşik araçlar, basit mobil uygulamalardan internet ölçeğinde çözümlere kadar her isteneni etkili bir şekilde oluşturmaya yardımcı olur (Azure, 2018).

Mobil DevOps'tan sunucusuz bilgi işleme kadar çeşitli Azure tümleşik araçları üretkenliğe destek olmaktadır. Araçlar ve açık kaynak teknolojileri kullanılarak uygulamalar oluşturulabilmektedir. Azure; bir dizi işletim sistemi, programlama dili, altyapı, veritabanı ve cihazı destekleyebilmektedir.

Başlıca mobil platformları için sağlanan destekle cihazlar arası deneyimler gerçekleştirilebilmektedir. Sistem içerisinde Linux veya Windows tabanlı tüm yığınları çalıştırılabilmekte, Azure Container Service'te Kubernetes kümesi gibi gelişmiş özellikler kullanılabilmektedir (Azure, 2018).

Azure sistemi hibrit bir bulut yapısı olup, verilerin istenilen yerde ve zamanda oluşturulabilmesini sağlar. Bulut içinde veriler ve uygulamalar arasında bağlantı kurularak, taşınabilirlik en üst düzeye çıkarılabilmektedir. Azure; uygulama geliştirme,

yönetim ve güvenlik, kimlik yönetimi alanlarında ve veri platformunun tamamında hibrit tutarlılık sunmaktadır (Azure, 2018).

Azure'u kullanarak veriye dayalı, akıllı uygulamalar oluşturulabilmektedir. Resim tanımadan robot hizmetlerine kadar çeşitli Azure veri hizmetleri sunulmaktadır. Bu noktada yapay zekâdan yararlanarak ölçeklenebilen yeni deneyimler tasarlanabilmektedir. Ayrıca bu sistem; derinlikli öğrenmeyi, HPC simülasyonlarını, her türden ve her boyutta veri üzerinde gerçek zamanlı analizi gerçekleştirebilmektedir. İstenilen veriler üzerinde özel yapay zekâ modelleri oluşturulabilmekte, ayrıca Microsoft'un ve açık kaynak verilerinin en iyi yönleri yapay zekâ yenilikleriyle birleştirilebilmektedir (Azure, 2018).

3.3. Microsoft Life Cam HD-3000

Siyah plastik HD-3000, 1/7-inç oval gövdeye oturan bir cam mercekle içermektedir. Kamera, çerçevede pozisyonu ayarlamak için sadece sağa ve sola döndürülebilmektedir. Kamera, farklı parlaklık koşulları için video parlaklığını ve rengini optimize eden LifeCam Studio ile aynı TrueColor teknolojisine sahiptir (Microsoft, 2016).



Şekil 3.5 Microsoft Life CAM HD-3000 (Microsoft, 2016)

Microsoft Life CAM HD-3000 özellikleri Çizelge 3.1'de görüldüğü gibidir (Microsoft, 2016).

Çizelge 3.1 Microsoft Life CAM HD-3000 Özellikleri

<i>Uygulamalar</i>	<i>Multimedya ve İnternet Uygulamaları</i>
Video çözünürlüğü	HD 720p (1280x720)
Sensör	720p HD
Mikrofon	Gürültü Önleyici Mikrofon
Mac / PC	Windows® 7, Windows Vista®, Windows XP (Service Pack 2 (SP2) veya üstü)
Skype sertifikalı	Evet
Skype uyumlu	Evet
Menzil / USB Kablosu	USB 2.0

3.4. Elektrik Kapı Kilidi

Elektrik kapı kilidi, erişim kontrol okuyucusu ile birlikte kullanıldığında kapıyı kitleme ve açma özelliklerine sahiptir. Güç bağlandığında kapının kilidini açan ve elektrik kesildiğinde kapıyı kilitleyen hata korumalı (fail secure) cihazı sistem içinde mevcuttur. Kapı, bir anahtarla açılabilirdiği gibi sistem tarafından otomatik olarak da açılabilir (Amazon, 2018).

Kapı kilidi Şekil 3.6’da sunulmakta olup, kilide dair özellikler Çizelge 3.2’ de görüldüğü gibidir.

**Şekil 3.6** Elektrik kapı kilidi**Çizelge 3.2** Elektrik kapı kilidi özellikleri

Ürün Tipi	Bileşen
Satış Kategorisi	Elektrik Gücü
Gerilim ve Görev	12 Volt DC Sürekli Görev

3.5 5V 2-Kanallı Röle Modeli

Android röle modülü; Arduino, Raspberry Pi, AVR, PIC gibi mikro denetleyicilerle birlikte kullanılabilir. 2-kanallı röle modülü Şekil 3.7' de sunulmaktadır (Essabre-90xx-rpi,2018).



Şekil 3.7 5V 2-Kanallı röle modülü (Essabre-90xx-rpi, 2018)

Röle Modülüne Dair Özellikler:

- Model: SRD-05VDC-SL-C
- Rölelerin Sayısı: 2
- Kontrol sinyali: TTL seviyesi
- Nominal yük: 7A / 240V AC, 10A / 125V AC, 10A / 28V DC
- Temas süresi: 10ms / 5ms

Tanımlama:

- 2 Sinyal rölesi
- Arabirim tanımı
- IN (x): Yüksek seviye / düşük seviye kesme
- Her röle için LED göstergesi
- FR-4 fiber cam PCB (Çift Katmanlı).
- PCB kalınlığı: 1.6mm.
- Güç Kaynağı: 5V DC / 40mA (hepsi röle açık)
- Boyut: 54 x 46 x 16 mm

Giriş Kontrol Sinyal Voltajı:

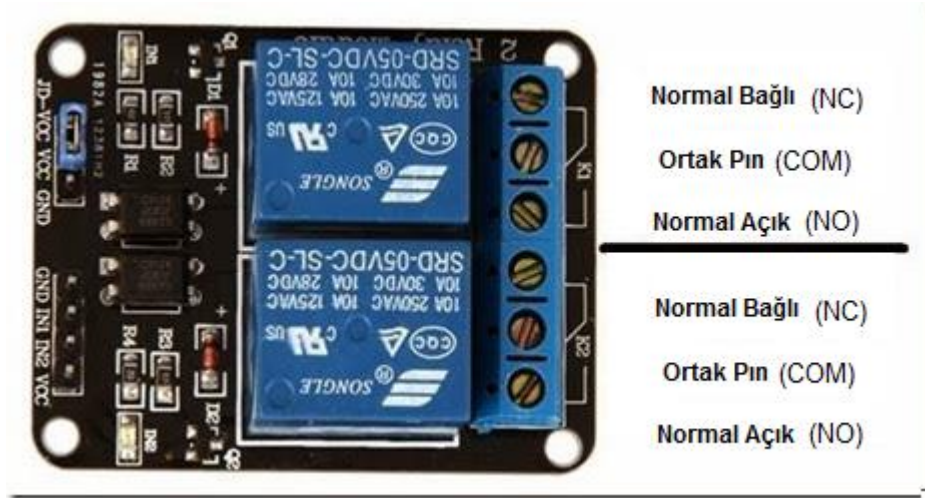
- 0V - 0,5V Düşük kademe (röle kapalı),
- 0.5V \pm 2.5V (bilinmeyen durum).
- 2.5V - 5V Yüksek durum (röle açık)

Giriş kontrol sinyali yüksek durum akımı:

- 2.5V: 0.1mA
- 3.3V: 0.18mA
- 5V: 0.35mA
- Voltaj: 5V DC.
- Akım: 8.3mA.
- Direnç: 2880 ohm +-10%.

İletişim değerlendirilmesi:

- Temas Kapasitesi: 125V AC / 1A, 30V DC / 2 A
- Temas Dayanımı: <0.1 ohm.



Şekil 3.8 Rölenin pin tanımı (Essabre-90xx-rpi,2018)

3.6. Logitech C270 Kamera

Logitech kamera HD video arama ve kayıt için tasarlanmıştır. Kamera geniş ekranı, 3 MP çözünürlük, 2,4GHz intel core 2 duo, 2 GB RAM, 200 MB sabit bir disk

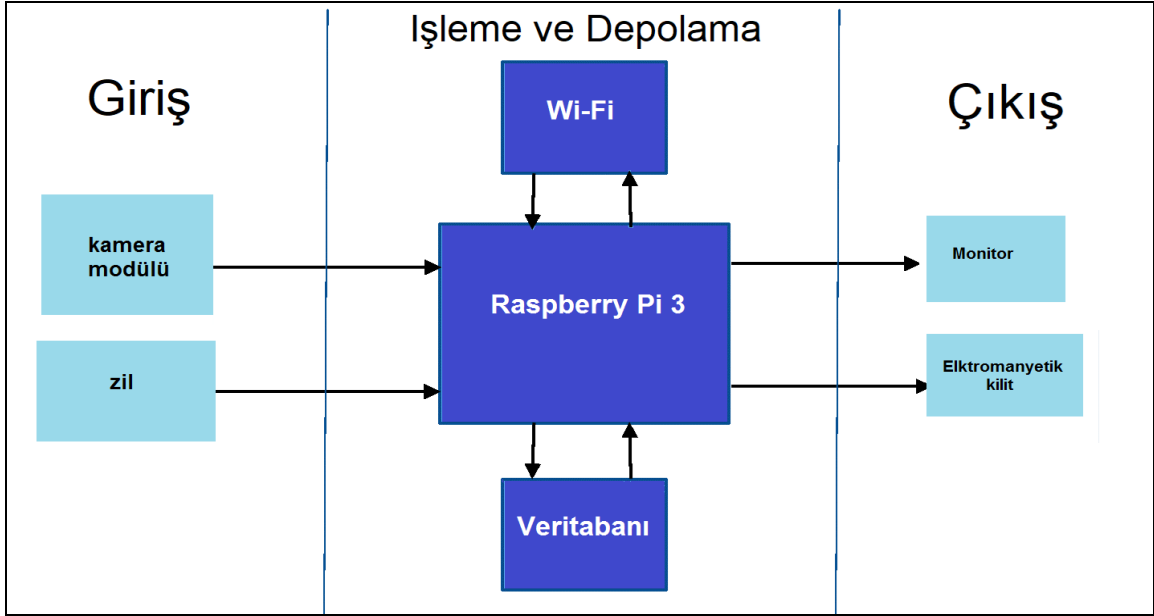
alanına sahiptir. Video kayıt doğruluğu 1280 x 720' dir. 3 MP .Net fotoğraf teknolojisine sahip, Windows ile uyumlu ve USB 2.0 hızına sahip bir kameradır. Windows7 (32 bit veya 64 bit) veya Windows 8 ile çalışmakta ve desteklenmektedir. Şekil 3.9' da tez çalışmasında kullanılan Logitech C270 Kamera sistemi gösterilmiştir (Biosis, 2018).



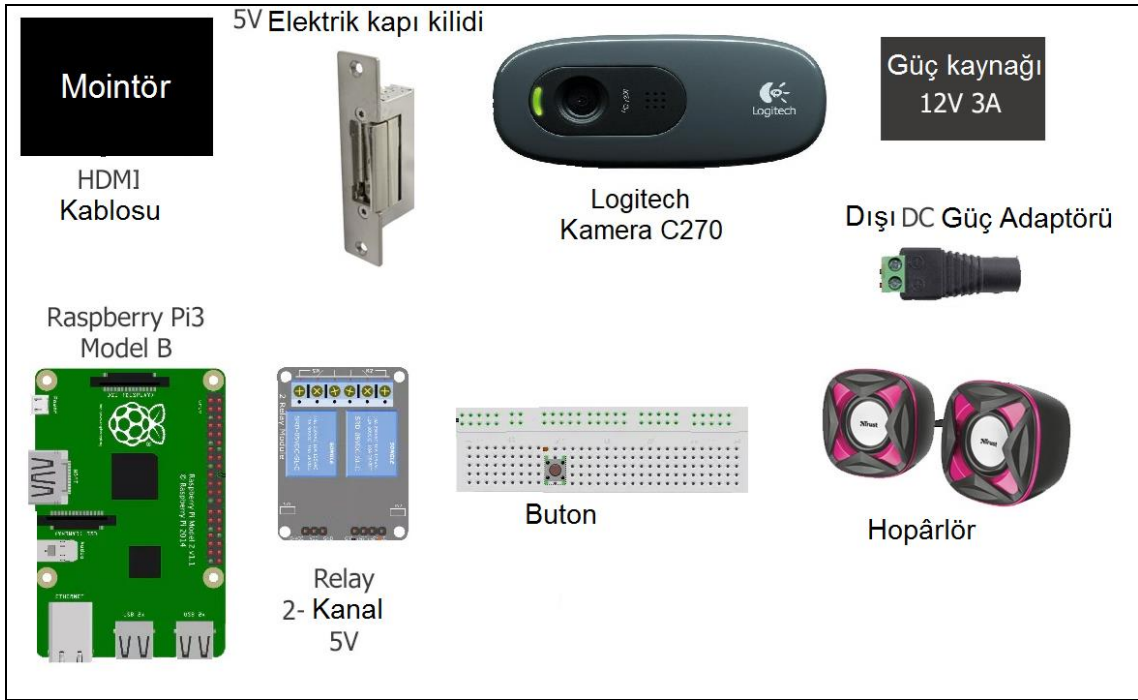
Şekil 3.9 Logitech C270 kamera (Biosis, 2018)

3.7. Sistem Mimarisi

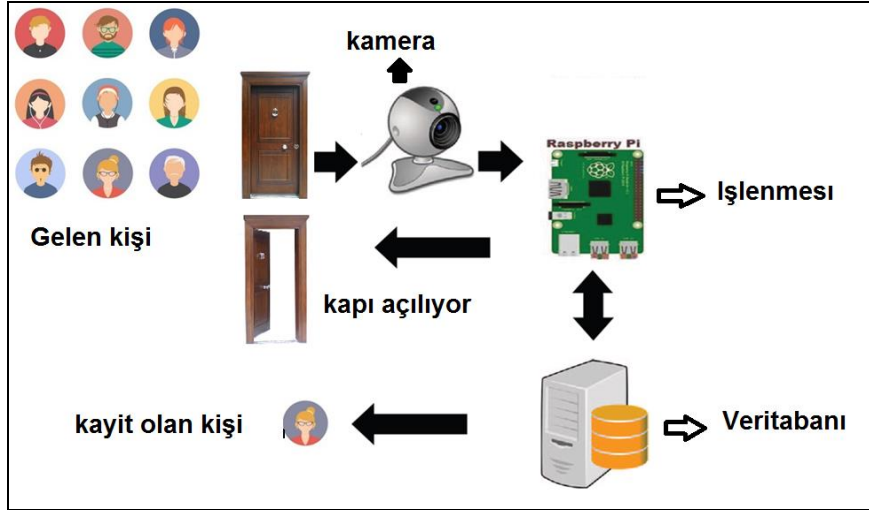
Sisteme giriş olarak kameradan yüz tanıma için görüntü alınmıştır. Raspberry Pi içerisine C# hazırlanan görüntü işleme kodu ile yüz tanıma yapılmıştır. Tanıma yapılan yüzlerin Microsoft Azure veritabanında kayıtlı görüntüler ile karşılaştırması yapılmıştır. Şüpheli yüz tanıma durumlarında Raspberry Pi 3 üzerindeki Wifi modül ile internete bağlanılarak sistem yöneticisine mail atmaktadır. Eğer yöneticinin izin verdiği veya şüpheli olmayan bir yüz tanınması durumunda çıkıştaki manyetik kapı açılmaktadır. Aynı zamanda hoparlör sistemi ile tanınan kişiye ismiyle hitap ederek karşılama yapmaktadır. Tasarlanan sistemin genel çalışma şekli Şekil 3.10'da, mimarisi Şekil 3.11'de gösterilmektedir. Sistemin çalışması Şekil 3.12'de görsel olarak sunulmuştur.



Şekil 3.10 Sistemin genel çalışma şekli



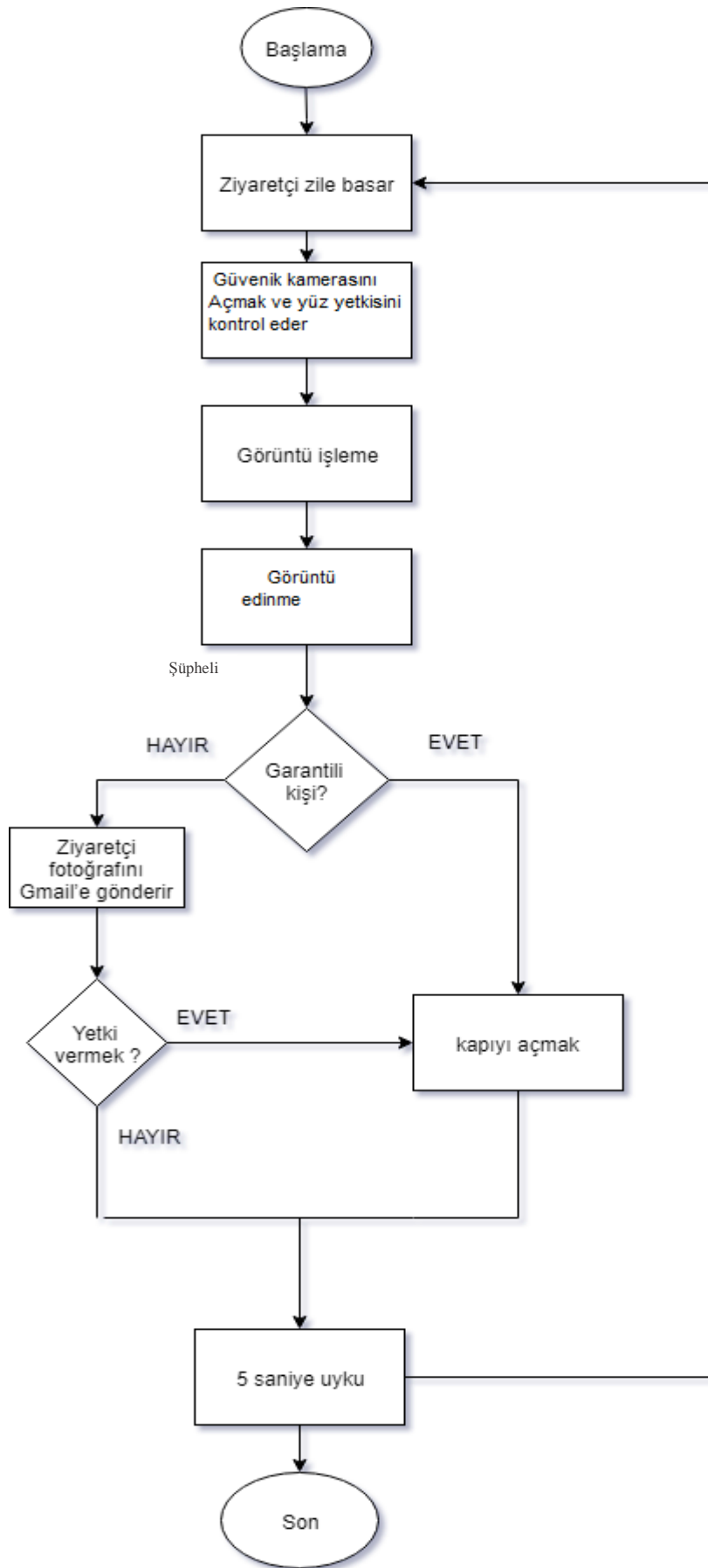
Şekil 3.11 Raspberry Pi 3 ve çevre donanım birimleri ile tasarlanan sistem mimarisi



Şekil 3.11 Sistem açıklaması

3.8. Genel Kontrol Sistemi ve Algoritmasının Akış Şeması

Genel sistem blok diyagramında olduğu gibi, uygulanan ilgili kontrol algoritması, akış şeması ile birlikte aşağıda tarif edilmiştir. Algoritma şemasında sistemin adımları sunulmaktadır:



Şekil 3.12 Sistemin akış şeması

Ziyaretçi düğmeye (zil) bastığında, ziyaretçinin yüzü kamera tarafından yakalanır ve karşılaştırma işlemi için MS Azure veritabanına gönderilir. Tespit, kırpma, tanıma ve karşılaştırma veritabanında ele alınır. Veriler, MS Azure veritabanında işlendikten sonra, sonuç kesin bir değerle JSON formatında sisteme gönderilir. Bu değere bağlı olarak ziyaretçinin bilinip bilinmediğini MS Azure tarafından tespit edilir.

Ziyaretçi veritabanında kayıtlı ise yani tanınıyorsa kapı açılır. Aksi halde, alarm mesajı güvenlik kontrolü için sistem tarafından kullanıcıya gönderilir. Kullanıcı kapıyı uzaktan açma imkânına sahiptir. Kullanıcı gelen ziyaretçiyi kabul etmek isterse sisteme geri mesaj göndererek kapıyı açar. Kabul etmek istemezse mesajı görmezden gelebilir. İşlem bittikten sonra, sistem 5 saniye sonra tekrar uyku moduna geçer.

3.9. Yüz görüntülerinin alınma yöntemleri

Yüz görüntülerinin alınma yöntemleri, temel alınan uygulamaya bağlıdır. Örneğin, görüntü veri tabanı araştırmaları standart bir kamera tarafından alınan statik yoğunluk modellemesinde alınmasını gerektirirken, gözetim uygulamalarında bir video kamera vasıtasıyla yüz görüntülerinin yakalanması suretiyle en iyi şekilde çalışmaktadır. Üst düzey güvenlik alanlarına erişim gibi bazı uygulamalar, kullanıcının bir 3B tarayıcı veya kızıl ötesi sensör önünde durmasını gerektirmektedir. Böylece yüz tanıma özelliğinin kalitesini sürdürebilmesi sağlanır. Dolayısıyla, yüz veri toplama yöntemine bağlı olarak, yüz tanıma teknikleri üç ana kategoriye ayrılabilir: Yoğunluk modellemesi üzerinde çalışan yöntemler, video dizileri ile uğraşan yöntemler ve 3B bilgi veya kızıl-ötesi görüntüler gibi diğer sensör / veri gerektiren yöntemler.

3.10 Sistem Tanımı

3.10.1 Kullanılan Yazılım

Bu sistemde kilitleme / açma kapılarını kontrol etmek için bir Raspberry Pi paneli üzerinde işletim sistemi olarak NOOBS kullanılmıştır. NOOBS Raspberry Pi donanımı üzerinde yüklü bir işletim sistemidir ve raspbian tabanlıdır. Bilgisayardaki farklı işletim sistemine bağlanmada uyum problemi bulunmamaktadır. Ayrıca diğer

işletim sistemlerinin bir listesini sunar. Böylece kullanılacak programlara göre farklı bir işletim sistemine geçmemize olanak sağlar (Raspberrypi, 2015).

3.10.2 İnsanlarda Yüz Tanıma

Yüz tanıma, bugüne kadar en çok uğraşılan örüntü tanıma problemidir. Fakat bu problemin ne kadar zor olduğunu ilk psikologlar fark etmiştir. Bu yüzden de insan beyninde bu problemin nasıl çözüldüğü pek çok araştırmaya konu olmuştur. İnsan beyni bir ya da iki kere gördüğü bir insan yüzünü aradan uzun bir zaman geçtikten sonra, farklı bir ışık altında, değişik bir pozda görse bile hatırlar. Henüz bilgisayar sistemleri insan beyninin bu başarısını gösteremezler. Bu yüzden de bilgisayarlar için birçok yüz tanıma sistemi geliştirilmiştir (Sinha ve ark., 2006).

3.10.3 Yüz Tanıma Sistemleri

Biyometrik sistemler son yıllarda geniş bir uygulama alanına sahip olmuştur. Amerikan pazar araştırma şirketi Global Industry Analysts'in yeni Pazar araştırma çalışmasına göre, global biyometrik sanayi pazarının 2017 yılında 16.47 milyar doları aşması öngörülmüştür (planetbiometrics, 2011).

Başlıca biyometrik teknolojilerden biri olan yüz tanıma, görüntü yakalama araçlarındaki (gözetim kameraları, cep telefonlarındaki kameralar, vb...) hızlı gelişmeler, web ortamında çok büyük miktarlarda yüz görüntülerinin bulunması ve yüksek güvenliğe olan ihtiyacın artması sonucunda, giderek daha önemli bir hale gelmiştir (Jain ve Li, 2011).

RNCOS araştırma şirketinin Mart 2011'de yayınlanan 2012 Yılı Küresel Biyometrik Pazarı Beklenti Raporu'nda, (Global Biometric Forecast to 2012) yüz tanıma teknolojisinin dünya çapında kabul edilen biyometrik teknolojiler arasında en hızlı büyüyen teknoloji olarak ortaya çıktığı belirtilmiştir. Ayrıca bu teknolojinin 2011- 2013 döneminde yıllık bileşik büyüme oranı (CAGR) yaklaşık %31 olacağını ve gelecekte de aynı trendi takip edeceğini öngörülmektedir (Rncos, Jul 20, 2011).

Yüz tanıma sistemleri, kişileri belirlemek için kişilerin yüz özelliklerini kullanır. Genellikle bu sistemler yüz görüntülerinden belirli bazı özellikleri seçer ve ardından bu özelliklerini kullanarak yüz eşlemeyi gerçekleştirir. Bir yüzün belirli özellikleri; gözler

arasındaki mesafe, burun, elmacık kemikleri, çene çizgisi, çene ve benzeri pozisyonların genişliğini içerir (Adeoye, 2010).

Yüz tanıma sistemlerinin avantajı görüntü yakalama cihazı (kamera) ile herhangi bir fiziksel temas gerektirmemesidir. Bir yüz tanıma sistemi gelişmiş bir donanıma gerek duymamaktadır. Bir sahnenin tüm yüzlerinin tanındığı ve yüz özelliklerine göre bireysel erişimi engellemek veya reddetmek üzere sınıflandırıldığı konvolüsyonel nöral ağları kullanan bir yüz erişim kontrol sisteminin geliştirilmesini sunmaktadır. Sistem, bir video dizisinde 5 saniyeliğine ayrı ayrı tanıdıktan sonra iki özel kişinin erişimine izin verir ve görüş alanında bulunan bir dış insanın varlığı tespit edildiğinde erişimi engeller. Program, haar sınıflandırıcıları tarafından bir yüz algılama sistemi, KLT Algoritması (Kanade-Lucas-Tomasi) tarafından bir nokta takip sistemi ve doğruluk yüzdelerinin% 96'nın üzerine ulaştığı konvolüsyonel sinir ağları tarafından bir sınıflandırma tekniği kullanılmaktadır(Useche ve ark., 2018).

3.11 Yüz Tanıma İçin Farklı Teknikleri

3.11.1 Farklı Yaklaşım

Durağan görüntüden yüz tanımada temel olan sabit resimden ya da videoda yüz tanıma yapılabilir. Hareketsiz görüntüler için farklı yüz tanıma yaklaşımları, üç ana grupta ele alınabilir (Mehrdad, 2013):

1. Bütünsel yaklaşım
2. Özelliğe dayalı yaklaşım
3. Karma yaklaşım

3.11.2 Bütünsel Yaklaşım

Bütünsel yaklaşımda, tüm yüz bölgesi, yüz algılama sistemine girdi verileri olarak kullanılmaktadır. Bütünsel yöntemin önde gelen alt dalları eigenfaces (Yüz tanımada en çok kullanılan yöntem), olasılıksal özyüzler, fisher yüzler, destek vektör makineleri, en yakın özellik hatları (NFL) ve bağımsız bileşen analizi yaklaşımlarıdır (Mehrdad,2013).

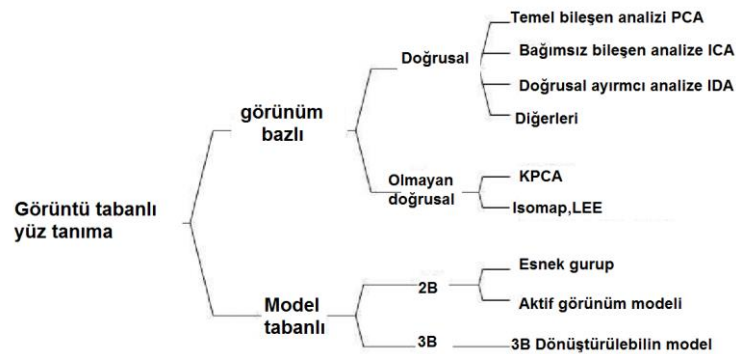
3.11.3 Özellik Temelli Yaklaşım

Özellik temelli yaklaşımlarda, yüzdeki yerel özellikler bulunur. Bu özellikler içinde önce burun, sonra gözler bölümlere ayrılır ve daha sonra yapısal sınıflandırıcı için girdi verileri olarak kullanılır. Sentetik geometri (saf geometri), dinamik bağlantı mimarisi ve gizli markov modeli yöntemleri bu kategoriye aittir (Mehrdad, 2013).

3.11.4 Hibrid Yaklaşım

Bu yöntemin çalışma şekli insan görme sisteminin hem yüzü bütün olarak hem de özelliklerini nasıl algıladığı ile ilgilidir. Hibrid yaklaşım; modüler özyüzleri, hibrid yerel özelliği, şekil normalleştirmeyi, bileşen tabanlı yöntemleri kapsar. (Mehrdad,2013).

Yüz tanıma yöntemlerinin algoritmaya göre sınıflandırılması şekil 3.14'te belirtilmiştir.



Şekil 3.13 Bazı yüz tanıma yöntemleri (Mehrdad, 2013).

3.11.5 Özyüzler Yöntemi

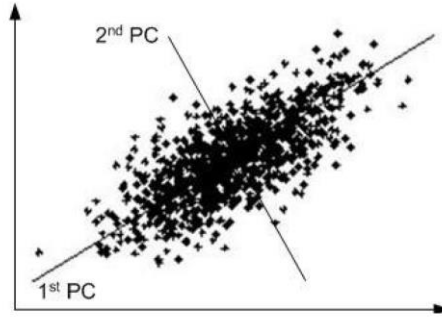
Özyüzler, çeşitli yüz görüntülerinin istatistiksel analizine dayanan bir standart yüz bileşenleri setidir. Matematiksel olarak açıklanacak olursa; olası insan yüzlerini temsil eden yüksek boyutlu bir vektörün kovaryans matrisinden türetilen özvektör setine önyüz denir. Örneğin, bir kişinin yüzünün bir kısmı bir önyüz modeli ile diğer bir kısmı başka bir önyüz modeli ile temsil edilebilir (Motoyama ve Kaneda, 1989).

Sisteme eklenen yeni yüz görüntüsü, görüntü ile ortalama arasındaki farkın çarpılmasıyla yüz uzayına yansıtılır ve sonuç her bir nokta özvektör ile çarpılır. Bu

işlem ile girdi yüz resminin temsil eden her bir önyüzün ağırlıklandırılmış katkısı ortaya çıkar ve yüz resimleri için temel seti olarak önyüzü ele alır. Her bir yüz sınıfından alınan öklid mesafesi, giriş görüntüsüne en iyi uyan sınıfı belirler. Özyüzler aracılığıyla sistem, yüzün varlığını da algılar. Yüz alanına yansıtılan yüz görüntüsü radikal olarak değişmezken, herhangi bir yüzey dışı görüntü oldukça farklı görünecektir. Böylece yüz görüntüleri ve yüz harici görüntüleri ayırt etmek kolaylaşır. Bu temel düşünceyi kullanarak, yüz alanı üzerine görüntü yansıtılır ve öklid uzaklığı, ayarlanmış giriş görüntüsü ve yüz alanı üzerine yansıtma arasındaki ortalamadan hesaplanır. Uzaklık “yüz alanı” olarak kullanılır ve bu sayede mesafe hesaplamasının sonucu bir “yüz haritası” ortaya çıkar. Mesafe hesaplamasında düşük değerler bir yüzün olduğunu gösterir (Ferdous ve Ahmed, 2008).

3.11.6 Temel Bileşen Analizi (PCA)

Temel bileşen analizi (PCA), Karhunen - Loeve'nin dönüşümünden elde edilir. Her bir yüzün bir görüntü kümesindeki bir s boyutlu vektör temsili göz önüne alındığında, temel bileşen analizi (PCA), temel vektör uzamda temel vektörleri maksimum varyans yönüne karşılık gelen bir t -boyutlu alt uzay bulmaya eğilimlidir. Bu yeni alt uzay, normalde düşük boyutludur ($t \ll s$). Görüntü elemanları rassal değişkenler olarak kabul edilirse, PCA temeli vektörleri dağılım matrisinin özvektörleri olarak tanımlanır. Eigenface algoritması, tüm görüntü alanı içinde yüz görüntülerinin dağılımını en iyi şekilde hesaba katan vektörleri bulmak için boyutsallık azaltımı için PCA kullanır. Bu vektörler yüz görüntülerinin alt uzayını tanımlar ve alt yüzeye yüz alanı denir. Eğitim setindeki tüm yüzler, yüz vektöründe her bir vektörün katkısını tanımlayan bir dizi ağırlık bulmak için yüz boşluğuna yansıtılır. Bir test görüntüsünü tanımlamak için, karşılık gelen ağırlık setini elde etmek için test görüntüsünün yüz uzayına yansıtılmasını gerektirir. Test görüntüsünün ağırlıklarını, eğitim setindeki yüzlerin ağırlıkları ile karşılaştırarak, test görüntüsündeki yüz tanımlanabilir. PCA'daki temel prosedür, Karhunen-Loeve dönüşümüne dayanmaktadır. Görüntü unsurları rassal değişkenler olarak kabul edilirse, görüntü bir stokastik sürecin bir örneği olarak görülebilir. PCA temeli vektörleri scatter matrix ST 'nin özvektörleri olarak tanımlanır (Blair, 2009).



Şekil 3.14 İki boyutlu noktaların temel bileşenleri (PC). Birinci ana bileşen, ortalama kare hatası anlamında 2B'den 1B'ye kadar optimal doğrusal boyut küçültme sağlar (Ferdous ve Ahmed, 2008)

3.11.7 Bağımsız Bileşen Analizi (ICA)

Bağımsız bileşen analizi (ICA), Gaussian olmayan şekilde tasarlanması ile PCA'dan ayrılır. ICA, girdi verilerindeki ikinci ve daha yüksek dereceden bağımlılıklarını en aza indirir ve verilerin (onlara yansıtıldığında) istatistiksel olarak bağımsız olduğu temelleri bulmaya çalışır. Bartlett ve arkadaşları, yüz tanıma görevi için iki ICA mimarisi geliştirmiştir. Birincisi, istatistiksel olarak bağımsız temel görüntüler, ikincisi ise faktöriyel kodu gösterimidir (Ferdous ve Ahmed, 2008).

3.11.8 Doğrusal Ayırıcı Analizi (LDA)

Hem PCA hem de ICA, yüz sınıfının bilgilerini kullanmadan yüz alanını oluşturur. Tüm yüz eğitim verileri bir bütün olarak alınır. LDA'da amaç, yüz vektör uzayını temsil etmenin verimli veya farklı bir yolunu bulmaktır. Fakat sınıf bilgisinin kullanılması tanımlama işlevlerinde yardımcı olur.

Doğrusal ayırıcı analizinde (LDA), sınıfların en iyi şekilde ayrılmasını sağlayan temeldeki vektörler bulunur. Bütün sınıfların tüm örnekleri için sınıflar arası dağılım matrisi S_B ve sınıf içi dağılım matrisi S_W tanımlanır. Amaç, S_W 'yi en aza indirirken S_B 'yi maksimize etmek, diğer bir deyişle, $\det | S_B(S_{\text{Sınıf arasında dağılım matrisi}}) | / \det | S_W(S_{\text{Sınıf içi dağılım matrisi}}) |$ oranını maksimize etmektir. Bu oran, yansıtılan matrisinin sütun vektörleri $(S_W^{-1} \times S_B)$ özvektörleri olduğunda maksimize olur (Lu ve ark., 2003).

3.11.9 Evrimsel Takip (EP)

Bir uygunluk fonksiyonunu en üst düzeye çıkarmak için en iyi projeksiyon eksenleri kümesini arayan ve aynı zamanda sistemin sınıflandırma doğruluğunu ve genelleme yeteneğini ölçen bir öz-uzay tabanlı adaptif yaklaşımdır. Bu problemin çözüm alanının boyutu çok büyük olduğu için, evrimsel takip (EP) olarak adlandırılan belirli bir genetik algoritma kullanılarak çözülür (Liu ve Wechsler, 2000).

3.11.10 Kernel Yöntemleri

Alt uzaydaki yüzün bakış yönü sınıflaması doğrusal olması gerekmez. Kernel yöntemleri, doğrusal yöntemlerin geliştirilmesidir. Bu doğrusal olmayan bakış yönü öğrenilmesi için doğrudan doğrusal olmayan bakış yönü şemaları araştırılmıştır (Yang, 2002).

3.11.11 İz Dönüşümü

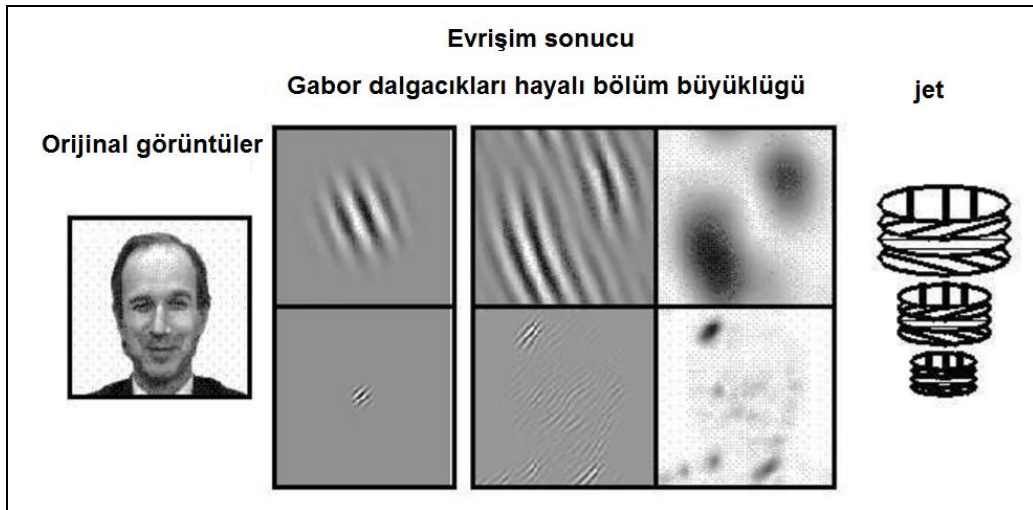
Radon dönüşümünün bir genellemesi olan iz dönüşümü; dönme, yer değiştirme ve boyutlama gibi dönüşümler altındaki nesnelere tanımak için kullanılabilen görüntü işleme için yeni bir araçtır. İz dönüşümü üretmek için, bir görüntünün izleme çizgileri boyunca işlevsel olarak hesaplanır. Farklı izleme işlevleri kullanılarak farklı iz dönüşümleri bir görüntüden üretilebilir (Goudelis ve ark., 2013).

3.11.12 Destek Vektör Makinesi (SVM)

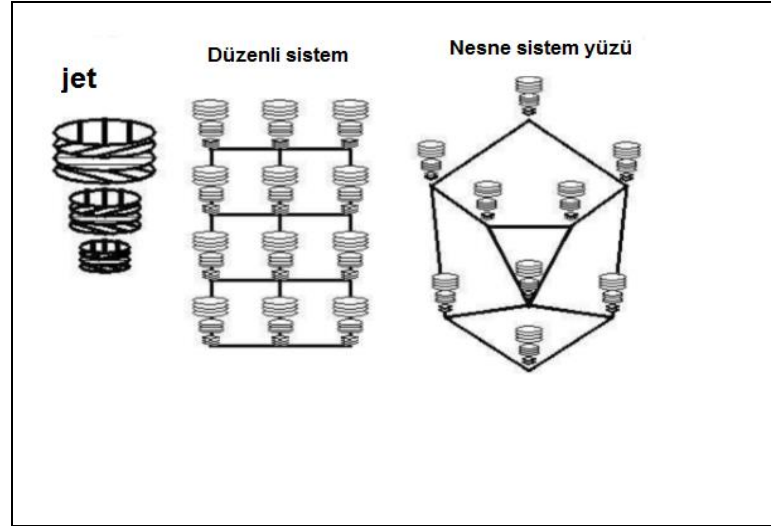
İki sınıfa ait bir dizi nokta verildiğinde, bir destek vektör makinesi (SVM), aynı sınıftaki noktaların mümkün olan en büyük fraksiyonunu aynı tarafta ayıran hiper düzlemi bulur ve her iki sınıftan hiper düzleme olan mesafe maksimize edilir. Yüz görüntülerinin özelliklerini çıkarmak için PCA kullanılır ve her bir görüntü çifti arasındaki ayırma işlevleri SVM'ler tarafından öğrenilir (Guo ve ark., 2000).

3.11.13 Elastik Demet Grafik Eşleştirme (EBGM)

Tüm insan yüzleri benzer bir topolojik yapıya sahiptir. Yüz, burun, göz köşeleri referans noktaları olarak konumlandırılmış düğümlerle grafik olarak temsil edilir ve köşeler $2B$ mesafe vektörleri ile etiketlenir. Her bir düğüm, farklı ölçeklerde ve faz, genlik gibi yönelimlerde 40 karmaşık gabor dalgacık katsayısı kümesi içerir. Bu katsayılara "jets" denir. Tanımlama, etiketli grafiklere dayanmaktadır. Etiketli grafik, köşeler tarafından bağlanan bir dizi düğümdür. Bu düğümler jets ile etiketlenir. Kenarlar da uzaklıklarla etiketlenir (Wiskott ve ark., 1997).



Şekil 3.15 Konvolüsyon sonucu (Ferdous ve Ahmed, 2008)

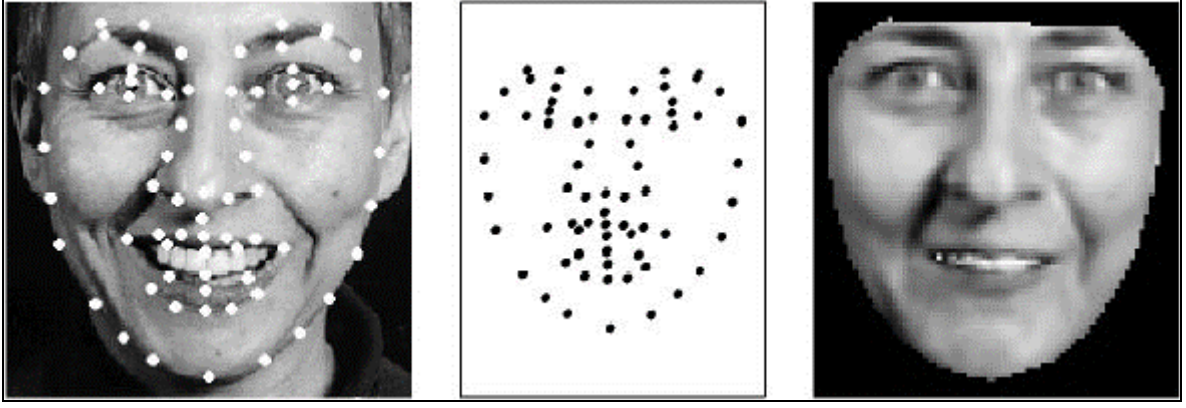


Şekil 3.16 Etiketli grafik (Ferdous ve Ahmed, 2008)

3.11.14 Aktif Görünüm Modeli (AAM)

Bir aktif görünüm modeli (AAM), bir yüz modelinin çeşitli şekilleri ile şekil normalleştirilmiş bir çerçevede görünüm varyasyonlarının birleştirildiği entegre istatistiksel modeldir. Bir AAM, herhangi bir örneğe genelleme yapabilen, ilgilenilen nesnenin şekli ve gri-düzey görünümü varsa, bir istatistiksel model içerir. Bir görüntü eşleme, görüntü ile görüntüye yansıtılan sentezlenmiş bir model örneği arasındaki farkı en aza indiren model parametreleri bulmayı içerir.

AAM, ana özelliklerin ana hatlarını çizmek için her örnekte kilit noktalarda işaret noktalarının işaretlendiği işaretlenmiş görüntülerin bir eğitim setine dayanılarak oluşturulmuştur (Edwards ve ark., 1998).



Şekil 3.17 Eğitim görüntüsünün şekil ve normalleştirilmiş şekilli dokuya ayrılması (Ferdous ve Ahmed, 2008)

3.11.15 3B Dönüştürülebilir Model

İnsan yüzü, 3B uzayda tanımlanabilen bir yüzeydir. Bu nedenle, 3B model, özellikle poz, aydınlatma gibi yüz değişimlerini ele alırken yüzleri temsil etmede en etkili bir modeldir. Blantz ve arkadaşları, model parametreleri açısından şekil ve dokuyu kodlayan bir 3B dönüştürülebilir yüz modelini ve bu parametreleri bir yüz görüntüsünün tek bir görüntüsünden alan algoritmayı önermiştir (Blanz ve Vetter, 2003).

3.11.16 3B Yüz Tanıma

Bu yaklaşımın ana yeniliği, yüz ifadelerinden kaynaklanan doğal deformasyonlardan bağımsız olarak yüzeyleri karşılaştırma özelliğidir. İlk olarak, yüzün görüntüsü ve yüzün dokusu elde edilir. Ardından, uzaklık imgesi, tanıma sürecini zorlaştırabilen saç gibi belirli bölümlerin çıkarılması ön-işlemlerinden geçirilir. Son olarak, yüz yüzeyinin kanonik şekli hesaplanır. Böyle bir temsil, kafa oryantasyonlarına ve yüz ifadelerine karşı duyarsızdır, dolayısıyla tanıma prosedürünü önemli ölçüde basitleştirir. Tanıma, kanonik yüzeyler üzerinde gerçekleştirilir (Papatheodorou ve Rueckert, 2007).

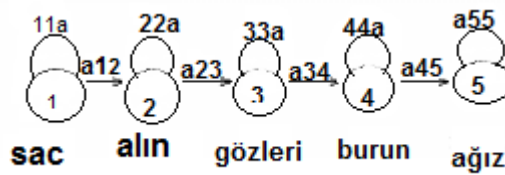
3.11.17 Bayes Çerçeve

Bayes'e dayalı olasılıksal benzerlik ölçüsü, görüntü yoğunluğu farklılıklarının, bir bireyin görünüşündeki tipik varyasyonların karakteristiğidir. İki sınıf yüz resmi varyasyonu tanımlanmıştır, içsel (intrapersonal) varyasyonlar ve ekstra kişisel varyasyondur. Yüzler arasındaki benzerlik, Bayes kuralını kullanarak hesaplanır (Liu ve Wechsler, 1998).

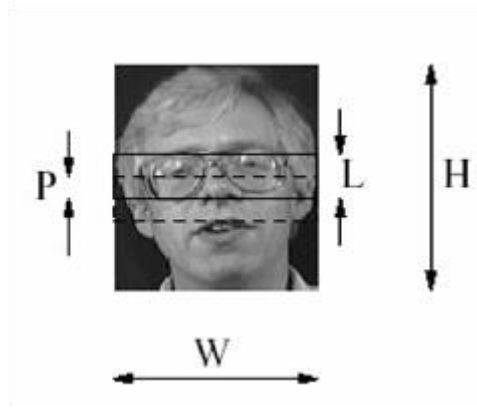
3.11.18 Gizli Markov Modelleri (HMM)

Gizli markov modelleri (HMM), bir sinyalin istatistiksel özelliklerini karakterize etmek için kullanılan bir dizi istatistiksel modeldir. HMM birbiriyle ilişkili iki süreçten oluşur. Bunlar, sınırlı sayıda durum, bir durum geçiş olasılık matrisi ve bir başlangıç durumu olasılık dağılımı olan bir altta, gözlenemeyen markov zinciri ve her durumla ilişkili bir olasılık yoğunluk fonksiyonu kümesi.

Yüz tanıma için 2 boyutlu ayrık kosinüs dönüşümü (DCT) özellik vektörlerinin çıkarılmasına dayanan HMM yaklaşımıdır. Modelde, özellik çıkarma için DCT sıkıştırma özelliğinden yararlanır. Bir görüntü, gözlem vektörü ile ilişkili bir alt-blokun bloklarına bölünür. HMM'de modelde sınırlı sayıda statüye sahip gözlemlenemeyen markov zinciri, gözlem sembolü olasılık matrisi B, durum geçiş olasılık matrisi A, başlangıç durum dağılımı π ve olasılık yoğunluk fonksiyonlarının (PDF) kümesi bulunmaktadır. Bir HMM $\lambda = (A, B, \pi)$ üçlüsü olarak tanımlanır. Önden insan yüzü görüntüleri için, önemli yüz bileşenleri saç, alın, göz, burun, ağız ve çene gibi yukarıdan aşağıya doğru sırayla görünür. Görüntü, görüntü düzleminde hafifçe dönmesine rağmen belirlenmeye devam eder. Yüz bölgelerinin her biri, 1B sürekli HMM'de bir duruma atanır. Geçiş olasılığı a_{ij} ve yüz model yapısı aşağıdaki şekilde gösterilmektedir (Nefian ve Hayes, 1998).



Şekil 3.18 Yüz tanıma görüntüsü için HMM (Ferdous ve Ahmed, 2008)



Şekil 3.20 Blok çıkarma formu (Ferdous ve Ahmed, 2008)

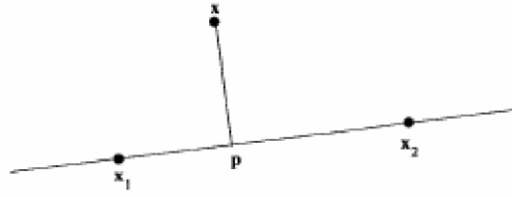
3.11.19 Yükseltme ve Topluluk Çözümler

Yükseltmenin ardındaki fikir, bir tip için sınıflandırıcıları genelleştirmek için belirli bir eğitim örnekleminin ağırlıklı bir versiyonu üzerinde zayıf bir öğreniciyi sırayla kullanmaktır. Herhangi bir sınıflandırıcı tek başına rastgele tahminden biraz daha iyi performans gösterse de oluşan topluluk doğruluğu yüksek yani güçlü bir sınıflandırıcı sağlar. Viola ve Jones, yüz tanıma araştırmasında önemli bir atılım olarak kabul edilen AdaBoost'u kullanarak ilk gerçek zamanlı yüz tanıma sistemini kurdular. Öte yandan tarafından AdaBoost yöntemlerini kullanarak yüz tanıma konusunda ilk yaklaşımlar Guo ve arkadaşları geliştirilmiştir (Lu ve ark., 2006).

3.11.20 En Yakın Özellik Hattı Yöntemi

Özyüzler yaklaşımında karşılaşılan sorunları aşmak için bütünsel eşleme yöntemidir. Özellik uzayında bir özellik noktası oluşturmak için, aynı sınıftan (resim) en az iki prototip özellik noktası bulunması gerektiği varsayılır. İki prototip özellikten geçen bir satır, iki özellik noktasını genelleştiren bir özellik çizgisi (FL) oluşturur. Özellik çizgisi, farklı yüz görüş açısı veya farklı ışık aydınlatması gibi çeşitli koşullar altında yakalanan iki prototipin yaklaşık bir görüntüsünü temsil eder. Bir girdi görüntüsü daha sonra, ilgili görüntünün özellik noktası ile prototip görüntülerinin FL'si arasındaki mesafeye göre, karşılık gelen bir sınıf ile tanımlanır. Yüz resmi, bir öz uzay olan, özellik alanında bir nokta olarak temsil edilir. $x_1 \times x_2$ olarak belirtilen aynı sınıftaki x_1 ile x_2 arasındaki satır, bu sınıfın özellik çizgisi olarak adlandırılır. Giriş görüntüsü x olan sorgu, bir FL üzerine K olarak yansıtılır. $92/5000$ FL mesafesi, x ve $x_1 \times x_2$ d (x ,

$x_1x_2) = |x - p|$ olarak tanımlanmıştır, burada $| \cdot |$ bazı normlardır. Yansıtma noktası, parametre ile bir vektör çizgi denklemi kurularak bulunur. μ işaretine bağlı olarak, p 'nin konumu x_1 'in solunda, x_2 'nin sağında veya x_1 ile x_2 arasında olabilir. Parametrenin değeri ne kadar büyük olursa, x_1 veya x_2 'nin konumundan gelen p değeri de büyük olur. Giriş görüntüsünün sınıflandırması şu şekilde yapılır: c_i x ve c_j x özellik alanında iki ayrı prototip noktası olsun (Li ve Lu, 1999).



Şekil 3.19 x_1 ve x_2 prototip özellik noktalarının genellenmesi (Turk ve Pentland, 1991)

Görüntü noktası x ve $i \neq j$ olduğu her FL c_i x c_j x arasındaki mesafe her c sınıfı için hesaplanır. Bu sayede N mesafelerin sayısı üretilir ve bu N mesafe her biri sınıf C^* ile ilişki oranıyla artan şekilde sıralanır. Birinci sıra, en iyi eşleştirilen C^* sınıfını ve sınıfın en iyi eşleşen i^* ve j^* prototipden oluşan NFL sınıflandırmasını verir (Li ve Lu, 1999).

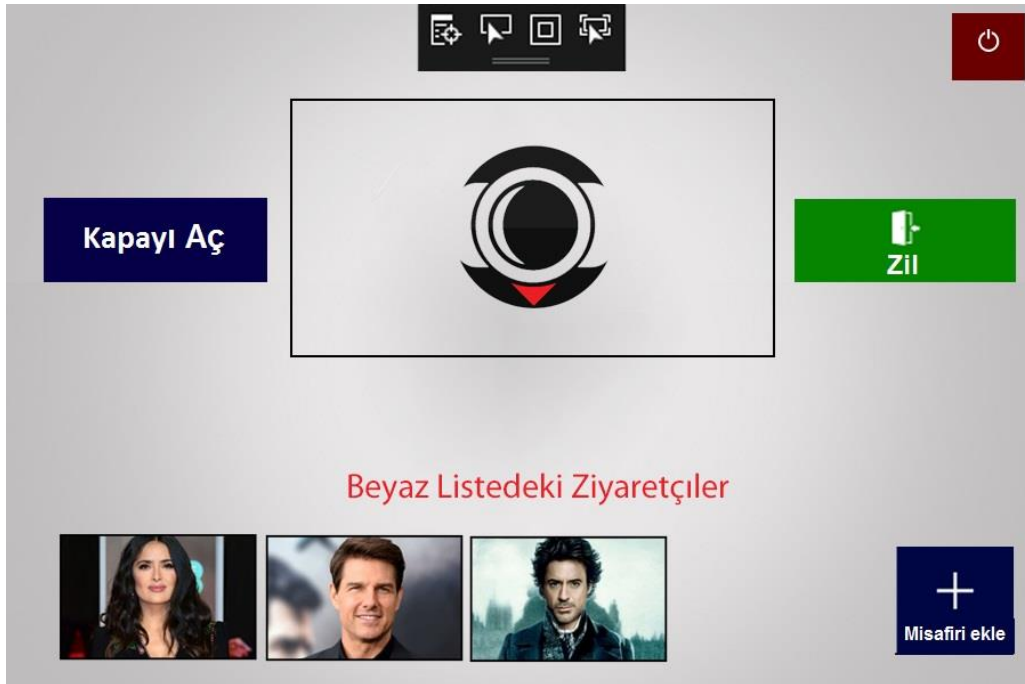
3.12 Tasarım ve Uygulama

Erişim kontrol sistemini tasarlamak ve modellemek için önemli olan mikro kontrolörler ve arayüzler hakkında ileri düzey bilgi ve bunun uygulamaya konmasıdır. Bu sistemi kullanmak için masaüstü uygulaması geliştirmede çeşitli sürücülerle ve arayüzlerle uyumlu bir şekilde çalıştığı için Raspberry Pi bilgisayar sistemi kullanılmıştır. Yukarıda belirtilen işlevleri sağlamak için Yüksek kaliteli bir kamera işlemci üzerinden ekrana bağlanır. Kamera fotoğrafı alır ve Raspberry Pi'ye gönderir (Ferdous ve Ahmed, 2008).

3.13 Sistemin Süreçleri:

Oluşturulan sistemde farklı işlevler vardır ve bunları bir sonraki bölümde ayrıntılarıyla açıklayacağız. Tüm sistemin dört ana işlevi vardır.

- 1) Önce kapıya gelen kişi kapı ziline basar. Kapıya entegre edilmiş kamera modülü kişinin görüntüsünü yakalar ve kişinin yüzünü ayırır. Daha sonra bu yüz görüntüsünü bağlı olan Microsoft Yüz Tanıma API'sine gönderir. Microsoft Azure bulut kurulumu, daha önce buluttaki veritabanından depolanmış olan yüz fotoğrafları ile sistemden gelen yüz fotoğrafını karşılaştırır. Eğer veritabanındaki herhangi bir kullanıcı ile eşleşirse, işlemci “hoş geldiniz (kişinin adı)” sesini çalar ve röle modülünü kontrol eden kapıyı açar. Sistemin bu işlevi şekil 3.23'te gösterilmiştir.



Şekil 3.20 Sistemin kullanıcı arayüzü

- 2) Projenin ikinci işlevi kullanıcıları veritabanına eklemektir. Yeni bir kullanıcı eklemek için “Misafiri ekle” butonuna basılır. Daha sonra sisteme bağlı olan kamera modülü açılır ve fotoğraf çeker. Kameranın önünde bulunan kişinin yüzünün görüntüsünü yakaladığı zaman sistem kişinin ismini sorar. Daha sonra yüz görüntüsünü ve isim bilgileri veritabanına aktarılır.

- 3) Üçüncü işlevi, kapıya gelen kişi sistem tarafından tanınmadığı zaman kullanıcıya e-posta gönderir. Böylece tanınmayan kişinin kapıdan geçmesi engellenir.
- 4) Bu projede son ana işlev, uygulama aracılığıyla kullanıcıya canlı video akışı sağlanmasıdır. Böylece sistem, kullanıcının gelen kişinin kapıdan geçip geçmemesine yetki verir. Kullanıcı “Kapıyı Aç” butonuna tıkladığı zaman sistem gelen kişinin yüz görüntüsünün veritabanında eşleşme yapmasına gerek kalmadan geçmesine izin verir.

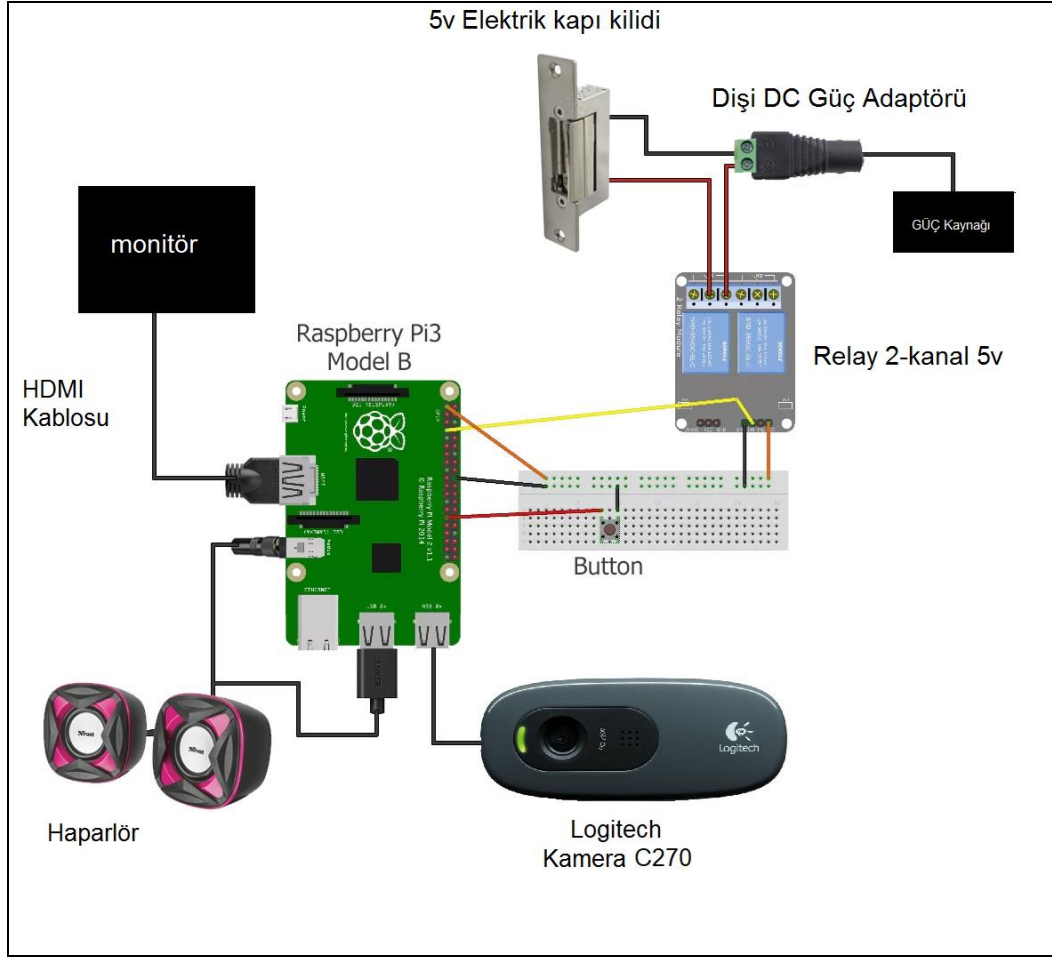
3.14 Metot

3.14.1 Donanım

Bu projede sistemin yönetimini Raspberry Pi 3 ile yapılmıştır. Raspberry Pi 3’ün kurulumu Windows 10 IoT ile yapılmıştır. Daha sonra Raspberry Pi 3’e bir ekran bağlanmıştır. Kamera, giriş portu üzerinden Raspberry Pi’ye bağlanır ve onun üzerinden arayüze veri aktarır.

Eve gelen kişinin görüntüsünü yakalayan sistem, uygulama aracılığıyla canlı video akış sürecini gerçekleştirir. Gelen kişinin görüntüsünü bulut üzerindeki veritabanında eşleşme yapan sistem bu süre boyunca kapının kilitletlenmesini sağlar. Raspberry Pi video akışını sürekli denetler ve gelen kişinin anlık video akışı gerektiğinde kullanıcıya ulaştırılırken video ayrıca bulutta depolanır. Gelen kişi sistem tarafından tanınmazsa kullanıcıya bir mesaj gönderir ve zili çalar.

Donanım bağlantısı aşağıdaki şekil 3.23’de gösterilmiştir. Kablosuz klavye / Mouse, kamera ve hoparlörleri USB portu üzerinden ve ekran HDMI portu üzerinden Raspberry Pi’ye bağlanmıştır. Kapı zili için GPIO 5 ve röle (kapı kilidi) için GPIO 4 kullanılmıştır. 5V röle kapı kilidine bağlanmıştır. Her iki röle ve kapı kilidi, güç kaynağına (12V, 3A) Dışı DC güç adaptörü ile sağlanmıştır.



Şekil 3.21 Donanım bağlantıları

3.14.2 Yazılım

Yüz tanıma, resimlerden algılanan yüzlere göre görüntülerin otomatik olarak düzenlenmesini sağlayan, resimlerinizi akıllı bir şekilde kırpmanızı, konumlandırmanızı, yeniden boyutlandırmanızı ve dönüştürmenizi kolaylaştıran bir özelliktir.

Makine Öğrenimi, özellikle son yıllarda veri işlemede oldukça popüler hale gelmiştir. Bununla bağlantılı olarak, uzman olmayan kişilere, Makine Öğrenmesi ile ilgili çeşitli araçlar ve API'ler (Uygulama Programı Arayüzü) sunulmuştur. Microsoft, Azure Machine Learning paketini geliştirmiştir ve Cortana analiz paketi ile geliştiricilerin Makine Öğrenimi hakkında nispeten kolay şekilde çalışmasına izin veren çeşitli API'ler hazırlamıştır. Ayrıca yüksek hassasiyetli yüz algılama için bir yüz API'si sağlayan Cognitive Services ile işbirliği yaparak bulut tabanlı algoritmalarla daha hızlı veri analizi yapılmasını sağlamıştır. Bunun için Yüz API teknolojisi, bir görüntüdeki

insan yüzlerini algılamaktan daha fazlasını yapabilen Gelişmiş Yüz Tanıma Algılama eklentileri ile bir görüntüdeki yüz hakkında anlamlı ve gelişmiş veriler de çıkarılmasını sağlamaktadır. Böylece görüntü kategorizasyonunuz üzerinde daha da fazla kontrol sağlanmasını ve bu ayrıntıları, algılanan yüz özelliklerine göre görüntüleri kırpma, konumlandırmak, döndürmek ve üst üste bindirmek için otomatik olarak kullanılmasına yol açmıştır.

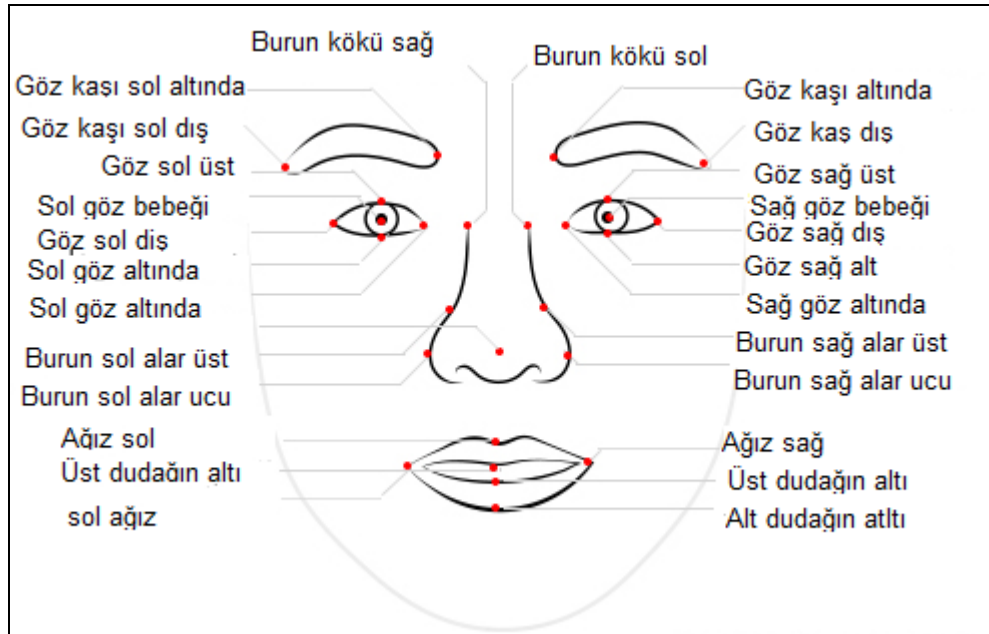
3.14.3 Yüz Tanıma Uygulamalarının Çalışması

Oluşturulan uygulamada, anlık görüntü alan web kamerasından gelen bilgilerden yüzler algılanır. Bir düğmeye basıldığında, kameradaki o anki görüntünün bir fotoğrafı çekilir ve daha sonra, API aracılığıyla cognitive Servise yüklenir. Servis, bir yüz nesnesini döndürür. Göz, burun, kaş ve dudakların pozisyonları, yaş, cinsiyet gibi özelliklerle birlikte iki düzineden fazla “yüz yer işareti” içeren bilgi veri tabanına kaydedilir. API'ye iki ayrı yüz nesnesi verilirse, iki yüz arasındaki benzerliği hesaplar ve bu iki yüzün aynı kişiye ait olup olmadığını belirler. Daha sonra iki resim arasındaki benzerliği ölçer ve 0–1 aralığında bir güven değeri geri döndürür. Bu değer 0,5'in üzerinde ise iki yüz resmi aynı kişiye ait olarak işaretlenir. Fakat Bu değer 0,5'in altındaki değerler için iki yüz resmi arasında bir eşleşme olmadığını kabul eder.. Uygulamada web kamerası görüntü akışının sabit kare görüntüsü daha sonra bir dizi referans resimle karşılaştırılır. Daha sonra yakalanan görüntüde tespit edilen yüz çerçevesi veritabanından eşleşmiş olduğu yüz ve güven değeri gösterilir (Microsoft.Azure, 2018).

Bir görüntü yükleme ve yüz algılama: Yüz algılama işleminin gerçekleştirilmesinin en temel yolu, bir görüntünün doğrudan yüklenmesidir. Görüntüyü yüklemek için uygulama içe bir “POST” isteği gönderilir. Bu gönderi sekizli-akış içerik tipinde olmalıdır. Görüntünün boyutu 4 MB'ı geçmemelidir. İstemci kütüphanesi kullanarak, karşıya yükleme yoluyla yüz algılama bir Aktarım nesnesine yüklenerek yapılır. DetectAsync yöntemi eşzamanlı iken, FaceServiceClient yöntemi eşzamansızdır. “await” yöntemini kullanmak için çağrı yöntemi de eşzamansız olarak işaretlenmelidir. FaceRectangle özelliği, tanımlanan yüz ile dönderilir ve yüzdeki temel bölgelerin piksel cinsinden konumlarını içerir. Genellikle bu dikdörtgen gözleri, kaşları, burnu ve ağzın konumlarını içerir fakat başın üst kısmı, kulakları ve çeneyi içermez. Tam bir kafa veya orta çekim portresi (yüz haricinde vücudun yarısında bulunduğu

fotoğraf), yüzün alanı bazı uygulamalar için çok küçük olabileceğinden, dikdörtgen yüz çerçevesinin alanını genişletilmesi gerekir (Microsoft.Azure, 2018).

Yüz belirli noktalarını algılama ve kullanma: Yüz belirli noktaları, yüzdeki ayrıntılı noktalarından oluşan bir gruptur; temel olarak gözbebeği, kantas (göz köşesi) veya burun gibi yüz bileşenlerinin noktalarıdır. Yüz belirli noktaları, yüz algılama sırasında analiz edilebilen isteğe bağlı niteliklerdir. Yüz Algılama'yı çalıştırırken returnFaceLandmarks sorgu parametresine "true" değerini bir Boolean değeri olarak atanabilir veya algılama sonuçlarına yüz belirli noktalarının dâhil olması için FaceServiceClient sınıfı DetectAsync metoduna returnFaceLandmarks'ın seçimli parametresini kullanılır. Varsayılan olarak tanımlı 27 belirli nokta vardır. Aşağıdaki şekilde bütün noktalar açıklanmıştır (Microsoft.Azure, 2018) :



Şekil 3.22 Belirli noktalar (Microsoft.Azure, 2018)

Yüz dikdörtgen çerçevesinde olduğu gibi geri döndürülen noktalar piksel birimidir. Bu sayede görüntüdeki belirli noktaların tespit edilmesi kolaylaşır. Yüz belirli noktaları da yüzün yönünü doğru bir şekilde hesaplamak için kullanılır. Örneğin, ağız ortasından gözlerin ortasına doğru bir vektör olarak yüzün yönü tanımlanır. Yüzün bulunduğu yön bilindiğinde, dikdörtgen yüz çerçevesini yüzle hizalamak için döndürülür. Yüz belirli noktalarını kullanmanın daha fazla ayrıntı ve fayda sağlayacağı açıktır.

Diğer yüz özelliklerini kullanma: Yüz belirli noktalarının yanı sıra, Yüz - Algılama API'si ayrıca bir yüzdeki yaş, cinsiyet, gülümseme yoğunluğu, sakal, 3B kafa pozunu gibi öznitelikleri de analiz eder (Microsoft.Azure, 2018)

3.14.5 Microsoft Bilişsel Hizmetler

Microsoft bilişsel hizmetler (MS cognitive services), yüzleri, görüntüleri ve duyguları hakkında bilgi edinerek içerikleri otomatik olarak yönetmeye ve kişiselleştirilmiş uygulamalar oluşturmaya yardımcı olan (son teknoloji) bir görüntü işleme algoritmasıdır (Hariharan, 2017). Bu API, Yüz Algılama, Yüz Doğrulama, Benzer Yüz Arama, Yüz Gruplaması ve Yüz Tanıma gibi çeşitli işlemlere sahiptir. Görme, konuşma, dil, bilgi ve arama alanlarında makine öğrenimi ve yapay zekâ yetenekleri sağlar. Proje için Face API'sini kullanılmıştır. Bir arkadaş listesi oluşturulmuştur ve listedeki kişilerin resimleri Microsoft Azure bulutundaki bir listeye eklenmiştir. Yüzleri görsel benzerliklerine göre gruplara ayırmak için Face API'nin Yüz Gruplama özelliği kullanılmıştır. Kapıya gelen ziyaretçinin fotoğrafı çekilir. Resimler çekildikten sonra Azure'a yüklenir. Bir Mantık Uygulaması, bu görüntüleri bir Blob Depolamasına yüklenecek olan Azure'dan tüketir. Bilişsel Hizmetler Yüz API'sine iletilen URI'yi (Face API'nin alabileceği bir genel URI / URL sağlayarak) alır. Face API, SQL Azure'da depolanacak bir dizi özniteliği döndürecektir. İşlemcide, yüzün insan yüzü olup olmadığı incelenir. Resimde bir insan yüzü varsa, eşleşme olup olmadığını saptamak için Microsoft Azure bulutuna gönderilir. Oluşturulan veritabanından yüzün eşleşmelerini aranır. Daha sonra bir yüz eşleşmesi olasılığındaki güven düzeyi % 0 ile % 100 arasında bir sayı olarak döndürülür. Kod C # dilinde yazılmıştır ve Raspberry Pi ile yapılan projeyi kodlamak, derlemek ve hata ayıklamak için Visual Studio 2017 kullanılmıştır (social.Microsoft, 2016).

Kişileri tanımlamak için programda yapılacak işlemler:

- Görüntülerdeki Yüzleri Algılama
- Bir Kişi Grubu Oluşturma

olacaktır. Ön şartlar

- Microsoft Face API Windows SDK
- Onaylama Anahtarları

bulunmasıdır (JetBrains, 2017).

C # kullanarak Yüz API ile görüntülerde Yüzleri Algılama:

Bir görüntüdeki yüzleri algılamak ve aşağıdakileri içeren yüz özelliklerini döndürmek için Yüz Algılama yöntemini kullanın:

- Yüz Kimliği: Çeşitli Face API senaryolarında kullanılan benzersiz kimlik.
- Yüz Dikdörtgeni: Görüntüdeki yüzün yerini belirten sol üst noktanın koordinatı, genişlik ve yükseklik bilgilerinin bulunduğu değişken
- Belirli noktalar: Yüz bileşenlerinin önemli konumlarına işaret eden 27 noktadan oluşan bir dizi yer işareti.

Yaş, cinsiyet, gülümseme yoğunluğu, baş pozunu ve sakal ya da bıyık bilgilerinin kaydedildiği diğer yüz özellikleri

Yüz Tespiti Programı:

Ek 1 de vermiştir

Yüz algılama yanıtı

JSON' da gönderilen yanıt şekil 3.25'te gösterilmiştir:



Şekil 3.23 Gönderilen resimden json olarak geri dönen sonuç

Bu resmin üzerindeki sonuç bilgilerinin kodu Ek 2’de verilmiştir.

Programda “Kişi Grubu” oluşturma:

Sistemin sınıflama yapması için program içinde “Kişi Grubu” oluşturulur. Kişi grubu oluşturma kodları ek 3’te verilmiştir.

3.14.6 Kapı Kilidinin Açılması:

Gelen kişinin doğrulanmasından sonra kişinin geçmesi için kapının açılması gerekir. Kapıyı açmak için de servo motor kullanılmıştır. Bu servo motorun kontrolü, servo motor arabirimi üzerinden Raspberry Pi ile sağlanır. Belirli bir süre sonra kapı otomatik olarak kilitlenir. Böylece sistem güvenliği sağlamaya devam eder.

3.15 Karşılama

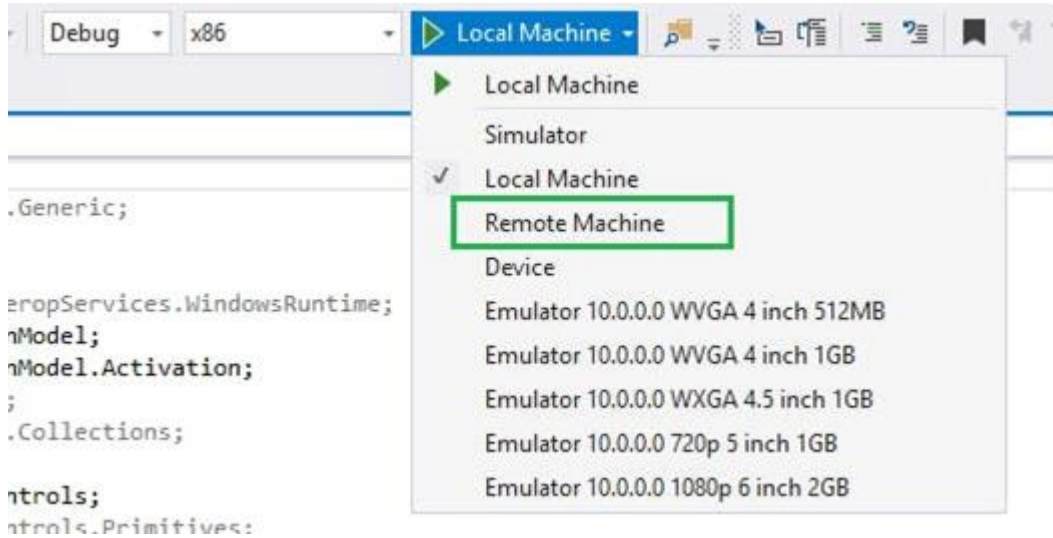
Akıllı sistem hoparlör ile sistemin, gelen kişiyi tanımlanması durumunda ziyaretçinin sesli mesaj alması ve ziyaretçinin adıyla selamlama yapması sağlanır. Sentezlenen konuşma mesajlarını oynatmak için kullanılan medya elemanı yüklendiğinde tetiklenir, SpeechHelper’ı başlatır ve kullanıcıyı selamlar.

3.16 Sistemin Değiştirilmesi

Raspberry Pi bir uygulama çalıştırılmış ve etkileşimli bir kullanıcı arayüzüne sahiptir. Böylece uygulama donanımlara erişmek ve istenen işlevleri yerine getirmek için API’leri çağırmasına izin verilmiş ve sonuç ekranda gösterilmiştir. Uygulama, Microsoft Visual Studio’da çalıştırılmış ve API (ler) bunun üzerinden çağırılmıştır.

- 1) Raspberry Pi bir güç kaynağına bağlanır ve OS (Windows 10 IOT) üzerine takılı mikro SD kart üzerine kurulur.
- 2) Raspberry Pi ARM kontrol ünitesini çevre birimleriyle (ekran, klavye, web kamerası, hoparlörler ve fare) bağlantısı yapılır.
- 3) Başka bir ana cihazda Windows 10 IOT Dashboard kurulur ve RPI işlemcisini onunla yapılandırılır (kablosuz bağlantı kurulabilir).
- 4) Dağıtılan uygulama, Microsoft Azure üzerinden API ile arayüzlenmelidir

- 5) Ebeveyn cihazında Microsoft Visual Studio'yu indirilir ve XAML ve C # 'nun yüksek düzey dilinde oluşturulan kodu aşağıdaki gibi Raspberry Pi'ye yüklenir.
- 6) Visual Studio'nun üst menüsünde, Raspberry Pi kullanıyorsanız Debug ve ARM seçilir. Bir MBM kullanılıyorsa, Hata Ayıklama ve x86 seçilir.



Şekil 3.24 MBM yapılandırması

Raspberry Pi kullanıcıları için x86 yerine ARM seçilir. "Uzak Bağlantılar" iletişim kutusunda Uzak Makineye tıklanır, Uzak Makine IP adresinizi girilir ve Kimlik Doğrulama Modu için "Evrensel (Şifrelenmemiş Protokol)" kullanılması gerekir.

IP adresini bulmak için WindowsIoTCore'i kullanılmalıdır.



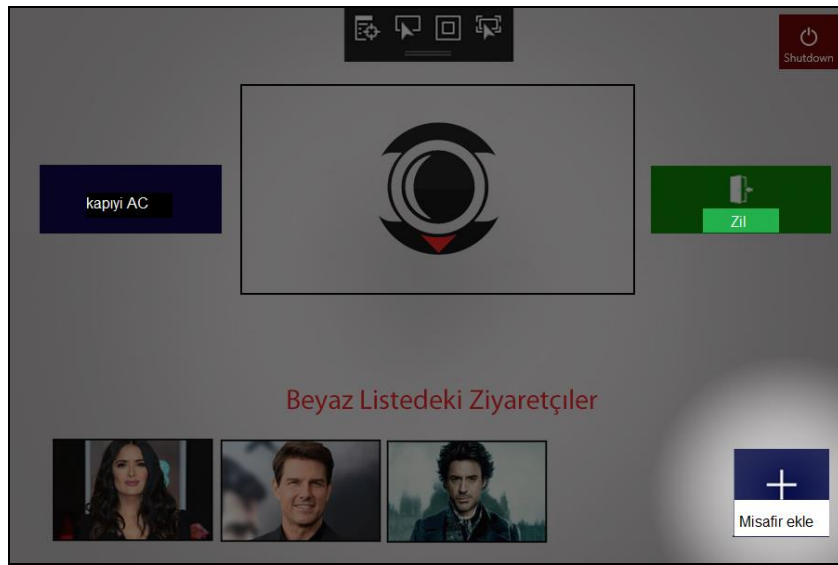
Şekil 3.25 Uzaktan bağlantı için cihazın ayarları

Bundan sonra kod çalıştırılabilir hale gelmiştir. Yeşil "play" oku "remote machine" düğmesine basılır.

Yüz tanıma, yapay sinir ağları ilkesi üzerinde çalıştığı için, sonucun hesaplanmasında yüksek hesaplama donanımının kullanılmasını gerektirir. Bu donanımlar ise Azure'daki uygulama dağıtımının sağladığı GPU desteğiyle yapılabilir.

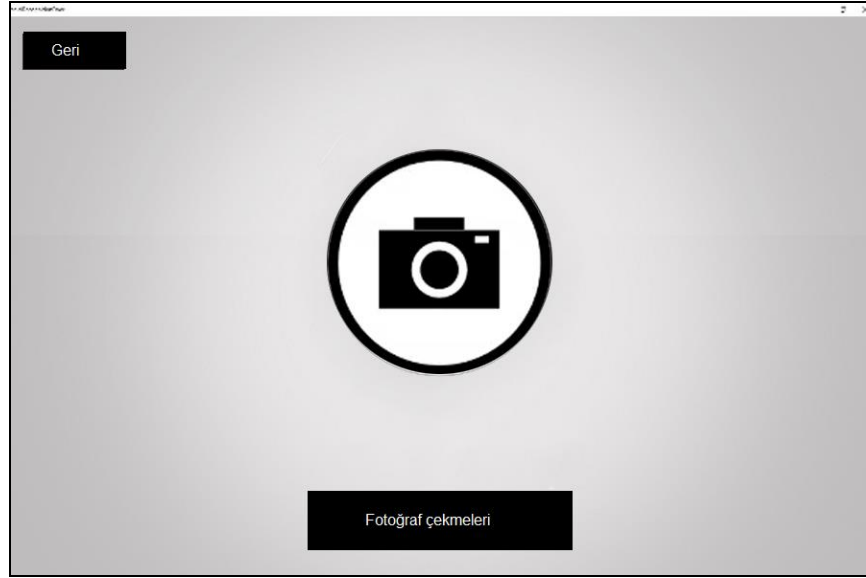
Aşağıdaki ekran görüntüleri Raspberry Pi gibi davranacak şekilde kurulmuş bir PC'de alınmıştır.

Proje ilk çalıştırıldığında ekrana ilk görüntü gelir. Şekil 3.28'da gösterilmiştir. Bir MBM'de, Software Setup'ın 6. adımında DisableLiveCameraFeed değişkenini false olarak ayarlanır ve kamera simgesini görmek yerine, kameranın gelen anlık görüntüler ekranda görünmüş olur.



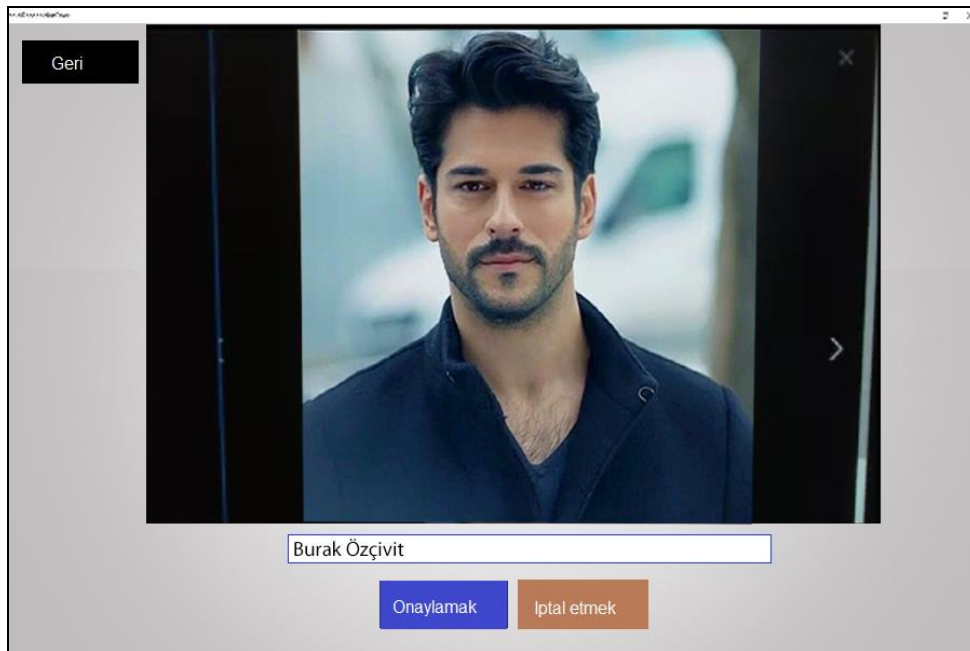
Şekil 3.26 Programın ana ekranı ve kişi eklemek için misafir ekle butonunun belirtilmesi

Yukarıdaki Şekil 3.28'de gösterildiği gibia It uygulama çubuğunda üç buton bulunur. Bunların ilki "Misafir Ekle" butonudur. Bu, "beyaz liste"ye yeni bir kullanıcı eklemek için kullanılır. Beyaz listedeki herhangi bir kullanıcı tanıdık olarak işaretlenmiştir ve bu kişilerin yüzlerinden kapının kilidi açılır. Şimdi "Misafir ekle" butonuna basılınca Şekil 3.29 gösterildiği gibi, fotoğraf alma ekranı görünür.



Şekil 3.27 Fotoğraf çekme ekranı

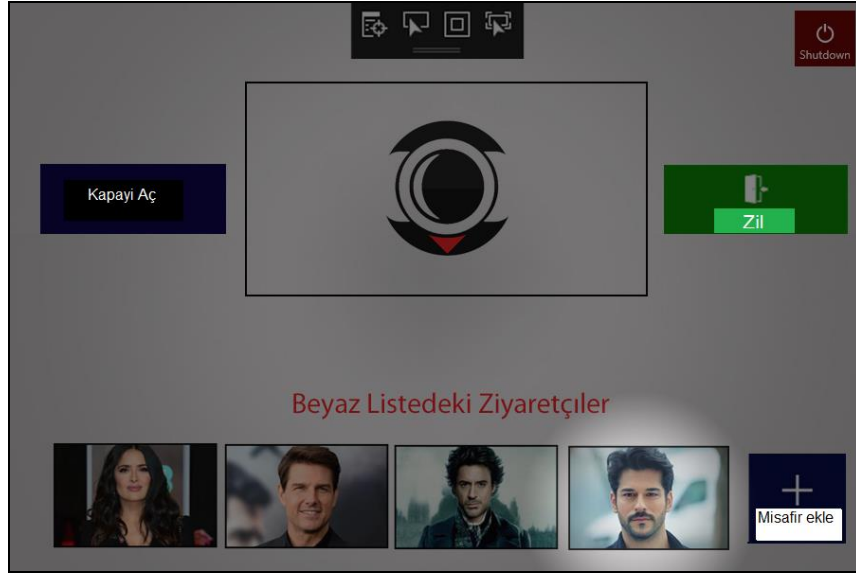
Web kamerasının önüne listeye eklenecek kişinin geçmesi gerekir. “Fotoğraf çekmeleri” butonuna basılınca kişinin fotoğrafı çekilmiş olur. Şekil 3.30’da gösterildiği gibi, bu ekranda yeni yakalanan fotoğraf görünür.



Şekil 3.30 Fotoğraf çekme sonunda gelen ekran

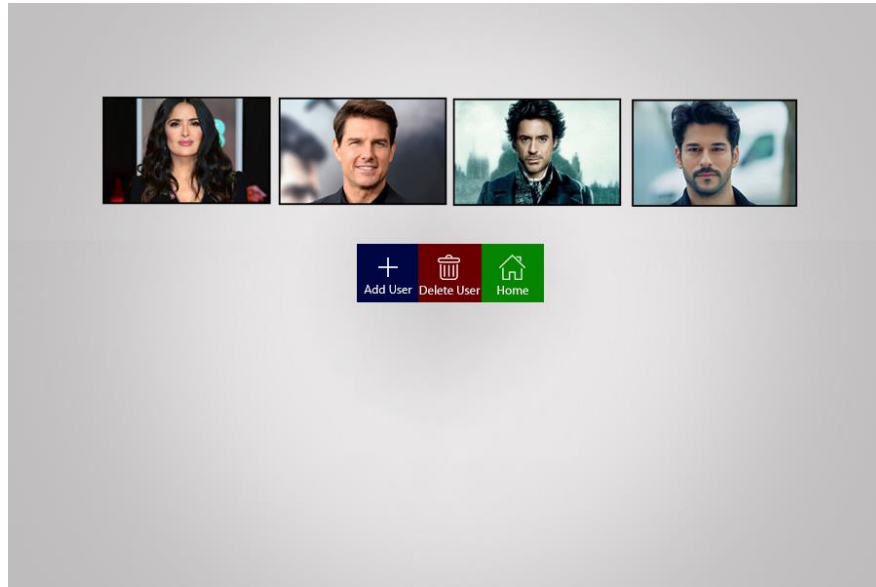
Fotoğrafın kabul edilmesi için fotoğraftaki kişinin adını girilir ve “Onaylamak” butonuna basılır. Böylece kişi beyaz Listeye kaydedilmiş olur. Eğer yeni bir fotoğraf çekilmesi istenirse “İptal etmek” butonuna basılır. Böylece bir önceki ekrana dönülür.

Onayla tuşuna bastıktan sonra ana sayfaya geri dönülür ve Beyaz Listeye Eklenmiş Ziyaretçiler bölümünde yeni kullanıcı görülür. Bu şekil 3.31 gösterilmiştir.



Şekil 3.28 Yeni kişi eklendikten sonra ana sayfanın görünüşü

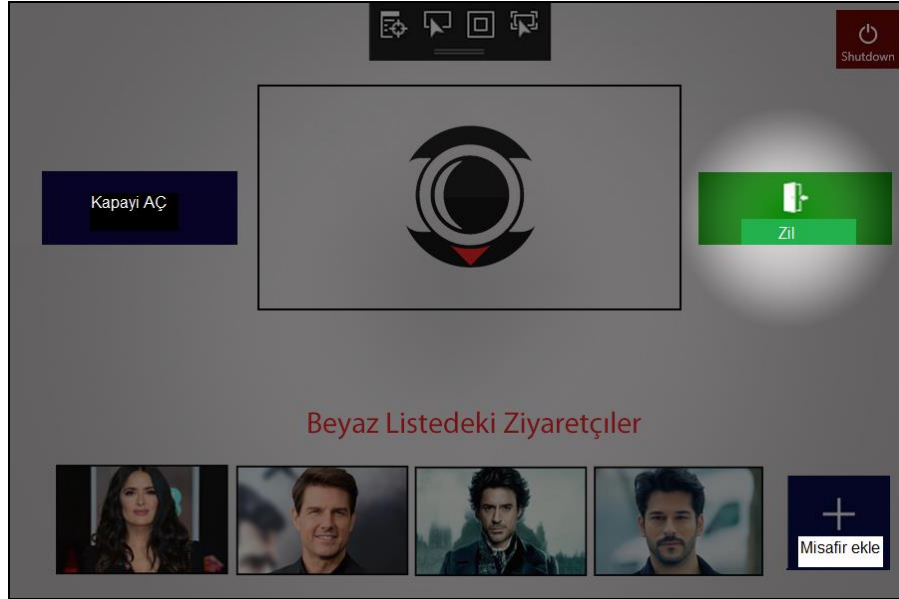
Şekil 3.31’de olduğu gibi, kullanıcının profil sayfasına gitmek için sayfanın sol alt köşesindeki ilgili resme tıklanır. Şekilde 3.32 kişinin profil sayfası ekrana gelir.



Şekil 3.29 Veritabanındaki kişilerin görüntüsü

Bu sayfada, daha fazla kişi eklemek için “Misafir ekle” simgesi kullanılır. Bu, sistemin yüz tanıma doğruluğunu artıracaktır. Ayrıca bu kullanıcıyı silmek için çöp kutusu simgesine tıklanır. İşlem yapılmayacaksa ev simgesine tıklanarak ana ekrana dönülür.

Kapının açılması için kayıtlı kullanıcı web kamerasının önüne konumlanır ve şekil 3.33'de gösterildiği gibi sanal kapı simgesine tıklanır. Ayrıca kapının açılması için kablolu hale getirilmiş fiziksel "Zil" düğmesine basılır. Kapının açıldığını bildiren sesli geri bildirim duyulur. Eğer kayıt edilmemiş bir kullanıcı kapının önüne geldiğinde kapı zili düğmesine basılsa bile kapı açılmaz. Kapının bir yabancı algıladığını ve kilidin açılmayacağını bildiren sesli geri bildirim duyulur.



Şekil 3.30 Zil butonu

4. SONUÇLAR VE TARTIŞMALAR

Bu projenin deney sonuçları tamamlanmış ve hem donanım hem de yazılım açıklanmıştır. Bu yüz tanıma otomatik erişim kontrol sistemi, projenin amacına ulaşan istenen sonucu sunmak için GUI sistemi ile entegre olmuştur. Bir önceki bölümde, bu sistem UWP (evrensel windows uygulaması) tarafından oluşturulmuştur. Sistem yeni kullanıcı oluşturma, kullanıcıyı düzenleme, kullanıcıyı silme, zili çalma ve kapıyı açma bölümlerinden oluşmaktadır. Bu sistemin iki ana amacı vardır; yüz tanıma ile kapı açma kontrolü sağlamak ve veritabanında kayıtlı olmayan ziyaretçilerin resmini ve geldiği zamanı kullanıcıya mesaj olarak göndermektir.

4.1 Donanım

Sistemin bilgisayarı olan Raspberry Pi'ye kablosuz klavye / Mouse, kamera ve hoparlörleri USB portuyla bağlanmıştır. Ekran ise HDMI kablosu ile bağlanmıştır. Kapı düğmesi için Raspberry Pi üzerindeki GPIO 5 ve kapı kilidi rölesi için GPIO 4 kullanılmıştır. 5V kapının kilidi için 5V ile röleye bağlanmıştır ve hem röle ve hem de kapı kilidini 12V'luk Güç Kaynağı ile beslemesi sağlanmıştır. Kapı kilininin beslemesi dışı Dc güç adaptörü üzerinden verilmiştir.

4.2 Yazılım

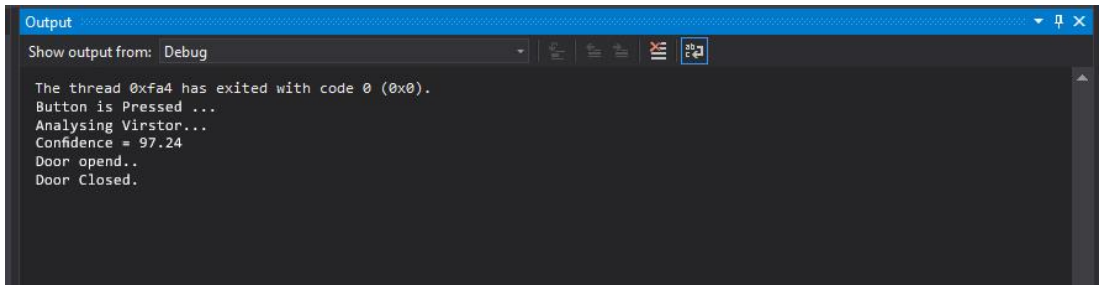
Sistemin kullanıcı arayüzünü Microsoft Visual studio C#'ın UWP (windows universal application) ile oluşturur. Sistem ile veritabanına yeni ziyaretçiler eklenir. Ayrıca, API kullanarak sistem resim göndermek ve sonuç olarak doğruluk bilgisini almak için veritabanına bağlanır. Önceki bölümdeki sistemin her bir butonu açıklanmıştır.

Bu sistem tanınan mod ve tanınmayan mod olmak üzere iki ana fonksiyondan oluşur. Bu modlar da kapıya gelen kişinin resminin veritabanında bulunup bulunmadığına bağlı olarak değişir.

4.2.1 Tanıma Modu

Kişinin yüzünü veritabanında bulunduğu zaman, yüzü kapıyı geçmek için kimlik olarak kullanılır. Düğmeye basıldığında, kamera ziyaretçinin yüzünü yakalar ve tanıma

işlemi için MS Azure'ye gönderir. Sonuçlar makine öğrenme testinin bir oranı olan doğruluk olarak geri döner. Oran % 50'ye eşit veya daha büyükse, ziyaretçinin tanındığı ve kapının birkaç saniye süreyle açılacağı ve sonrasında tekrar kapanacağı anlamına gelir. Eğer bu oran %50'den daha düşük olursa, bu, ziyaretçinin tanınmamış olduğu anlamına gelir. Sistem Kullanıcıya bir uyarı mesajı gönderir ve bu mesajda tanımlanmayan kişinin geldiği saat ekranda yer alır. Şekil 4.1'de tanınan bir kişinin sonucu ve Şekil 4.2'de tanınmaya bir kişinin sonucu gösterilmiştir. Kişi tanınmadığında kullanıcıya gönderilen e-posta Şekil 4.3'te gösterilmiştir.

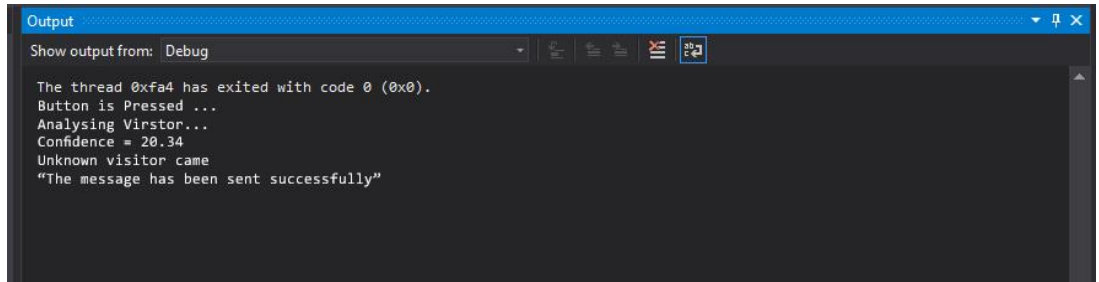


```

Output
Show output from: Debug

The thread 0xfa4 has exited with code 0 (0x0).
Button is Pressed ...
Analysing Virstor...
Confidence = 97.24
Door opend..
Door Closed.
  
```

Şekil 4.1 Bilinen ziyaretçiler için basılan zilin çıkışı göstermektedir.

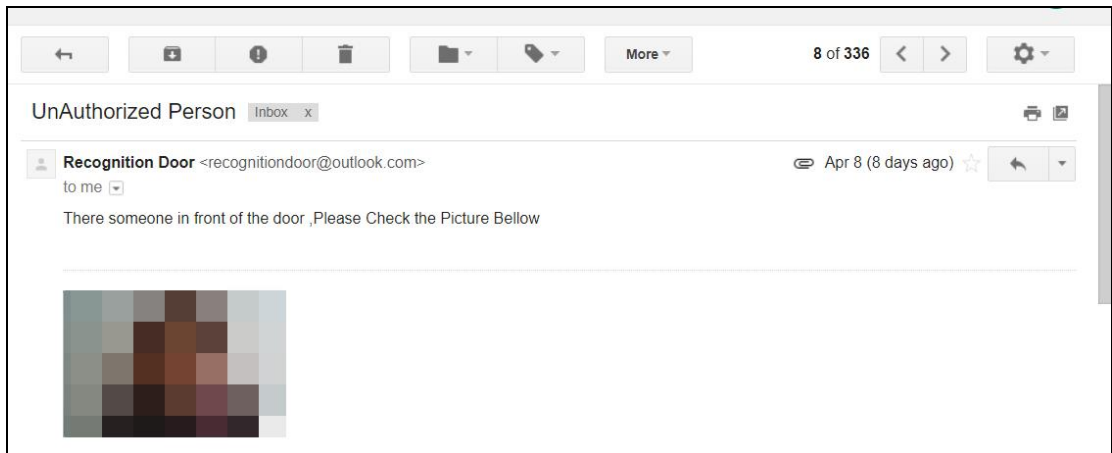


```

Output
Show output from: Debug

The thread 0xfa4 has exited with code 0 (0x0).
Button is Pressed ...
Analysing Virstor...
Confidence = 20.34
Unknown visitor came
"The message has been sent successfully"
  
```

Şekil 4.2 Bilinmeyen ziyaretçiler için basılan zil çıkışı göstermektedir



Şekil 4.3 Kullanıcıya giden uyarı mesajı

Mesaj geldikten sonra, "açık kapı" düğmesine basarak veya "misafir ekle" butonu ile yüzünü veritabanına eklenerek ziyaretçinin girmesine uzaktan izin verilir.

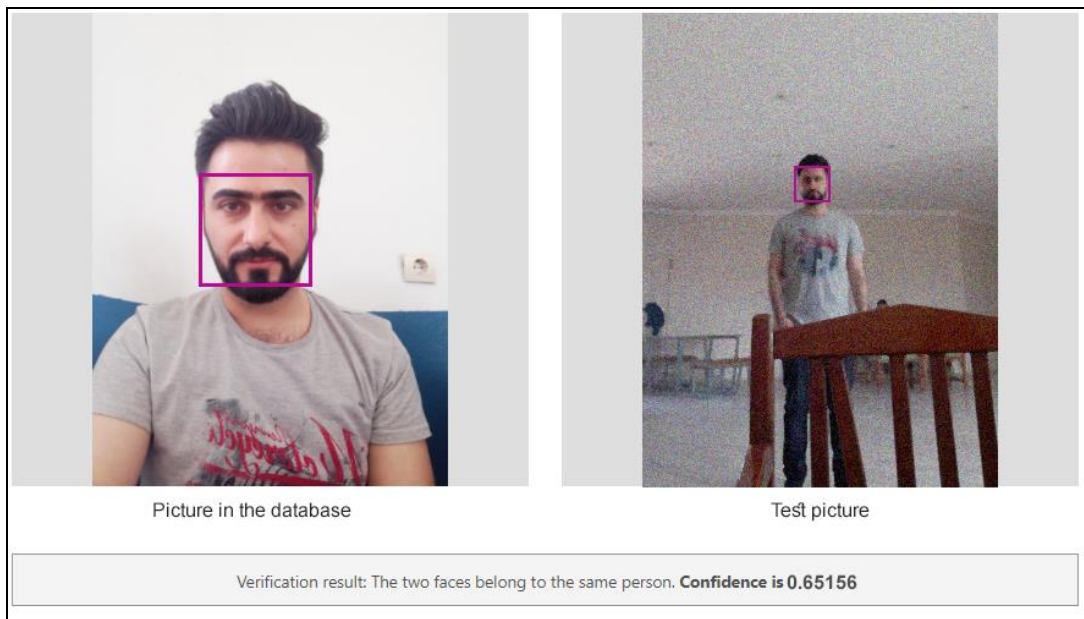
4.3 Deneyler ve Sonuç

Bu kısım, bir yüz tanıma sisteminin yazılım programlaması ve donanım simülasyonu açıklanmıştır. Bu tür bir sistemin tanıma doğruluğu ve donanım uygulama fizibilitesi incelenecektir. Logitech kamera C170 (5 megapiksel) ile doğruluğu kontrol edilmiştir. Kişinin fotoğrafı çeşitli uzaklıklarda ve değişik yüzün açıları ile çekilmiştir. Sonuçlar aşağıdaki tablo 4.1'de verilmiştir.

Çizelge 4.1 Deney 1.Yüz tanıma doğruluğu

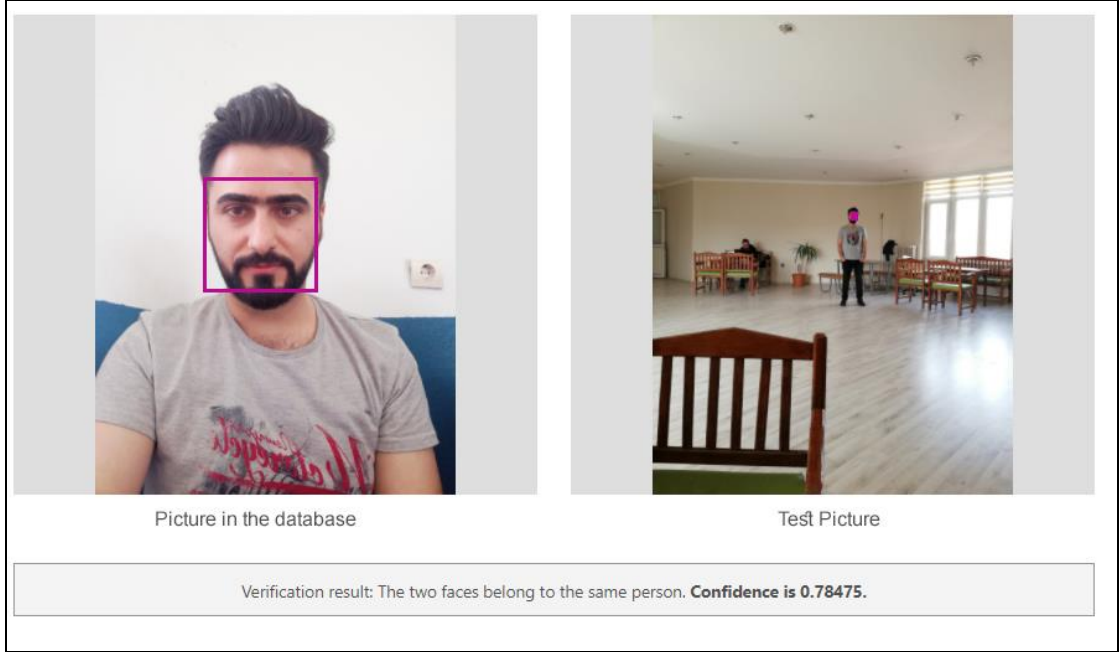
Mesafe/Açı	0°	15°	60°	75°	90°
0.5 m	98	85	83	73.38	60.72
1 m	97	84.45	82.59	71.6	69.45
1.5 m	95	83.4	82.34	70.2	68.30
2 m	67.15	67.3	66.5	65.90	65.04

Logitech kamera C170 (5 megapiksel) ile 3 metreye kadar doğru algılama ve tanıma mesafesine ulaşılmıştır ve aşağıdaki resim 4.4'te gösterildiği gibi güven yüzde 65.156 bulunmuştur.



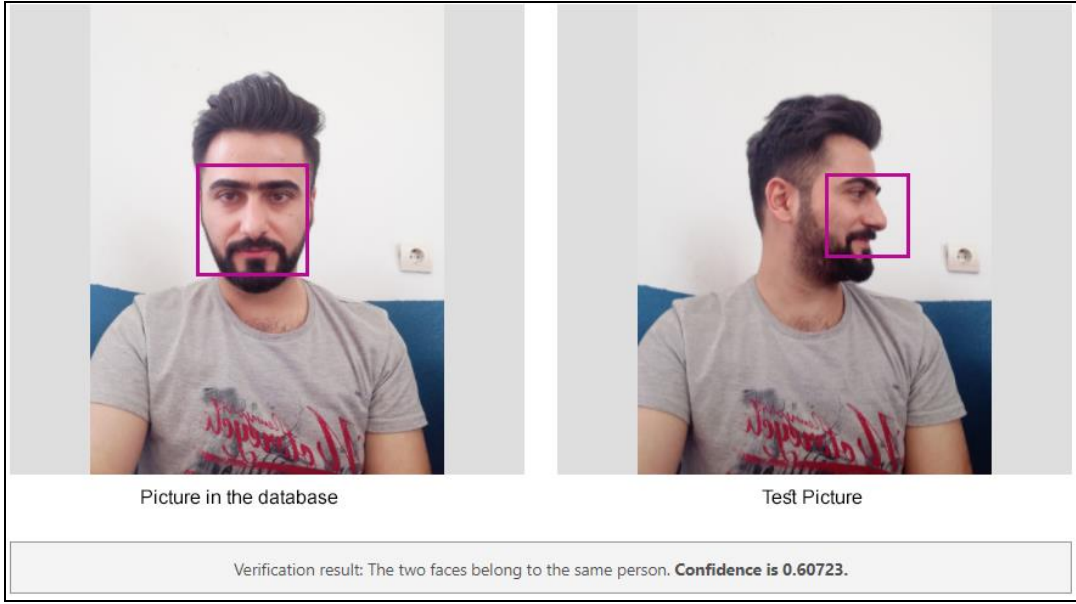
Şekil 4.4 Logitech kamera ile maksimum mesafe 3m

Deneyden, kameranın çözünürlüğü yüksek olursa tanıma doğruluğunun daha iyi olduğu görülmüştür. İkinci deney için Microsoft LifeCam HD-3000'i kullanılmıştır (720p HD). Bu kamera ile ziyaretçinin 8 metreye kadar olan bir mesafeden %78,5 güvenle tanıdığı görülmüştür. Bu deney sonucu aşağıda Şekil 4.5'de gösterilmiştir.

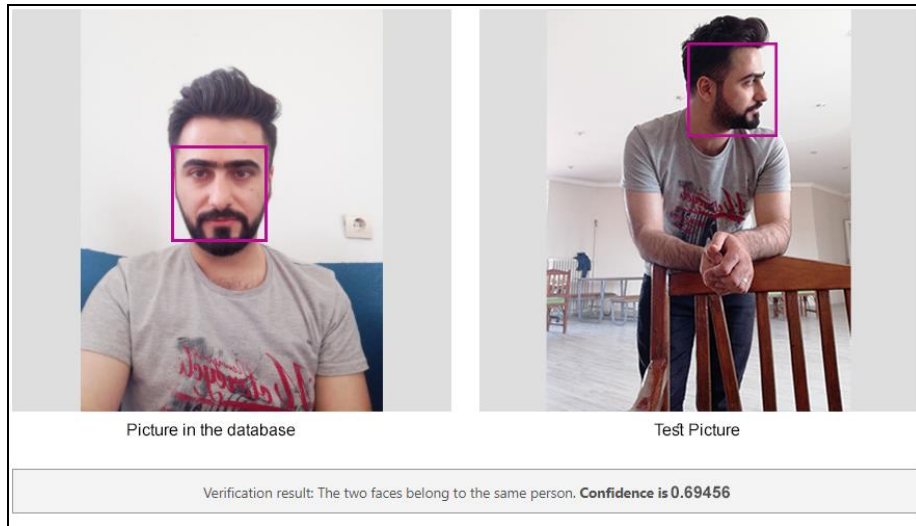


Şekil 4.5 8 metreye kadar mesafe Microsoft LifeCam HD-3000

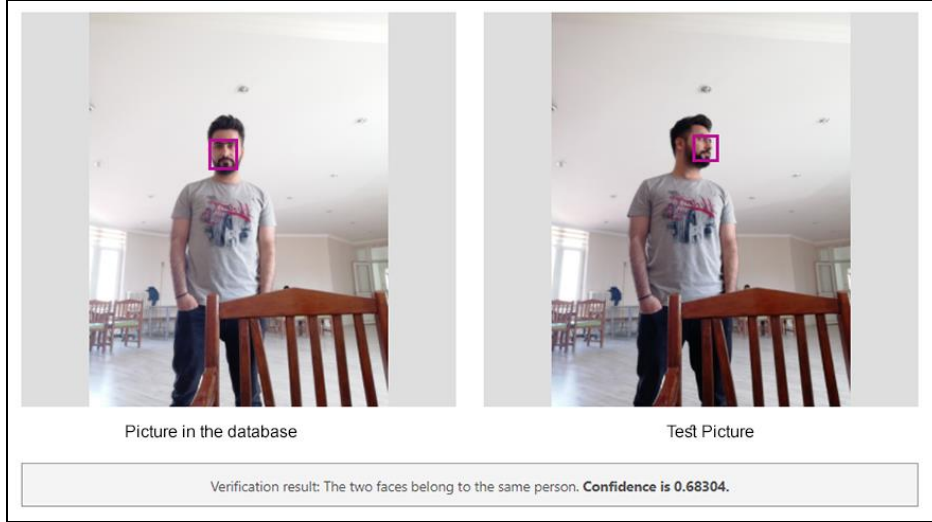
Ayrıca, sistem yüzün açısı 90° olsa bile kişiyi tanımıştır. Kişinin 90° yüz açısı ile farklı mesafelerde yapılan deneylerin sonuçları şekil 4.6, 4.7, 4.8 ve 4.9'da gösterilmiştir.



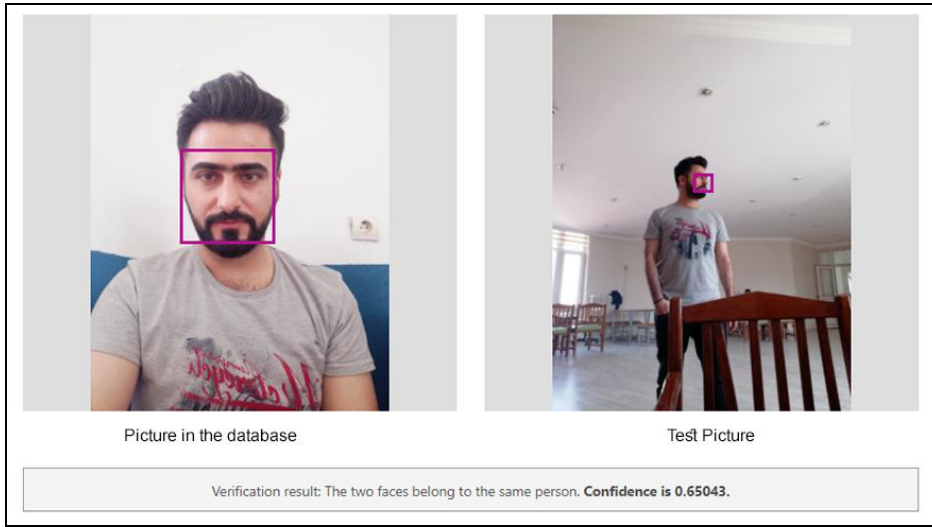
Şekil 4.6 Yüz 90° ve mesafe 0,5 m ile yüz tanıma



Şekil 4.7 90° açısı ve 1 m mesafe ile yüz tanıma



Şekil 4.8 90° açısı ve 1,5 m mesafesi ile yüz tanıma



Şekil 4.9 90° açı ve mesafe 2 m ile yüz tanıma

Elde ettiğimizi deneylerden aşağıdaki tablolarda sonuçları verilmiştir:

Çizelge 4.2 Deney 2. Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	99	84.9	83.4	72.48	69.72
1 m	98.8	84.4	83.49	71.98	69.55
1.5 m	97.2	83.3	81.22	71.12	68.20
2 m	67.4	67.2	66.4	66.01	65.34

Çizelge 4.3 Deney 3.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	98.2	86	82	74.8	71.42
1 m	97.5	85.05	82.9	72.6	69.33
1.5 m	97.12	82	81.34	71.2	68.31
2 m	70.15	64.44	63.5	61.52	65.24

Çizelge 4.4 Deney 4.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	97.3	86	82	74.28	72.32
1 m	97	85.45	83.5	73.65	70.25
1.5 m	95.2	81	80.34	72.4	68
2 m	65.3	65	63.5	62.3	65

Çizelge 4.5 Deney 5.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	97	86	80	73.38	70
1 m	96	85.45	83	72.6	69.4
1.5 m	95	85	82.4	72.2	68.30
2 m	67.15	65.3	64.58	63.43	62.04

Çizelge 4.6 Deney 6.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	98	95	94.5	75	74.72
1 m	97.7	84.45	83	73.6	72.45
1.5 m	93.3	81	82.34	72.2	68.11
2 m	80.12	64.3	62.5	61.43	60.14

Çizelge 4.7 Deney 7.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	98.2	94	85.2	79.38	70.72
1 m	97.5	92.44	84.23	78.2	69.95
1.5 m	96.14	91	83.4	75.2	67.42
2 m	95.15	90.1	82	70.3	66

Çizelge 4.8 Deney 8.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	96	91	88	69.3	65.72
1 m	96.8	88.45	86.01	69.6	64.8
1.5 m	91	85	85.31	66.2	64.30
2 m	80.15	71.3	70.52	65.1	63.04

Çizelge 4.9 Deney 9.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	96	80.5	80	79.38	70.72
1 m	95.2	70.21	79.77	77.6	68.44
1.5 m	94.5	70.19	70	69.2	67.10
2 m	77.2	69.3	68.5	67.43	62.04

Çizelge 4.10 Deney 10.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	88	85	84	74.6	62.99
1 m	87	84	83	73.6	61.45
1.5 m	85	83.5	82.4	72.2	60.70
2 m	75	74	73.5	72.9	60.27

Çizelge 4.11 Deney 11.Yüz tanıma doğruluğu

Distance\Angle	0°	15°	60°	75°	90°
0.5 m	90.5	83.5	80	75.38	74.12
1 m	89.2	81.5	79.29	74.6	73.45
1.5 m	86.45	81	79	73.2	62.19
2 m	84.15	80.3	69.5	67.43	61.40

5. SONUÇ VE ÖNERİLER

5.1 Sonuç

Proje yazılım ve donanımdan oluşur Donanım bölümü, kablosuz klavye / Mouse, kamera ve Raspberry Pi'ye bağlı hoparlörler ile USB bağlantı noktasından, ayrıca Raspberry Pi'ye HDMI kablosuyla bağlı televizyondan oluşur. Düğme için GPIO 5 ve düğme (kapı kilidi) için GPIO 4 kullanılmaktadır. Kapı zilini 5V kapı kilidine bağladık ve her iki düğme ve kapı kilidini, 5V güç kaynağına bağladık. Yazılım bölümü ise veritabanında yeni bir kişi ekleme, güncelleme ve silme işlemleri için Visual studio C # UWP (Windows evrensel uygulama) yazılımını kullanılmıştır. Bu işlemler için UI (Kullanıcı Arayüzü) başarıyla oluşturulmuş. Program ile kişilerin bilgilerini başarıyla kaydedilir, veritabanında düzenlenebilir ve silinebilir. Ayrıca, resim göndermek ve sonuç olarak bilgi almak için API kullanarak Raspberry Pi'nin veritabanına güvenli bağlantı kurulması da sağlanmıştır. Raspberry Pi, kameradan alınan resmi, daha önce veritabanında saklanan resimle başarılı bir şekilde eşleştirme yapması sağlanmıştır. Raspberry Pi, bilinmeyen ziyaretçilerin gelmesi durumunda SMTP servisi kullanarak güvenlik kontrolüne başarılı bir şekilde uyarı mesajları göndermektedir. Mesaj, bilinmeyen ziyaretçinin resmiyle birlikte bir alarm metni içermektedir. Mesaj, bilinmeyen ziyaretçiye ulaştıktan sonra, ziyaretçinin "açık kapı" düğmesine basarak veya "kullanıcı ekle" düğmesiyle yüzünü veritabanına ekleyerek girmesine kullanıcıya izin verilmiştir. Kullanıcının sistemi başarılı bir şekilde kontrol etmesi amaçlanmıştır. Raspberry Pi içerisindeki sistemin UI'sini, istemci tarafından Windows IoT Uzak istemci programı kullanılarak uzaktan kontrol edilmiştir. Sisteme ses sistemini eklenmiştir. Bu sayede sistem ziyaretçinin adıyla karşılama ve reddetme uyarılarını ve hatalarını verebilme imkânına sahiptir. Bu çalışmada, yüz tanıma sistemi kullanılarak otomatik kapı erişim sistemi oluşturulmuştur. Raspberry Pi kontrolörü, PC'den başarılı çıktı alındıktan sonra kapı erişimini kontrol eder. Kapıdan ve monitörden anında yanıtlar gözlemlenmektedir. Yapılan deneyler sonucunda, sistemin yüksek doğrulukta kişiyi tanıma yeteneğine sahip olduğunu gördük. Ayrıca, gelen kişinin yüzü kameraya 90 derece dönmüş olsa bile sistem kişiyi tanımaktadır. Ayrıca, kameranın çözünürlüğü ne kadar yüksekse, tanıma doğruluğu o kadar iyidir. İkinci deney için Microsoft LifeCam HD-3000'i kullandık (720p HD), ziyaretçiyi 8 metreye varan mesafelere kadar güvenle tanıyabileceğimizi gördük. kamera C170 (5 megapiksel) ve güven yüzde

65.156 idi. Bu sistem, terörizm ve kötü niyetli kurumların karşılaştığı, güvenlik ihtiyacının azami olduğu ve güvenliğin tehlikeye giremeyeceği hükümet ve özel binalar gibi birçok yerde kullanılabilir.

5.2 Gelecek İş ve Tavsiye

Raspberry Pi görüntüden yüzü algılayan uygulama ile sadece yüzün bulunduğu çerçeveyi Microsoft Yüz Tanıma API'ye gönderir. Microsoft Yüz Tanıma API, diğer IoT protokolleri ve API'lere ek olarak cihazı daha akıllı hale getirmiştir. İleriki dönemlerde OpenCV platformunu ve dâhili kütüphanelerini de kullanarak daha yüksek doğruluk oranları ile sistemin çalışması yapılabilir.

Kameranın yüksekliğini Microsoft LifeCam HD-3000 (720p HD) gibi yüksek çözünürlükte kullanabileceğimiz sistemin doğruluğunu artırıyoruz. Ayrıca, kullanıcı adınızı ve şifrenizi kullanarak giriş sistemini yaparak sistemin güvenliğini artırabiliriz.

Yüz bilgilerinden elde edilen yaş, cinsiyet ve saçlar ile ilgili detaylar ve görüntüdeki kişinin duygusal durumu da program tarafından algılanması gelecekteki uygulamalarda sistem tarafından bilinmeyen kişileri hatırlamada kullanıcıya faydalı olabilir.

KAYNAKLAR

- Assaf, M. H., Mootoo, R., Das, S. R., Petriu, E. M., Groza, V. ve Biswas, S., 2012, Sensor based home automation and security system, *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, 722-727.
- Avci, S., 2016, Raspberry Pi 3 İlk İncelemesi, <http://1volt1amper.com/Raspberry-pi/Raspberry-pi-3/>: [7 Mayıs, 2018].
- Azure, M., 2018, What is Azure?, <https://azure.microsoft.com/en-us/overview/what-is-azure/>: [24 Nisan, 2018].
- Azure, M., 03/01/2018, How to identify faces in images, <https://docs.microsoft.com/en-us/azure/cognitive-services/face/face-api-how-to-topics/howtoidentifyfacesinimage>: [23 Nisan, 2018].
- Adeoye, O. S., 2010, A survey of emerging biometric technologies, *International Journal of Computer Applications*, 10.
- Andrews, T. J. ve Schluppeck, D., 2004, Neural responses to Mooney images reveal a modular representation of faces in human visual cortex, *Neuroimage*, 21 (1), 91-98.
- Amazon, 2018, Leviton 79A00-1 12-Volt DC Electric Door Strike with Access Control, <https://www.amazon.com/Leviton-79A00-1-12-Volt-Electric-Control/dp/B004HN48BI>: [23 Nisan, 2018].
- Arun, S., Joshua, V., Pillai, A. ve Nageswaran, A., 2014, Fingerprints Age Determination in Different Methods Using Image Processing, *International Journal of Soft Computing*, 9 (2), 85-87.
- Assaf, M. H., Mootoo, R., Das, S. R., Petriu, E. M., Groza, V. ve Biswas, S., 2012, Sensor based home automation and security system, *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, 722-727.
- AVCI, S., 2016, Raspberry Pi 3 İlk İncelemesi, <http://1volt1amper.com/raspberry-pi/raspberry-pi-3/>: [7 Mayıs, 2018].
- Azure, M., 2018, What is Azure?, <https://azure.microsoft.com/en-us/overview/what-is-azure/>: [24 Nisan, 2018].
- Biosis, 2018, LOGITECH C270 HD Dahili Mikrofonlu Webcam 960-001063, <https://www.biosis.com.tr/logitech-c270-hd-dahili-mikrofonlu-webcam-960-001063>: [23 Nisan, 2018].
- Blair, D., November 10, 2009, COMPUTATIONAL HUMAN GENETICS - SEARCHING FOR RELATIONS BETWEEN GENES, DISEASES, AND POPULATIONS, <http://slideplayer.com/slide/8342815/>: [23 Nisan, 2018].
- Blanz, V. ve Vetter, T., 2003, Face recognition based on fitting a 3D morphable model, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25 (9), 1063-1074.
- Blair, D., November 10, 2009, COMPUTATIONAL HUMAN GENETICS - SEARCHING FOR RELATIONS BETWEEN GENES, DISEASES, AND POPULATIONS, <http://slideplayer.com/slide/8342815/>: [23 Nisan, 2018].
- Chellappa, R., Wilson, C. L. ve Sirohey, S., 1995, Human and machine recognition of faces: A survey, *Proceedings of the IEEE*, 83 (5), 705-741.
- Chowdhury, M. N., Nooman, M. S. ve Sarker, S., 2013, Access Control of Door and Home Security by Raspberry Pi Through Internet, *Int. J. Sci. Eng. Res*, 4, 550-558.

- Developer, M., 18 Nisan, 2018, Get started with Windows 10 IoT Core, <https://developer.microsoft.com/en-us/windows/iot/getstarted>: [23 Nisan, 2018].
- Diener, D., 11th May 2016, 4 Cool Raspberry Pi Projects to Check Out, <https://www.maketecheasier.com/cool-Raspberry-pi-projects/>: [23 Nisan, 2018].
- Diener, D., 2016, 4 Cool Raspberry Pi Projects to Check Out, <https://www.maketecheasier.com/cool-raspberry-pi-projects/>: [23 Nisan, 2018].
- Dunstone, T. ve Yager, N., 2008, Biometric system and data analysis: Design, evaluation, and data mining, Springer Science & Business Media, p.
- Edwards, G. J., Cootes, T. F. ve Taylor, C. J., 1998, Face recognition using active appearance models, *European conference on computer vision*, 581-595.
- Essabre-90xx-rpi, Februar 25, 2018, DIY: RELAY-CONTROLLED SOURCE-FILTER SELECTION ES9038Q2M (ARDUINO), <http://essabre-90xx-rpi.sfb2.com/category/stm32f103-arduino-esp32-8266/>: [23 Nisan, 2018].
- Elinux, 3 January 2018, RPi USB Webcams, https://elinux.org/RPi_USB_Webcams: [23 Nisan, 2018].
- Essabre-90xx-rpi, Februar 25, 2018, DIY: RELAY-CONTROLLED SOURCE-FILTER SELECTION ES9038Q2M (ARDUINO), <http://essabre-90xx-rpi.sfb2.com/category/stm32f103-arduino-esp32-8266/>: [23 Nisan, 2018].
- Ferdous, S. I. ve Ahmed, S., 2008, Literature survey of automatic face recognition system and eigenface based implementation, *BRAC University*.
- Ferdous, S. I. ve Ahmed, S., 2008, Literature survey of automatic face recognition system and eigenface based implementation, *BRAC University*.
- Filković, I., Kalafatić, Z. ve Hrkać, T., 2016, Deep metric learning for person re-identification and de-identification, *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2016 39th International Convention on*, 1360-1364.
- Fox, N. A., Gross, R., Cohn, J. F. ve Reilly, R. B., 2007, Robust biometric person identification using automatic classifier fusion of speech, mouth, and face experts, *IEEE Transactions on multimedia*, 9 (4), 701-714.
- Goudelis, G., Karpouzis, K. ve Kollias, S., 2013, Exploring trace transform for robust human action recognition, *Pattern Recognition*, 46 (12), 3238-3248.
- Guo, G., Li, S. Z. ve Chan, K., 2000, Face recognition by support vector machines, *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 196-201.
- Grudin, M. A., 2000, On internal representations in face recognition systems, *Pattern recognition*, 33 (7), 1161-1177.
- Hassan, H., Bakar, R. A. ve Mokhtar, A. T. F., 2012, Face recognition based on auto-switching magnetic door lock system using microcontroller, *System Engineering and Technology (ICSET), 2012 International Conference on*, 1-6.
- Hassan, H., Bakar, R. A. ve Mokhtar, A. T. F., 2012, Face recognition based on auto-switching magnetic door lock system using microcontroller, *System Engineering and Technology (ICSET), 2012 International Conference on*, 1-6.
- Hariharan, S., 2017, Logic Apps and Cognitive Services Face API – Part 1 (Middleware Friday), <https://blogs.biztalk360.com/logic-apps-and-cognitive-services-face-api-part-1-middleware-friday/>: [3 Haziran 2018].
- Hassan, H., Bakar, R. A. ve Mokhtar, A. T. F., 2012, Face recognition based on auto-switching magnetic door lock system using microcontroller, *System Engineering and Technology (ICSET), 2012 International Conference on*, 1-6.

- Ibrahim, R. ve Zin, Z. M., 2011, Study of automated face recognition system for office door access control application, *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, 132-136.
- Ibrahim, R. ve Zin, Z. M., 2011, Study of automated face recognition system for office door access control application, *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, 132-136.
- Jain, A. K. ve Li, S. Z., 2011, Handbook of face recognition, Springer, p.
- Jain, A. K. ve Li, S. Z., 2011, Handbook of face recognition, Springer, p.
- Turk, M. ve Pentland, A., 1991, Eigenfaces for recognition, *Journal of cognitive neuroscience*, 3 (1), 71-86.
- Jain, A. K. ve Li, S. Z., 2011, Handbook of face recognition, Springer, p.
- JetBrains, 2017, What algorithms does Face API used?, [algorithms-does-Face-API-used:](https://www.quora.com/What-Kim, J., Choi, J., Yi, J. ve Turk, M., 2005, Effective representation using ICA for face recognition robust to local distortion and partial occlusion, <i>IEEE Transactions on Pattern Analysis and Machine Intelligence</i>, 27 (12), 1977-1981.</p>
<p>Konen, W. ve Schulze-kruger, E., 1995, ZN-Face: A system for access control using automated face recognition, <i>In Proceedings of the International Workshop on Automated Face-and Gesture-Recognition</i>.</p>
<p>Kosugi, M., 1995, Human-face search and location in a scene by multi-pyramid architecture for personal identification, <i>Systems and computers in Japan</i>, 26 (6), 27-38.</p>
<p><a href=) [3 Haziran 2018].
- Kalocsai, P., von der Malsburg, C. ve Horn, J., 2000, Face recognition by statistical analysis of feature detectors, *Image and Vision Computing*, 18 (4), 273-278.
- Kosugi, M., 1995, Human-face search and location in a scene by multi-pyramid architecture for personal identification, *Systems and computers in Japan*, 26 (6), 27-38.
- Kruger, N., 1997, An algorithm for the learning of weights in discrimination functions using a priori constraints, *IEEE transactions on pattern analysis and machine intelligence*, 19 (7), 764-768.
- Kılıçdağı, A., Haziran 2014, Raspberry Pi, p. 2-3.
- Kılıçdağı, A., Haziran 2014, Raspberry Pi, p. 2-3.
- Lee, J. H., Kim, Y.-S., Kim, B. K., Ohba, K., Kawata, H. ve Ohya, A., 2007, Security door system using human tracking method with laser range finders, *Mechatronics and Automation, 2007. ICMA 2007. International Conference on*, 2060-2065.
- Lee, J. H., Kim, Y.-S., Kim, B. K., Ohba, K., Kawata, H. ve Ohya, A., 2007, Security door system using human tracking method with laser range finders, *Mechatronics and Automation, 2007. ICMA 2007. International Conference on*, 2060-2065.
- Li, S. Z. ve Lu, J., 1999, Face recognition using the nearest feature line method, *IEEE Transactions on Neural networks*, 10 (2), 439-443.
- Liu, C. ve Wechsler, H., 1998, A unified Bayesian framework for face recognition, *Image Processing, 1998. ICIIP 98. Proceedings. 1998 International Conference on*, 151-155.
- Liu, C. ve Wechsler, H., 2000, Evolutionary pursuit and its application to face recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 (6), 570-582.

- Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B. ve Song, L., 2017, Spheraface: Deep hypersphere embedding for face recognition, *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Lu, J., Plataniotis, K. N. ve Venetsanopoulos, A. N., 2003, Face recognition using LDA-based algorithms, *IEEE Transactions on Neural networks*, 14 (1), 195-200.
- Lu, J., Plataniotis, K. N., Venetsanopoulos, A. N. ve Li, S. Z., 2006, Ensemble-based discriminant learning with boosting for face recognition, *IEEE Transactions on Neural networks*, 17
- McKenna, S. J. ve Gong, S., 1997, Non-intrusive person authentication for access control by visual tracking and face recognition, *International Conference on Audio-and Video-Based Biometric Person Authentication*, 177-183.
- Mehrdad, Oct 15, 2013, Face recognition, https://www.slideshare.net/mehrdad_1992/face-recognition-27218990: [23 Nisan, 2018].
- Microsoft, 2016, LifeCam HD-3000, <https://www.microsoft.com/accessories/en-us/products/webcams/lifecam-hd-3000/t3h-00011>: [23 Nisan, 2018].
- Microsoft.Azure, 2018, How to Detect Faces in Image, <https://docs.microsoft.com/en-us/azure/cognitive-services/face/face-api-how-to-topics/howtodetectfacesinimage>: [24 Nisan, 2018].
- Moghaddam, B. ve Pentland, A., 1995, An automatic system for model-based coding of faces, *Data Compression Conference, 1995. DCC'95. Proceedings*, 362-370.
- Motoyama, S. i. ve Kaneda, S., 1989, Low-temperature growth of 3C-SiC by the gas source molecular beam epitaxial method, *Applied Physics Letters*, 54 (3), 242-243.
- (1), 166-178.
- McKenna, S. J. ve Gong, S., 1997, Non-intrusive person authentication for access control by visual tracking and face recognition, *International Conference on Audio-and Video-Based Biometric Person Authentication*, 177-183.
- Moghaddam, B. ve Pentland, A., 1995, An automatic system for model-based coding of faces, *Data Compression Conference, 1995. DCC'95. Proceedings*, 362-370.
- Mehrdad, Oct 15, 2013, Face recognition, https://www.slideshare.net/mehrdad_1992/face-recognition-27218990: [23 Nisan, 2018].
- Microsoft.Azure, 2018, How to Detect Faces in Image, <https://docs.microsoft.com/en-us/azure/cognitive-services/face/face-api-how-to-topics/howtodetectfacesinimage>: [24 Nisan, 2018].
- MicrosoftAzure, 2018, How to add faces, <https://docs.microsoft.com/en-us/azure/cognitive-services/face/face-api-how-to-topics/how-to-add-faces>: [23 Nisan, 2018].
- Nefian, A. V. ve Hayes, M. H., 1998, Hidden Markov models for face recognition, *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, 2721-2724.
- Planetbiometrics, 15 November 2011, New report predicts Global Biometrics Market to reach US\$16.47 Billion, <http://www.planetbiometrics.com/article-details/i/917/>: [20 Nisan, 2018].
- Rncos, Jul 20, 2011, Global Facial Recognition Market to Witness Double Digit Growth, https://www.rncos.com/Press_Releases/Global-Facial-Recognition-Market-to-Witness-Double-Digit-Growth.htm?pagename=Global-Facial-Recognition-Market-to-Witness-Double-Digit-Growth: [20 Nisan, 2018].

- Adeoye, O. S., 2010, A survey of emerging biometric technologies, *International Journal of Computer Applications*, 10.
- Padmapriya, S. ve KalaJames, E. A., 2012, Real time smart car lock security system using face detection and recognition, *Computer Communication and Informatics (ICCCI), 2012 International Conference on*, 1-6.
- Papatheodorou, T. ve Rueckert, D., 2007, 3D face recognition, In: *Face Recognition*, Eds: InTech, p.
- Petrini, A.-C. ve Ionescu, V.-M., 2016, Implementation of the huffman coding algorithm in windows 10 IoT core, *Electronics, Computers and Artificial Intelligence (ECAI), 2016 8th International Conference on*, 1-6.
- planetbiometrics, 15 November 2011, New report predicts Global Biometrics Market to reach US\$16.47 Billion, <http://www.planetbiometrics.com/article-details/i/917/>: [20 Nisan, 2018].
- Prasad, S., Mahalakshmi, P., Sunder, A. J. C. ve Swathi, R., 2014, Smart surveillance monitoring system using Raspberry Pi and PIR sensor, *Int. J. Comput. Sci. Inf. Technol*, 5 (6), 7107-7109.
- Raspberrypi, 2015, NOOBS, <https://www.raspberrypi.org/downloads/noobs/>: [24 Nisan, 2018].
- Rncos, Jul 20, 2011, Global Facial Recognition Market to Witness Double Digit Growth, https://www.rncos.com/Press_Releases/Global-Facial-Recognition-Market-to-Witness-Double-Digit-Growth.htm?pagename=Global-Facial-Recognition-Market-to-Witness-Double-Digit-Growth:
- Raspberrypi, 2015c, RASBIAN , <https://www.Raspberrypi.org/downloads/rasbian/>: [24 Nisan, 2018].
- Shaheen, F., Verma, B. ve Asafuddoula, M., 2016, Impact of automatic feature extraction in deep learning architecture, *Digital Image Computing: Techniques and Applications (DICTA), 2016 International Conference on*, 1-8.
- Sinha, P., Balas, B., Ostrovsky, Y. ve Russell, R., 2006, Face recognition by humans: Nineteen results all computer vision researchers should know about, *Proceedings of the IEEE*, 94 (11), 1948-1962.
- social.Microsoft, 2016, What algorithms does Face API used?, <https://social.msdn.microsoft.com/Forums/sqlserver/en-US/0e5e4521-9ec3-45f9-a04e-48b4af2d229f/what-algorithms-does-face-api-used?forum=mlapi>: [3 Haziran 2018].
- Syafeeza, A., Khalil-Hani, M., Liew, S. ve Bakhteri, R., 2014, Convolutional neural network for face recognition with pose and illumination variation, *International Journal of Engineering & Technology*, 6 (1), 0975-4024.
- Turk, M. ve Pentland, A., 1991, Eigenfaces for recognition, *Journal of cognitive neuroscience*, 3 (1), 71-86.
- Useche, M. P. C., Arenas, J. O. P. ve Moreno, R. J., 2018, Face Recognition Access Control System using Convolutional Neural Networks, *Research Journal of Applied Sciences*, 13 (1), 47-53.
- Wiskott, L., Krüger, N., Kuiger, N. ve Von Der Malsburg, C., 1997, Face recognition by elastic bunch graph matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19 (7), 775-779.
- Yang, G. ve Huang, T. S., 1994, Human face detection in a complex background, *Pattern recognition*, 27 (1), 53-63.
- Yang, M.-H., 2002, Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using Kernel Methods, *Fgr*, 215.

- Zhao, Y. ve Ye, Z., 2008, A low cost GSM/GPRS based wireless home security system, *IEEE Transactions on Consumer Electronics*, 54 (2).
- Zhao, Z.-Q., Huang, D.-S. ve Sun, B.-Y., 2004, Human face recognition based on multi-features using neural networks committee, *Pattern Recognition Letters*, 25 (12), 1351-1358.
- Zhao, Y. ve Ye, Z., 2008, A low cost GSM/GPRS based wireless home security system, *IEEE Transactions on Consumer Electronics*, 54 (2).

EKLER**EK-1** Yüz Tespiti Programı.

C#

```
1. using System;
2. using System.IO;
3. using System.Net.Http;
4. using System.Net.Http.Headers;
5. using System.Text;
6.
7. namespace CSHttpClientSample
8. {
9.     static class Program
10.    {
11.        // *****
12.        // *** Update or verify the following values. ***
13.        // *****
14.
15.        // Replace the subscriptionKey string value with your valid
        // subscription key.
16.        const string subscriptionKey = "<Subscription Key>";
17.
18.        // Replace or verify the region.
19.        //
20.        // You must use the same region in your REST API call as you
        // used to obtain your subscription keys.
21.        // For example, if you obtained your subscription keys from
        // the westus region, replace
22.        // "westcentralus" in the URI below with "westus".
23.        //
24.        // NOTE: Free trial subscription keys are generated in the
        // westcentralus region, so if you are using
25.        // a free trial subscription key, you should not need to
        // change this region.
26.        const string uriBase =
        "https://westcentralus.api.cognitive.microsoft.com/face/v1.0/detect";
27.
28.
29.        static void Main()
30.        {
31.            // Get the path and filename to process from the user.
32.            Console.WriteLine("Detect faces:");
33.            Console.Write("Enter the path to an image with faces that
            you wish to analyze: ");
34.            string imageFilePath = Console.ReadLine();
35.
```

```

36.         // Execute the REST API call.
37.         MakeAnalysisRequest(imageFilePath);
38.
39.         Console.WriteLine("\nPlease wait a moment for the results
to appear. Then, press Enter to exit...\n");
40.         Console.ReadLine();
41.     }
42.
43.
44.     /// <summary>
45.     /// Gets the analysis of the specified image file by using the
Computer Vision REST API.
46.     /// </summary>
47.     /// <param name="imageFilePath">The image file.</param>
48.     static async void MakeAnalysisRequest(string imageFilePath)
49.     {
50.         HttpClient client = new HttpClient();
51.
52.         // Request headers.
53.         client.DefaultRequestHeaders.Add("Ocp-Apim-Subscription-
Key", subscriptionKey);
54.
55.         // Request parameters. A third optional parameter is
"details".
56.         string requestParameters =
"returnFaceId=true&returnFaceLandmarks=false&returnFaceAttributes=age,
gender,headPose,smile,facialHair,glasses,emotion,hair,makeup,occlusion
,accessories,blur,exposure,noise";
57.
58.         // Assemble the URI for the REST API Call.
59.         string uri = uriBase + "?" + requestParameters;
60.
61.         HttpResponseMessage response;
62.
63.         // Request body. Posts a locally stored JPEG image.
64.         byte[] byteData = GetImageAsByteArray(imageFilePath);
65.
66.         using (ByteArrayContent content = new
ByteArrayContent(byteData))
67.         {
68.             // This example uses content type "application/octet-
stream".
69.             // The other content types you can use are
"application/json" and "multipart/form-data".
70.             content.Headers.ContentType = new
MediaTypeHeaderValue("application/octet-stream");
71.
72.             // Execute the REST API call.

```

```

73.         response = await client.PostAsync(uri, content);
74.
75.         // Get the JSON response.
76.         string contentString = await
response.Content.ReadAsStringAsync();
77.
78.         // Display the JSON response.
79.         Console.WriteLine("\nResponse:\n");
80.         Console.WriteLine(JsonPrettyPrint(contentString));
81.     }
82. }
83.
84.
85.     /// <summary>
86.     /// Returns the contents of the specified file as a byte
array.
87.     /// </summary>
88.     /// <param name="imageFilePath">The image file to
read.</param>
89.     /// <returns>The byte array of the image data.</returns>
90.     static byte[] GetImageAsByteArray(string imageFilePath)
91.     {
92.         FileStream fileStream = new FileStream(imageFilePath,
FileStream.Open, FileAccess.Read);
93.         BinaryReader binaryReader = new BinaryReader(fileStream);
94.         return binaryReader.ReadBytes((int)fileStream.Length);
95.     }
96.
97.
98.     /// <summary>
99.     /// Formats the given JSON string by adding line breaks and
indents.
100.    /// </summary>
101.    /// <param name="json">The raw JSON string to
format.</param>
102.    /// <returns>The formatted JSON string.</returns>
103.    static string JsonPrettyPrint(string json)
104.    {
105.        if (string.IsNullOrEmpty(json))
106.            return string.Empty;
107.
108.        json = json.Replace(Environment.NewLine,
""").Replace("\t", "");
109.
110.        StringBuilder sb = new StringBuilder();
111.        bool quote = false;
112.        bool ignore = false;
113.        int offset = 0;

```

```

114.         int indentLength = 3;
115.
116.         foreach (char ch in json)
117.         {
118.             switch (ch)
119.             {
120.                 case '"':
121.                     if (!ignore) quote = !quote;
122.                     break;
123.                 case '\\':
124.                     if (quote) ignore = !ignore;
125.                     break;
126.             }
127.
128.             if (quote)
129.                 sb.Append(ch);
130.             else
131.             {
132.                 switch (ch)
133.                 {
134.                     case '{':
135.                     case '[':
136.                         sb.Append(ch);
137.                         sb.Append(Environment.NewLine);
138.                         sb.Append(new string(' ', ++offset *
139.                             indentLength));
140.                         break;
141.                     case '}':
142.                     case ']':
143.                         sb.Append(Environment.NewLine);
144.                         sb.Append(new string(' ', --offset *
145.                             indentLength));
146.                         sb.Append(ch);
147.                         break;
148.                     case ',':
149.                         sb.Append(ch);
150.                         sb.Append(Environment.NewLine);
151.                         sb.Append(new string(' ', offset *
152.                             indentLength));
153.                         break;
154.                     case ':':
155.                         sb.Append(ch);
156.                         sb.Append(' ');
157.                         break;
158.                     default:
159.                         if (ch != ' ') sb.Append(ch);
160.                         break;
161.                 }
162.             }
163.         }
164.     }
165. }

```

```
159.         }
160.     }
161.
162.     return sb.ToString().Trim();
163. }
164. }
165. }
```

EK-2 JSON Kodu**Detection** result:

JSON:

```
[
  {
    "faceId": "e18f33ce-8f91-483c-a74a-562a4b503b18",
    "faceRectangle": {
      "top": 104,
      "left": 109,
      "width": 109,
      "height": 109
    },
    "faceAttributes": {
      "hair": {
        "bald": 0.01,
        "invisible": false,
        "hairColor": [
          {
            "color": "brown",
            "confidence": 1.0
          },
          {
            "color": "black",
            "confidence": 0.85
          },
          {
            "color": "other",
            "confidence": 0.23
          },
          {
            "color": "gray",
            "confidence": 0.18
          },
          {
            "color": "red",
            "confidence": 0.05
          },
          {
            "color": "blond",
            "confidence": 0.03
          }
        ]
      },
      "smile": 1.0,
      "headPose": {
        "pitch": 0.0,
        "roll": -6.8,
        "yaw": -0.4
      },
      "gender": "male",
      "age": 32.4,
      "facialHair": {
        "moustache": 0.3,
        "beard": 0.3,
        "sideburns": 0.3
      }
    }
  }
]
```



```
},
"glasses": "ReadingGlasses",
"makeup": {
  "eyeMakeup": true,
  "lipMakeup": true
},
"emotion": {
  "anger": 0.0,
  "contempt": 0.0,
  "disgust": 0.0,
  "fear": 0.0,
  "happiness": 1.0,
  "neutral": 0.0,
  "sadness": 0.0,
  "surprise": 0.0
},
"occlusion": {
  "foreheadOccluded": false,
  "eyeOccluded": false,
  "mouthOccluded": false
},
"accessories": [
  {
    "type": "glasses",
    "confidence": 0.98
  }
],
"blur": {
  "blurLevel": "low",
  "value": 0.0
},
"exposure": {
  "exposureLevel": "overExposure",
  "value": 0.8
},
"noise": {
  "noiseLevel": "medium",
  "value": 0.42
}
},
"faceLandmarks": {
  "pupilLeft": {
    "x": 135.9,
    "y": 138.7
  },
  "pupilRight": {
    "x": 184.7,
    "y": 131.8
  },
  "noseTip": {
    "x": 163.6,
    "y": 168.4
  },
  "mouthLeft": {
    "x": 138.4,
    "y": 184.4
  }
}
```

```
},
"mouthRight": {
"x": 193.7,
"y": 176.4
},
},
"eyebrowLeftOuter": {
"x": 112.9,
"y": 137.0
},
},
"eyebrowLeftInner": {
"x": 144.4,
"y": 128.0
},
},
"eyeLeftOuter": {
"x": 128.9,
"y": 140.1
},
},
"eyeLeftTop": {
"x": 136.5,
"y": 136.8
},
},
"eyeLeftBottom": {
"x": 136.1,
"y": 141.4
},
},
"eyeLeftInner": {
"x": 143.6,
"y": 139.4
},
},
"eyebrowRightInner": {
"x": 172.5,
"y": 124.9
},
},
"eyebrowRightOuter": {
"x": 206.1,
"y": 124.7
},
},
"eyeRightInner": {
"x": 178.5,
"y": 134.3
},
},
"eyeRightTop": {
"x": 185.1,
"y": 130.5
},
},
"eyeRightBottom": {
"x": 185.0,
"y": 135.0
},
},
"eyeRightOuter": {
"x": 191.6,
"y": 132.6
},
},
"noseRootLeft": {
"x": 154.5,
"y": 138.5
```

```
},
"noseRootRight": {
"x": 166.7,
"y": 136.9
},
"noseLeftAlarTop": {
"x": 149.5,
"y": 157.6
},
"noseRightAlarTop": {
"x": 173.8,
"y": 155.2
},
"noseLeftAlarOutTip": {
"x": 143.6,
"y": 166.3
},
"noseRightAlarOutTip": {
"x": 182.0,
"y": 161.7
},
"upperLipTop": {
"x": 164.9,
"y": 181.9
},
"upperLipBottom": {
"x": 165.2,
"y": 184.8
},
"underLipTop": {
"x": 166.6,
"y": 193.2
},
"underLipBottom": {
"x": 167.2,
"y": 199.0
}
}
}
```

EK-3 K1şı Grubu oluřturma Kodu

C#

```
using System;
using System.Net.Http.Headers;
using System.Net.Http;

namespace CSHttpClientSample
{
    static class Program
    {
        static void Main()
        {
            Console.WriteLine("Enter an ID for the group you wish to create:");
            Console.WriteLine("(Use numbers, lower case letters, '-' and '_'. The maximum length of the personGroupId is 64.)");

            string personGroupId = Console.ReadLine();
            MakeCreateGroupRequest(personGroupId);

            Console.WriteLine("\n\nWait for the result below, then hit ENTER to exit...\n\n");
            Console.ReadLine();
        }

        static async void MakeCreateGroupRequest(string personGroupId)
        {
            var client = new HttpClient();

            // İstek üstbilgileri- bu örnek anahtarı geçerli anahtarınızla deęiřtirin.
            client.DefaultRequestHeaders.Add("Ocp-Apim-Subscription-Key", "<Subscription Key>");

            // URI dizesi iste.
            // NOT: Abonelik anahtarlarınızı almak için kullandığınız REST çağrısında aynı bölgeyi kullanmalıyız.
            // Örneęin, abonelik anahtarlarını westus'tan aldıysanız, "westcentralus" ifadesini
            // "Ufus" ile ařaęıdaki URI."
            string uri = "https://westcentralus.api.cognitive.microsoft.com/face/v1.0/persongroups/" + personGroupId;

            // Burada "isim" gösterilmek ve benzersiz olmak zorunda deęildir. Ayrıca, "userData" isteęe baęlıdır.
            string json = "{\"name\":\"My Group\", \"userData\":\"Some data related to my group.\"}";
        }
    }
}
```

```
HttpContent content = new StringContent(json);
content.Headers.ContentType = new MediaTypeHeaderValue("application/json");

HttpResponseMessage response = await client.PutAsync(uri, content);

// Grup başarıyla oluşturulduysa, "Tamam" ı görürsünüz. // Otherwise, if a
group with the same personGroupId has been created before, you'll see
"Conflict".

Console.WriteLine("Response status: " + response.StatusCode);
}
}
}
```

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : EEVİR ABDUL QADER RASHID RASHID
Uyruğu : IRAQ
Doğum Yeri ve Tarihi : KERUK 14/01/1991
Telefon : 05389358908
Faks :
e-mail : Ayver.dabbagh@gmail.com

EĞİTİM

Derece	Adı, İlçe, İl	Bitirme Yılı
Lise	: Kerkuk kız lisesi ,kerkuk	2010
Üniversite	: AL QALAM Üniversite	2015
Yüksek Lisans :		
Doktora :		

İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
-----	-------	--------

UZMANLIK ALANI

YABANCI DİLLER

BELİRTMEK İSTEĞİNİZ DİĞER ÖZELLİKLER

YAYINLAR

Eevir RASHID, Cemil SUNGUR, 2018. RASPBERRY BASED AUTOMATED FACE RECOGNITION FOR SECURITY SYSTEMS. International Conference on Engineering Technologies ICENTE 18, Konya, Turkey.