

T.C
SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

YÜZ TANIMA GÜVENLİK SİSTEMİ

BİTİRME ÇALIŞMASI
SUPHANİCHA MAHACHOKTHORANEE
YUSUF ZALT

Fakülte Anabilim Dalı : BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ

Tez Danışmanı : Dr.Öğr.Üyesi FATİH ÇALLI

İçindekiler

Giriş.....	4
Yüz Tanıma	5
PyCharm	6
Dlib	6
Python ile Yüz Tespiti Nasıl Yapılır	7
DLib nasıl çalışıyor.....	7
Gerekli Kütüphanelerin Yüklenmesi.....	8
OpenCV ile Yüz Tespiti	8
Literatür Taraması.....	9
YÜZ TANIMA ALGORİTMALARI.....	10
Yüz tanımadaki üç temel unsur şöyle tanımlanır:.....	12
Yüz Tanıma Sisteminin Ana Yapıtaşları	12
Devşirilme ve kayıt.....	12
Veri işleme	12
Avantajlar ve Dezavantajlar	13
Diğer biyometrik sistemlerle karşılaştırıldığında	13
Zayıf Yönler	13
İnsan Yüzü Tanımadaki Daha Önceden Yapılmış Çalışmalar	14
1-İnsanların Kullandıkları Yüz Karakteristikleri.....	14
2-Template Matching.....	14
3-Özellik Tabanlı Denkleştirme	14
Yüz tanıma nasıl çalışır	15
Yüz Tanıma ile İlgili Sorunlar	16
OPEN-CV	17
Uygulama alanları	17
OPENCV'DE YÜZ TANIMA.....	18
GUI(Graphical User Interface)- tkinter	18
Uygulama.....	20
Veri Tolama	20
Yüz Tanıma Metodolojisi.....	20
Yüz tanıma	20
Dlib ile Yüz Hizalama	20

Veri Tabanına kayıt	21
Son adım olarak algılanan yüzün özellikleri ve kırıldıktan sonra veri tabanına kayıt ediyoruz.	21
İki Resim Karşılaştırma	22
WebCam ile Yüz Tanıma.....	23
MICROSOFT'UN FACE API İLE YÜZ TANIMA	24
Model oluşturma	24
Kaynakça.....	25

Giriş

Yüz algılama, dijital görüntülerde insan yüzlerini bulmaya / görselleştirmeye yardımcı olan bir bilgisayarla görme teknolojisidir.) dijital görüntülerde ve videolarda. Teknolojinin gelişmesiyle birlikte yüz tanıma, özellikle fotoğrafçılık, güvenlik ve pazarlama gibi alanlarda büyük önem kazandı.

Bu çalışmada hedefimiz video kameradan alınan görüntüleri kullanarak kimlik tespiti yapabilmektir. Alınan görüntüler ile ilgili bilgiler bir veri tabanında tutulacaktır. Yeni bir görüntü alındığında veri tabanı ile karşılaştırılacak eğer bulunursa kimliği tespit edilecek, bulunamazsa sistem tanımayacaktır. Alınan görüntünün farklı açılardan olabileceği göz önüne alınarak insan yüzünün bilgileri çeşitli yöntemler uygulanarak bulunacaktır. İnsan yüzü tanıma kullanılan teknikler iki grupta toplanabilir : statik (kontrollü ve düzenli fotoğraflardan tanıma) ve dinamik (kontROLSÜZ video görüntülerinden tanıma) uyumu. Bu gruplar arasında görüntü kalitesine ve arka planın karmaşıklığına bağlı olarak önemli farklılıklar vardır. Eğer görüntü kalitesi iyi değilse yapılabilecekler : parlaklık ayarı, uygun filtreleme yöntemini kullanma ya da uygun eşik değerini uygulamaktır. Eğer arka plan ile ilgili sorunlar varsa uygun bölümlendirme ve sınıflama teknikleri kullanılarak iyi sonuçlar alınabilir.

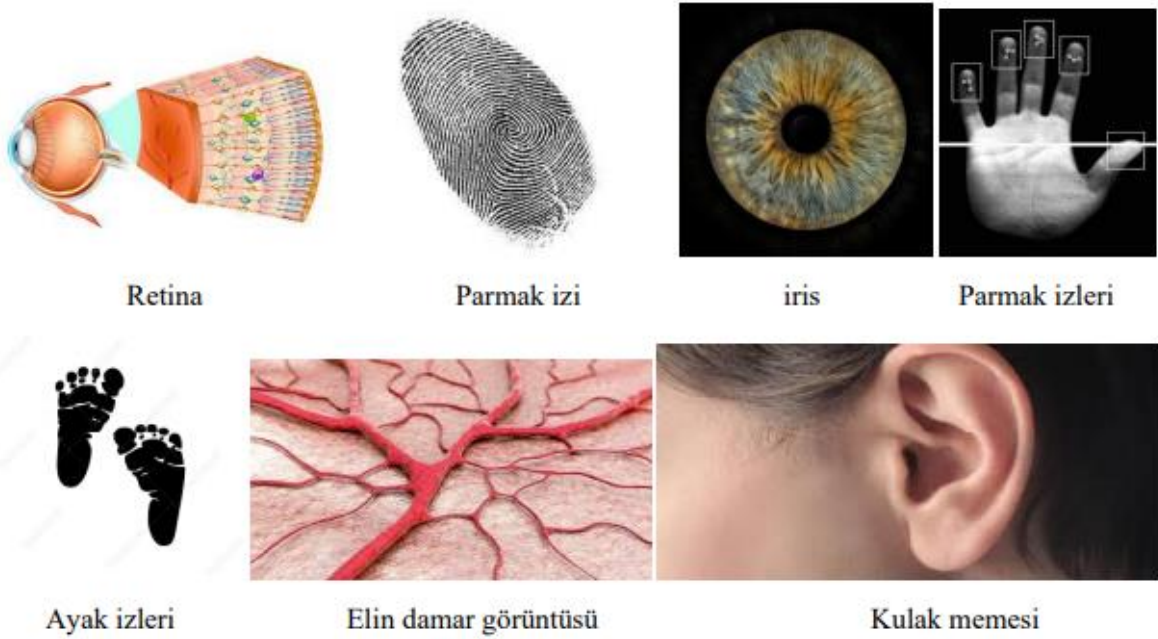
Yüz tanıma, biyometri, gözetim, güvenlik ve kimlikten kimlik doğrulamaya kadar çeşitli uygulamalarda hayati bir rol oynar. Bu yazıda, erişimin eğitim veritabanında yüzleri bulunan kişilerle sınırlı olduğu kısıtlı alan için akıllı bir güvenlik sistemi tasarlıyor ve uyguluyoruz. Önce insan hareketini algılayarak yüzü tespit edeceğiz. Ardından kişinin alana girme yetkisini belirlemek için yüz tanıma işlemi yapılır. Sistem yüzü tanımazsa Bu kişinin alana girmesine izin verilmeyecektir.

Yüz tanıma güvenlik sistemleri, biyometrik tanıma sistemleri içerisinde şüphesiz ki en güvenilir olan ve en başarılı olan sistemlerdir. Bu sistemler, kişilerin eşsiz ve benzersiz yüz algoritmasını kullanarak kişileri tanımakta olan sistemlerdir. Peki, eşsiz yüz algoritması nedir? yüz tanıma güvenlik sistemleri, kişilerin alın – çene ve burun kısımlarında yer alan izleri kullanarak tanıma yaparlar. Bu alanlarda yer alan izler, her insanın yüzünde farklıdır. Bu da bu yüz tanıma güvenlik sistemlerinin her zaman için çok daha güvenilir bir hal almasını sağlamaktadır.

Yüz Tanıma

Başlangıçta bir bilgisayar uygulaması biçimi olsa da, son zamanlarda mobil platformlarda ve robotik gibi diğer teknoloji türlerinde daha geniş kullanım gördü. Genellikle güvenlik sistemlerinde erişim kontrolü olarak kullanılır ve parmak izi veya göz irisi tanıma sistemleri gibi diğer biyometriklerle karşılaştırılabilir. Biyometrik bir teknoloji olarak yüz tanıma sisteminin doğruluğu iris tanıma ve parmak izi tanımadan daha düşük olmasına rağmen, temassız ve noninvaziv süreci nedeniyle yaygın olarak benimsenmiştir. Son zamanlarda, ticari bir tanımlama ve pazarlama aracı olarak da popüler hale geldi. Diğer uygulamalar, diğerleri arasında gelişmiş insanbilgisayar etkileşimi, video gözetimi, görüntülerin otomatik indekslenmesi ve video veri tabanını içerir.

Yüz tanıma, Biyometri kişilerin fiziksel ve davranışsal özelliklerini göz önüne alarak yürütülen kimlik belirleme çalışmaları olarak tanımlanır. Bu kişi özelliklerine örnek olarak, retina ve iris görüntüleri, el geometrisi, Ayak izileri ve parmak izleri, elin damar görüntüsü, konuşma sesi, yüz özellikleri, kulak memesi, yürüyüş, klavye kullanım şekli, konuşma devinimleri örüntüleri verilebilir. Daha genel anlamda, biyometri bilişim teknolojilerinin bir parçası olup, insanların fiziksel özelliklerini ve davranış biçimlerini irdeleyip kişiyi tanıma amaçlı ipuçları çıkaran bir bilim dalıdır. Bizim içinde çalıştığımız bu biyometrik in içinde yüz özelliklerine bağlı insan tanıma, diğer bir deyiş ile “yüz bulma ve yüz tanıma”dır.



PyCharm

Pycharm, python için oluşturulmuş IDE programıdır. Python piyasasında kullanılan en iyi IDE dir. Bu IDE ile çok hızlı ve kolay şekilde uygulama geliştirebilir.

PyCharm, en popüler Python IDE'lerinden biridir. Çapraz platform uygulaması olarak mevcut olan PyCharm, Linux, macOS ve Windows platformlarıyla uyumludur. Günümüzde en çok kullanılan python IDE'lerinden biridir. Hem Python 2 (2.7) hem de Python 3 (3.5 ve üstü) sürümleri için destek sağlar. Pycharm ilk kez 2010 yılının Şubat ayında halka yayınlandı.

Dlib

Dlib, çok çeşitli makine öğrenimi algoritmaları içerir. Tümü, temiz ve modern bir C ++ API aracılığıyla son derece modüler, uygulaması hızlı ve kullanımı kolay olacak şekilde tasarlanmıştır. Robotik, gömülü cihazlar, cep telefonları ve büyük yüksek performanslı bilgi işlem ortamları dahil çok çeşitli uygulamalarda kullanılır. Veya başka deyişle Dlib , C ++ programlama dili ile yazılmış genel amaçlı bir çapraz platform yazılım kitaplığıdır . Tasarımı, sözleşme ve bileşen tabanlı yazılım mühendisliği tasarımından gelen fikirlerden büyük ölçüde etkilenir . Bu nedenle, her şeyden önce, bir dizi bağımsız yazılım bileşenidir. Öyle açık kaynak yazılım bir altında yayınlanan Boost Software License . Geliştirme 2002'de başladığından beri, Dlib çok çeşitli araçları içerecek şekilde büyüdü. 2016 itibariyle, ağ oluşturma , iş parçacıkları , grafik kullanıcı arayüzleri , veri yapıları , doğrusal cebir , makine öğrenimi , görüntü işleme , veri madenciliği , XML ve metin ayrıştırma, sayısal optimizasyon , Bayes ağları ve diğer birçok görevle başa çıkmak için yazılım bileşenleri içerir . 12 Son yıllarda, gelişmelerin çoğu, geniş bir istatistiksel makine öğrenimi araçları seti oluşturmaya odaklandı ve 2009'da Dlib,Makine Öğrenimi Araştırmaları Dergisi . O zamandan beri çok çeşitli alanlarda kullanılmaktadır.

Python ile Yüz Tespiti Nasıl Yapılır

Bu yazımızda, Python ile yüz ve yüzde bulunan göz, burun ve ağız gibi özelliklerin tespiti nasıl yapılır, bunu öğreneceğiz. Bu yazıdan sonra, Snapchat ve Instagram'da gördüklerinize benzer şekilde efektler ve filtreler oluşturabileceksiniz.

Yüz tespiti (yüz algılama olarak da adlandırılır), dijital görüntülerde insan yüzlerini bulmak ve tanımlamak için kullanılan yapay zeka (AI) tabanlı bir bilgisayar teknolojisidir. Yüz algılama teknolojisi; sağlık, askeri, güvenlik alanlarında oldukça kullanılan bir teknolojidir. Yüz tanıma, temel bilgisayar görme tekniklerinden makine öğrenimindeki ilerlemelere, giderek daha karmaşıklaşan yapay sinir ağlarına ve ilgili teknolojilere kadar ilerlemiştir.

DLib nasıl çalışıyor

Yüzümüzün tanımlanabilecek çeşitli özellikleri vardır, bunlar göz, ağız, burun, dudak vb.dir. Bu özellikleri saptamak için **DLib** algoritmalarını kullandığımızda aslında her özelliği çevreleyen noktaların bir haritasını alırız. Peki bu DLib nedir? DLib, karmaşık gerçek dünya problemlerini çözmek için kullanılan ileri seviye bir makine öğrenmesi kütüphanesidir. Bu kütüphane C++ programlama dilini kullanarak yapılmıştır, C/C++, Python ve Java dillerinde de çalışmaktadır.

Bu harita 67 noktadan oluşmaktadır (yer işareti olarak da adlandırılmaktadır) ve aşağıdaki özellikleri tanımlayabilir:

Noktalar haritası

- Çene Noktaları: 0-16
- Sağ Kaş Noktaları: 17-21
- Sol Kaş Noktaları: 22-26
- Burun Noktaları: 27-35
- Sağ Göz Noktaları: 36-41
- Sol Göz Noktaları: 42-47
- Ağız Noktaları: 48-60
- Dudak Noktaları: 61-67



Gerekli Kütüphanelerin Yüklenmesi

***opencv-python**

***dlib**

Python ile yüz tespiti yapmak için öncelikle bu kütüphaneleri terminale şunları yazarak yüklememiz gerekmektedir:

```
pip install opencv-python,dlib
```

Bazı işletim sistemlerinde **dlib** kütüphanesini yüklerken bir takım hatalar alabilirsiniz. Bu hatalardan biri de **CMake** hatasıdır. Bunu halletmek için **dlib** kütüphanesini yüklemeyen önce **CMake** kütüphanesini yüklememiz gerekmektedir.

```
pip install cmake
```

Gerekli kütüphaneleri yükledikten sonra, bu kütüphaneleri projemize dahil edebiliriz.

OpenCV ile Yüz Tespiti

OpenCV genel amaçlı nesne tespiti için pek çok sınıflandırıcı sunmaktadır. Bu sınıflandırıcılardan en yaygın bilineni Paul Viola ve Michael Jones tarafından keşfedilen ve Viola-Jones yüz dedektörü olarak adlandırılan “haar-based cascade” sınıflandırıcıdır. OpenCV data klasörü altında bulunan bazı sınıflandırıcılar aşağıda verilmiştir:

Yüz tespit örneği için küçük bir Python uygulama tüm kodu aşağıda verilmiştir. Öncelikle gerekli kütüphaneler yüklenmiştir. Yüz tespit edilecek. jpg formatlı görüntü *cv2.imread* fonksiyonu ile okunmaktadır. *cv2.CascadeClassifier* fonksiyonu ile tabloda verilen yüz tespiti için kullanılacak olan XML dosya yolu belirtilmektedir. Görüntüdeki yüzleri bulmamızı sağlayan *detectMultiScale* fonksiyonu birkaç seçenek sunmaktadır. *minFeatureSize*, düşünülen minimum yüz boyutudur (20×20 piksel gibi). *searchScaleFactor* görüntüyü yeniden ölçekleme (rescale) miktarıdır. Kod içinde verilen 1.1 değeri, verilen görüntünün %10 oranında boyutunun azaltılması anlamına gelmektedir. *minNeighbors* parametresi bir yüzü dikdörtgen içine almak için ne kadar komşu sayısının gerekli olduğunu hesaplamaktadır ve genellikle 3 ya da 5 iyi bir seçimdir. *detectMultiScale* fonksiyonu tespit edilen yüzlerin (x,y,w,h) konum bilgisini tutmaktadır. Yüz tespitinin başarısını göstermek için *cv2.rectangle* fonksiyonu ile tutulan koordinat bilgilerini dikdörtgen kutu içerisine alınmaktadır. Bir görüntü OpenCV kullanarak yüklendiğinde BGR renk formatındadır. Bu sebeple *matplotlib* kullanarak renkli görüntüyü göstermek istediğimizde *cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)* ile görüntüyü RGB formatına çevirmemiz gerekmektedir.

OpenCV 3.3 ayrıca yüz tanıma için derin sinir ağıları (deep neural networks-dnn) modülü sunmaktadır. Bu modül Caff , TensorFlow ve PyTorch derin  ğrenme platformlarını desteklemektedir. Bu    pop ler platformdan  ğitilmi  modelleri y kleme ve kullanma imkanı sunan OpenCV ayrıca farklı sınıflandırma modelleri, nesne tespiti, cinsiyet ve ya  tespiti gibi pek  ok probleme   z m sunmaktadır.

Literat r Taraması

Y z tanıma teknolojisi 1965’lerde kullanılmaya ba lamı tır. O zamandan bu yana bu konuda  ok sayıda  e itli  alı malar ger ekle mi tir. Bu konuda literat r taraması yaptıktan sonra benim  alı mama benzer     alı maları  zeti a ağıdadır:

- Resul DA ı , Berna POLAT ve G rkan TUNA tarafından 2019 yılında “Derin  ğrenme ile Resim ve Videolarda Nesnelerin Tanınması ve Takibi” adlı  alı mayı ger ekle tirmi lerdir. G r nt  i leme y ntemleri kullanılarak durağan g r nt lerin analizleri ger ekle tirilebilir ve s z konusu g r nt lerden anlamlı bilgiler  ıkarılabilir. Tespit ve tanıma sonrasında takip edilecek olan nesnenin deėi ken bir ortam i inde bulunması zorla tırıcı unsurlardan birisidir. Bunun gibi zorla tırıcı unsurlarla ba a  ıkabilmek ve nesne takibini ba arıyla ger ekle tirebilmek i in farklı y ntemler geli tirilmi tir. Askeri uygulamalarda yaygın olarak kullanılan elektro-optik algılayıcı sistemleri hareketli ve sabit hedeflerin belirlenmesini saėlamaktadır. Son yıllarda yapay zek  tabanlı bile enlerle g  lendirilen bu sistemler hem daha hızlı hem de daha kesin hedef tespiti yapmayı saėlamaktadır.  te yandan, derin  ğrenme algoritmaları yapay zek  alanında bir devrim yaratmı tır. Derin  ğrenme algoritmalarının g r nt  i lemede kullanılması olduk a ba arılı sonu lar alınmasını ve karma ık g r nt  i leme problemlerinin kolaylıkla   z me kavu turulabilmesini saėlamaktadır.

Bu  alı mada derin  ğrenme ile hareketli nesne tanıma ve takibi i in Google’ın a ık kaynak kodlu makine  ğrenmesi k t phanesi olan TensorFlow kullanılmı tır. Nesne takibi i in Region Based Convolutional Networks k t phanesinden Faster R-CNN modeli ele alınmı tır. Bu k t phaneler ile durağan g r nt ler, video g r nt leri ve webcam g r nt leri  zerinde nesne tanıma i lemi ger ekle tirilmi  ve incelenen k t phanelerin g  l  ve zayıf y nleri ortaya konmu tur

S. Sharma, Karthikeyan Shanmugasundaram ve Sathees Kumar Ramasamy tarafından 2016 yılın ’’ Dlib kullanan CNN tabanlı verimli y z tanıma tekniėi’’ adlı  alı mayı ger ekle tirmi lerdir. Y z tanımadaki ilerlemeye raėmen, son birkaç on yılda ara tırma alanında  ok daha fazla ilgi g rd  ve ticari pazarlarda bu proje, Dlib y zl  Convolutional Neural Network (CNN) kullanan Derin  ğrenmeye dayalı y z tanıma sistemi i in etkili bir

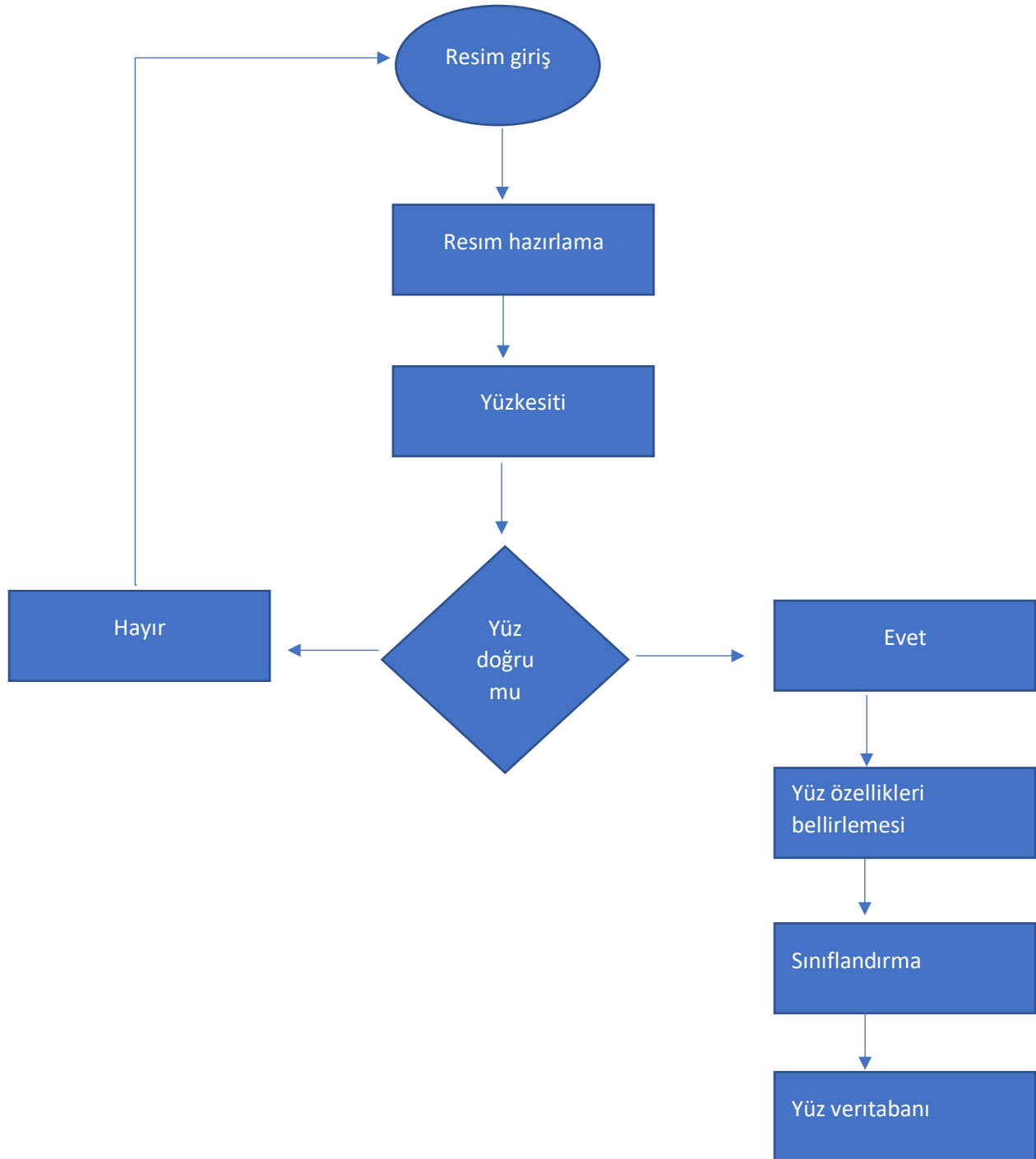
teknik sunuyor. hizalama. Kağıt, yüz hizalama ve özellik çıkarma gibi yüz tanıma ile ilgili süreci açıklamaktadır. Makale ayrıca yüz hizalamasının önemini vurgulamaktadır, bu nedenle doğruluk ve Yanlış Kabul Oranı (FAR) önerilen teknik kullanılarak gözlemlenmiştir. Hesaplamalı analiz, diğer son teknoloji yaklaşımlardan daha iyi performansı gösterir. Çalışma, Yüz Tanıma Büyük Mücadele (FRGC) veri kümesi üzerinde yapıldı ve 0,1 FAR ile % 96 doğruluk sağladı.

Nataliya Boyko, Oleg Basystiuk ve Nataliya Shakhovska tarafından 2018 yılında "" Dlib ve Opencv Kitaplığına Dayalı Yüz Tanıma"" adlı çalışmasını gerçekleştirmişlerdir. Yüz tanıma için bilgisayarla görme algoritmalarının zaman karmaşıklığına genel bakış ve araştırma. Ana makale fikri, iki popüler bilgisayar görme kitaplığını karşılaştırmaktır, bunlar OpenCV ve dlib'dir, özellikleri araştırır, her birinin artılarını ve eksilerini analiz eder ve her birinin hangi durumda en uygun olduğunu anlamaktır. Yöntem. Yüz tanıma için kullanılan bilgisayarla görme teknolojileri geliştirildi. İki popüler bilgisayar görme kütüphanesinin araştırması yapıldı. Özellikleri analiz edilir ve her birinin avantajları ve dezavantajları tahmin edilir. Yüz bulma için histogram odaklı gradyanlara, yüz oryantasyonu için yüz dönüm noktası tahminine ve bilinen yüzlerle karşılaştırmak için derin evrişimli sinir ağına dayalı bina tanıma uygulaması örnekleri. Makale yüz tanıma kavramını genelleştiriyor. Yüz tanımanın bilimsel temeli ve eksiksiz bir tanıma sisteminin yapısı açıklandı. Yüz tanıma programlarının temel ilkeleri formüle edilmiştir. Her iki kütüphanenin üretkenliğinin - yürütme zamanı ile uygulanan algoritmaların yinleme sayısı ile ilgili karşılaştırmalı bir analizi sunuldu. Ayrıca bu kitaplıkları temel alarak ve performanslarını karşılaştırarak yüz tanıma için iki basit uygulama oluşturdu.

YÜZ TANIMA ALGORİTMALARI

Yüz tanıma algoritmaları genel yapı itibarıyla ikiye ayırabiliriz. Bunlardan birincisi resimler üzerinden yapılan yüz tanıma tekniği, diğeri ise hareketli bir görüntü üzerinden yüz tanıma tekniğidir. Bu iki teknolojide günümüzde onlarca yerde kullanılmaktadır. Bunlara örnek olarak birçok ülkenin kullanıma bağıladığı pasaport kontrolündeki yüz taramaları ve mobese kameraları gibi ülke çapında kullanılan kameralarda Interpol'ün yapmış olduğu yüz tanımlama uygulaması gösterilebilir (Torun ve diğ., 2007). Pasaport kontrolündeki mantık kişinin yüzündeki biyometrik detaylar kiği ile eğiştirilerek bir sonraki pasaport geçişlerinde aynı kiği olup olmadığına dayanır. Bu yöntemde kişinin yüzünün biyometrik özellikleri tanımlanarak (örneğin burnun ağız ile arasındaki uzaklık, gözlerin birbiriyle olan uzaklığının ağzının genişliğine oranı gibi.) sisteme kaydedilir. Sonrasında geçişlerde sadece kameraya bakmak suretiyle eski ve yeni veriler karşılaştırılır. Mobese kameralarındaki Interpol uygulaması ise daha önceden veri tabanına kaydedilmemiş bir görüntünün anlık yakalanan görüntülerle karşılaştırılması mantığına dayanmaktadır. Böylelikle aranan kişinin görüntüsü veri tabanında tutulup diğerkontrol edilen kişilerin görüntüleri veri tabanında

tutulmamaktadır. Fakat bu yöntemin en büyük dezavantajı iğlenecek verinin çok fazla olmasından dolayı çok iyi donanıma sahip olunması gerekmektedir.



Yüz tanımlama algoritmalarında işleyiş adımları şöyledir;

- Web cam veya trafik kameraları gibi optik bir kaynaktan resim girişi,

- Resmi eğiterek yüzün tamamının alınmasındansa sadece kesitinin alınması için hazırlanması,
- Yüz kesitinin alınarak veri tabanındaki yüz kesiti ile karşılaştırılması,
- Uyumsuzluk durumunda resim giriğinden itibaren adımların tekrar edilmesi,
- Yüz kesitinin veri tabanındaki kayıtla eşleşmesi veya yeni bir kayıt alınması durumunda yüzün biyometrik özellikleri belirlenerek direk veri tabanına kayıt edilebilir veya bir sınıflandırmaya tabii tutularak karakteristik özelliklerine göre ayrıştırma yapılabilir.

Yüz tanımadaki üç temel unsur şöyle tanımlanır:

- *Kişi tanılama:* Bir nüfus kalabalık içinde bir kişinin kim olduğunu anlamak. -
- *Kişi onaylama:* Kişinin önceden kayıtlı kişi olup olmadığını, kayıtlı ise bu kaydın güncellenmesini sağlamak.

Yüz Tanıma Sisteminin Ana Yapıtaşları

Devşirilme ve kayıt

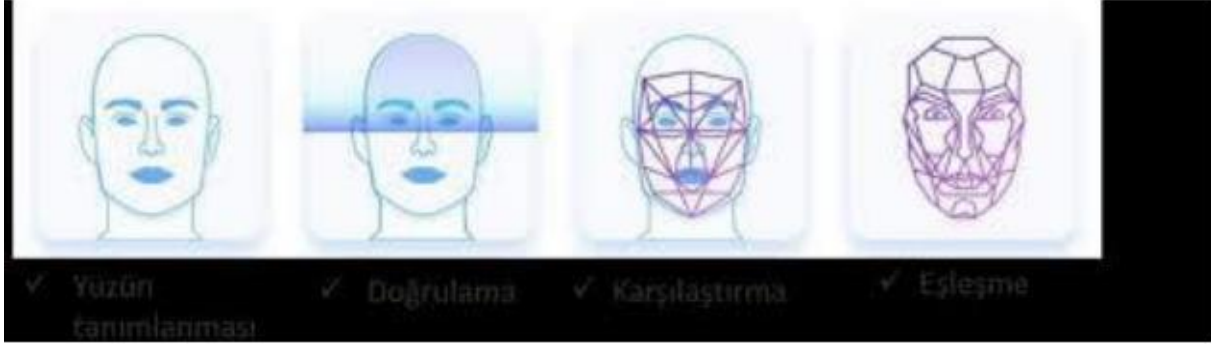
Kişiler, bir ya da birden fazla zaman diliminde bir ya da daha fazla fiziksel/davranışsal özelliklerini ölçtürerek sisteme kayıt olurlar.

Veri işleme

Ölçülen yüz özellikleri süzgeçlenir, gürültüden arıtmaya çalışılır, veriler istatistiksel olarak değerlendirilir, veri sınıflandırıcılar alıştırma döneminden geçirilir, birden fazla veri parçası veya karar mekanizması tümleştirilmeye çalışılır. Mesela, yüz imgelerini ele alalım. Yüz imgeleri, ısı kamera, derinlik kamerası, ışık kamerası, stereo kamera veya video kamerası ile elde edilir. Ham veriler, poz kestirimi, ışıklandırma kaynaklı etkilerin giderimi, farklı ölçekteki yüzlerin aynı boyda getirilmesi gibi ön işlemlere tabi tutulur. Bu şekilde düzelenmiş (normalized) yüzlerden öznitelik vektörleri çıkarılır, sadeleştirilir ve gürbüzleştirilir, sınıflandırıcılar eğitilir ve geçerlilik testleri tamamlanır. Bu aşamada her kişi bir yüz şablonu ile tanımlanmaktadır. Birden fazla veri gösterimi, birden fazla şablon ya da karar mekanizması kullanan sistemlerde tümleştirilmeden yararlanılır.

Bir terminalden oturum açmak isteyen ya da korumalı bir yerleşkeye girmeye çalışan bir kişinin o anki ölçümlerinden yola çıkılarak yeni bir şablonu oluşturulur ve bu test şablonu,

şablon kütüphanesinde daha önce elde edilmiş şablonlarla karşılaştırılarak tanımaya çalışılır. Geçerli bir kişinin sistem tarafından tanınmayıp reddedilmesi oldukça olumsuz bir deneyim yaratacağından tanıma doğruluğunun yüksek olması gerekir. Öte yandan yanlış kabul olasılığı yüksekse yetkisiz kullanıcıların ve sahtekarların giriş yapması kolaylaşır. Bu bakımdan yüz tanıma sisteminin güvenilirliği, hem yanlış ret olasılığının düşük, hem de yanlış kabul olasılığının düşük olmasını gerektirir. Bu durum “Eşit Hata Oranı” diye tanımlanır.



Avantajlar ve Dezavantajlar

Diğer biyometrik sistemlerle karşılaştırıldığında

Yüz tanıma ile birlikte, diğer biyometrik tekniklerle karşılaştırıldığında, yüz tanıma en güvenilir ve verimli olmayabilir. Yüz görüntülerinde büyük ölçüde farklılıklar olabileceği için yüz tanıma sistemlerinde kalite ölçüleri çok önemlidir. Yüz yakalama sırasında aydınlatma, ifade, poz ve gürültü gibi faktörler yüz tanıma sistemlerinin performansını etkileyebilir. Tüm biyometrik sistemler arasında, yüz tanıma, en yüksek yanlış kabul ve ret oranlarına sahiptir, bu nedenle, demiryolu ve havaalanı güvenliği durumlarında yüz tanıma yazılımının etkinliği hakkında sorular gündeme gelmiştir.

Zayıf Yönler

Carnegie Mellon Robotik Enstitüsü'nde 2008'de araştırmacı olan Ralph Gross, yüzün görüş açısıyla ilgili bir engeli şöyle anlatıyor: "Yüz tanıma, tam ön yüzlerde ve 20 derece uzakta oldukça iyi hale geliyor, ancak profil, sorunlar var. " Poz varyasyonlarının yanı sıra, düşük çözünürlüklü yüz görüntülerinin tanınması da çok zor oluyor. Bu, sürveyans sistemlerinde yüz tanımanın ana engellerinden biridir. Yüz ifadeleri değişiklik gösteriyorsa yüz tanıma daha az etkilidir. Kocaman bir gülümseme, sistemi daha az etkili hale getirebilir. Şirketlerde biyometri verilerinin depolanması söz konusu olduğunda veri gizliliği ana husustur. Yüz veya biyometri ile ilgili veri depolarına, uygun şekilde depolanmadığı veya saldırıya uğramadığı takdirde

üçüncü taraflarca erişilebilir. Techworld'de Parris, (2017) ekliyor: "Bilgisayar korsanları yüz tanıma sistemlerini kandırmak için zaten insanların yüzlerini kopyalamaya çalışıyor olacaklar, ancak bu teknolojinin hacklenmesi geçmişte parmak izi veya ses tanıma teknolojilerinden daha zor oldu."

İnsan Yüzü Tanımada Daha Önceden Yapılmış Çalışmalar

İnsan yüzü tanıma konusunda değişik yaklaşımlarla, değişik çalışmalar yapılmıştır. Bu yaklaşımların her birinin önemli avantajları olmuştur. Fakat şu ana kadar mükemmel bir metot geliştirilememiştir.

1-İnsanların Kullandıkları Yüz Karakteristikleri

İnsanlar birbirlerini genellikle yüzlerinden tanırlar. Bazen bir insanı arkasından, tanıdığımız birisine benzetebiliriz. Fakat sadece yüzüne bakarak tahminimizin doğru olup olmadığını anlayabiliriz. Ressamlarda genellikle insan yüzü portreleri yaparlar. 1973 yılında bir Bilimsel Amerikan yayınında bu konuda ilginç bir yazı yayınlanmıştır. Bir ressam gösterilen önden çekilmiş bir fotoğraf ışığında, ressam kişinin yüz özelliklerini bir kağıda tarif etmiştir. Bu tarifler başka bir ressamı gösterilerek onun fotoğraftaki kişinin resmini yapması istenmiştir. Elde edilen sonuç, bazı özelliklerin aynen, bazılarının ise belirsiz olarak tarif edilmesidir.

2-Template Matching

Template matching, bilgisayar yardımı ile yüz tanıma işlemine bir yaklaşım olarak, tüm resmin grilik seviyesi kalıplarını kullanmaya dayanan çok basit bir tanıma metodudur. Resim template matching yönteminin en basit versiyonunda koyuluk değerlerini içeren bir matris olarak gösterilir ve bütün yüzü belirten tek bir kalıpla karşılaştırılır. Elbette template matching yöntemini uygulayabileceğimiz daha akıllıca yollar vardır. Mesela, grilik seviyelerini içeren bir matris farklı bakış yönlerinden tanıma yapmak için kullanılabilir. Diğer önemli bir metot ise ,tek bir bakış yönü için çoklu kalıplar kullanmaktır.

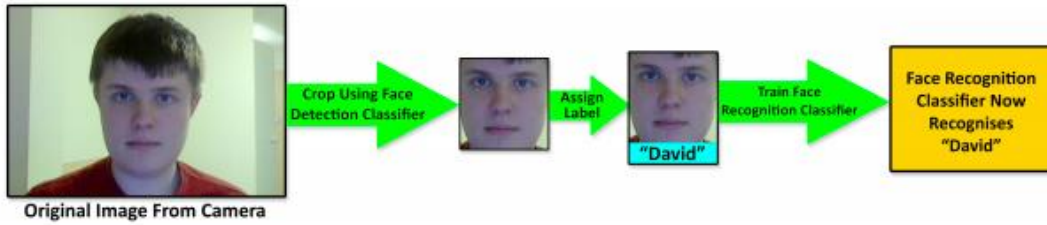
3-Özellik Tabanlı Denkleştirme

Yüz tanıma işlemi,yüz kısımları detaylar halinde çıkarılarak kaba bir şekilde yapılabilir. Ayırım için yüz kısımlarının kapsam-geometrik konfigürasyonu yeterlidir. Kapsamlı konfigürasyon yüzün ana kısımlarının boyutunu ve yerini belirten bir nümerik data vektörü tarafından belirtilebilir. Elde edilmiş yüz kısımlarının,yüzün dönmesinden,ölçüsünden ve yerinden bağımsız olabilmesi için,uygun bir şekilde normalize edilmiş olması lazımdır. Yüz tanıma zor olmasına rağmen, sunduğu ilginç avantajları yüz kısımlarının iyileştirilmesinde kullanıyoruz. İlk önemli avantajı iki taraflı simetridir. Diğer kolaylıklar için ise, her yüzün

benzer bir şekilde iki göz ,bir burun ve bir ağızdan oluşması gerçeğinden çıkartılabilir. Bu durum sınıflandırma işlemini zorlaştırmasına rağmen yüz kısımlarının çıkartılması işlemini kolaylaştırır. Yüz kısımlarını çıkartılması için en kullanışlı yöntem integral izdüşümüdür. Elimizdeki resim $T(X,Y)$ matrisi olarak ifade edilir

Yüz tanıma nasıl çalışır

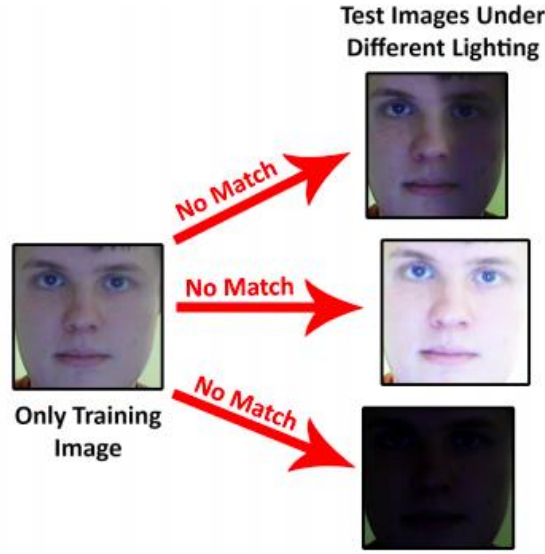
Yüzünden belirli bir kişinin kimliğini doğrulama süreci iki aşamaya ayrılmıştır. İlk aşama, bir görüntüdeki herhangi bir yüzü algılamak için bir sınıflandırıcı yetiştirmektir. Bu, sınıflandırıcıya birçok yüz görüntüsünü (pozitif görüntüler) ve yüzler dışındaki herhangi bir şeyin (negatif görüntüler) birçok görüntüsünü geçirerek elde edilir. Bu görüntülerin her birinden, sınıflandırıcının yüzü neyin oluşturduğuna ilişkin kararını iyileştirmek ve geliştirmek için kullanılan özellikler çıkarılır. Bu ilk aşama, kameradan alınan görüntü (hem eğitim hem de doğrulama için) aşağıdakilerden başka veriler içereceğinden, görüntülerde yüzün bulunduğu alanı seçmek için insan müdahalesi olmadan eğitim ve doğrulamanın gerçekleştirilmesine izin verdiği için gereklidir. görmezden gelinmesi gereken kişinin yüzü. İkinci aşama, belirli bir yüzü tanımlamak için ikinci bir sınıflandırıcı yetiştirmektir. Bu, sistemin tanımasını istediğiniz kişinin bir görüntüsünü alarak, yüz dışındaki tüm verileri kaldırmak için bu görüntüyü kırparak (yüz algılama sınıflandırıcısını kullanarak) ve bu kırpılmış görüntüyü (yeni yüzü temsil eden bir etiketle) geçirerek elde edilir. yüz tanıma sınıflandırıcısına



- bu süreç şekil 1.1'de gösterilmektedir⁶. Yüz tanıma sınıflandırıcı, verilen görüntüden özellikleri çıkaracak ve bunları verilen etiketle ilişkilendirecektir. Yüz tanıma sınıflandırıcısından bir görüntüdeki bir kişinin kimliğini tahmin etmesi istendiğinde, görüntü, yüz algılama sınıflandırıcısı kullanılarak kırpılır ve özellikler çıkarılır. Çıkarılan bu özellikler daha sonra sınıflandırıcıda depolanan yüzlerin özellikleriyle karşılaştırılır ve en benzer özelliklere sahip yüzün etiketi, sınıflandırıcının ne kadar 'kendinden emin' olduğunu belirten bir puanla birlikte döndürülür (bu, özellikleri). Yüz tanıma sınıflandırıcısının doğruluğunu artırmak için, sınıflandırıcıyı ideal olarak farklı konumlarda veya koşullarda aynı yüzün (aynı yüz olarak etiketlenmiş) birçok farklı görüntüsüyle eğitmek gerekir. Bu, sınıflandırıcının kişinin yüzüne ilişkin açıklamasını hassaslaştıracaktır çünkü aralarındaki benzerlikleri analiz etmek için birçok eğitim resmine sahip olacaktır.

Yüz Tanıma ile İlgili Sorunlar

Bu yüz tanıma çözümüyle ilgili bir sorun, sınıflandırıcıyı şekil 1.2'de gösterildiği gibi eğitirken kullanılan farklı olan, aydınlatmadaki zayıf performansdır. Bu sorunu gidermek için sınıflandırıcı, önemli ölçüde farklı aydınlatma koşulları altında kişinin birçok görüntüsüyle eğitilmelidir. Bu, test görüntüsündeki ışıklandırma farklıysa, benzer ışıklandırma altında bir görüntü ile eğitilmiş olacağından sınıflandırıcı tarafından doğru kişinin tahmin edilme olasılığının daha yüksek olduğu anlamına gelir. Bu yüz tanıma çözümüyle ilgili bir diğer sorun, gerçek bir kişi ile kamera önünde tutulan kişinin bir fotoğrafını birbirinden ayırt edememesidir. Sistem bu kadar kolay kandırılabilirse güvenli olmadığı için bu açıkça büyük bir güvenlik sorunudur. Sistem geliştirilirken bu sorunların ikisi de hesaba katılmış ve etkilerini azaltacak çözümler tasarlanmıştır.



OPEN-CV

OpenCV (Open Source Computer Vision Library, anlamı Açık Kaynak Bilgisayar Görüşü Kütüphanesi) gerçek-zamanlı bilgisayar görüşü uygulamalarında kullanılan açık kaynaklı kütüphane. İlk olarak Intel tarafından geliştirilmiş, daha sonra Willow Garage ve sonra Itseez (Intel tarafından satın alındı) tarafından sürdürüldü. Bu kütüphane çoklu platform ve BSD lisansı altında açık kaynaklı bir yazılımdır. C ++, Python, Java ve MATLAB arayüzlerine sahiptir ve Windows, Linux, Android ve Mac OS'yi destekler. OpenCV çoğunlukla gerçek zamanlı görüş uygulamalarına yönelir ve mevcut olduğunda MMX ve SSE talimatlarından yararlanır. Tam özellikli bir CUDA ve OpenCL arayüzleri şu anda aktif olarak geliştirilmektedir. 500'den fazla algoritma ve bu algoritmaları oluşturan veya destekleyen yaklaşık 10 kat fazla işlev vardır. OpenCV, C ++ ile yerel olarak yazılmıştır ve STL kapsayıcılarıyla sorunsuz bir şekilde çalışan şablonlu bir arayüze sahiptir.

Uygulama alanları

OpenCV'nin uygulama alanlarından bazıları şunlardır:

- Egomotion kestirimi
- Yüz tanıma sistemi
- Hareket tanıma
- İnsan-bilgisayar etkileşimi
- Gezgin robotlar
- Nesne tanıma
- Resim segmentleme
- Stereopsis görü
- Hareket takibi (motion tracking)
- Artırılmış gerçeklik



Yukarıdaki alanlardan bazılarını destekleyebilmek için, OpenCV bünyesinde bir makine öğrenimi kütüphanesi bulundurulur.

OPENCV'DE YÜZ TANIMA

gerçek zamanlı bilgisayar görüşü için programlama işlevleri kütüphanesidir. Projenin yüz algılama kısmı, Scala için bir OpenCV Kitaplığı kullanılarak yapıldı. Bunun nedeni, çoğu Yüz API'sinin yalnızca resimlerde algılama yapmakla sınırlandırılmış olması, oysa öğrenci katılımını kontrol etme sürecini hızlandırmak ve derslerden önce kuyrukları önlemek için projenin canlı bir video çekiminde yüz algılaması yapılması gerektiği idi. OpenCV kütüphanesi, bir yüzü gerçek zamanlı olarak doğru bir şekilde algılayabildiği ve yoldan geçen öğrencilerin yüzlerinin etrafına bir dikdörtgen çizerek onu vurgulayabildiği için proje için yeterince esnek olduğunu kanıtladı. Bunların hepsi yüz tanımadan ayrı bir pencerede gerçekleşir, böylece öğretim görevlisi, yüzleri algılanırken hem geçen öğrencileri hem de sistemin tanıma kısmından gelen geri bildirimleri takip edebilir. Yüzler tespit edilirken, uygulama her saniye canlı çekimin anlık görüntüsünü alır ve ardından bunu tanıma sistemine gönderir.

GUI(Graphical User Interface)- tkinter

Python, GUI (Grafik Kullanıcı Arayüzü) geliştirmek için birden fazla seçenek sunar. Tüm GUI yöntemlerinin dışında, tkinter en yaygın kullanılan yöntemdir. Python ile birlikte gönderilen Tk GUI araç setinin standart bir Python arayüzüdür. Tkinter ile Python, GUI uygulamalarını oluşturmanın en hızlı ve en kolay yoludur. Tkinter kullanarak bir GUI oluşturmak kolay bir iştir.

Window = tkinter.Tk () ile başlayıp Window.mainloop() biter.

tkinter gibi mevcut çeşitli popüler GUI kütüphanesi alternatifler vardır

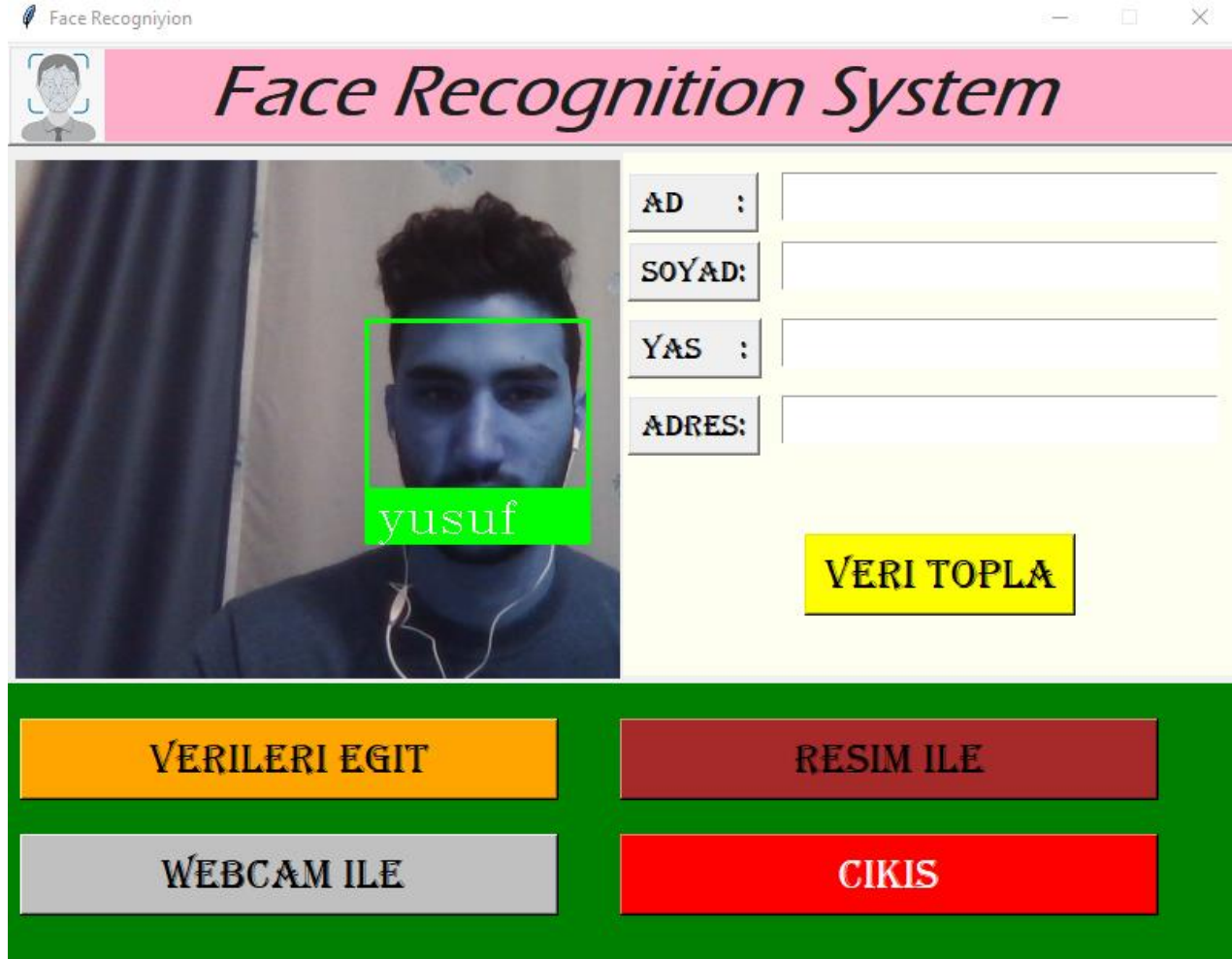
- wxPython
- PyQt,
- PySide
- pygame
- piglet
- PyGTK

Grafik kullanıcı arayüzünde bir uygulamayı oluşturan yapı taşlarından herhangi biri için genel terim. Esas olarak üç geometri yöneticisi sınıfı vardır bunlar:

- 1. pack () yöntemi:** Üst widget'a yerleştirmeden önce widget'ları bloklar halinde düzenler.
- 2. grid () yöntemi:** Ana widget'a yerleştirmeden önce widget'ları ızgarada (tablo benzeri yapı) düzenler.

3. place () yöntemi: Widget'ları programcı tarafından yönetilen belirli konumlara yerleştirerek düzenler.

Bu uygulama da arayüz olarak (tkinter) tercih edilmiştir ve tkinter kullanmak için (tkinter ve tkinter.messagebox) kullanılmıştır.



Şekil G.1.1

Uygulama

Veri Tolama

Yüz Tanıma Metodolojisi

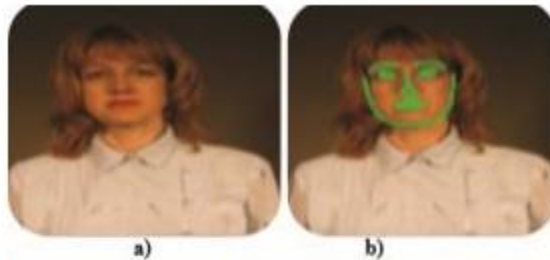
Gelen Tanım :Yüz algılama, yüz hizalama, yüz kırpma ve özellik çıkarma olmak üzere 4 sistemden oluşuyor

Yüz tanıma

Otomatik yüz tanımda ilk süreç, belirli bir nesne algılama türü olan yüz algılamadır. İnsan yüzlerinin doğru tespiti çok daha önemli bir süreçtir. Yüz algılama, insanın ön yüzlerini dijital görüntülerden veya videolardan algılayan bir bilgisayarla görme teknolojisidir. burun ve üst dudakların yüz işareti tespiti ile önden yüz algılama için bilgisayarla görme teknolojisini kullanır. Burun, yüzdeki işaretlerin merkez noktasıdır.

Dlib ile Yüz Hizalama

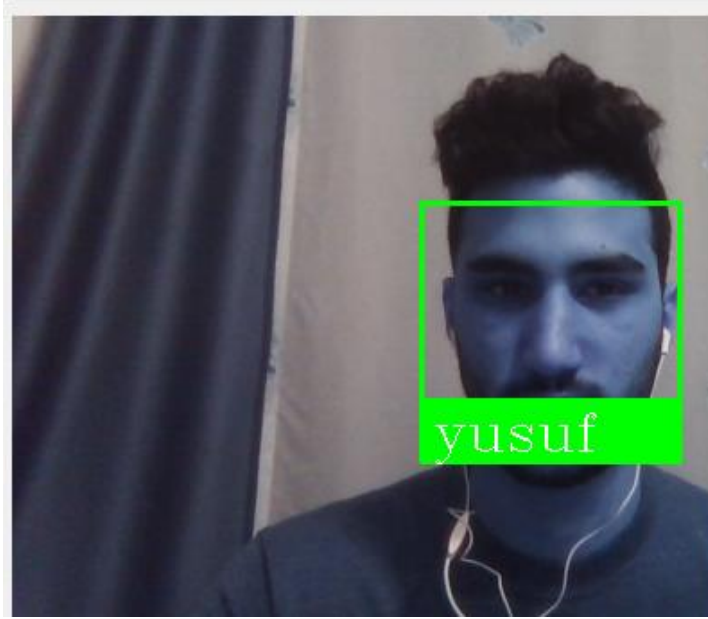
Dlib, C ++ 'da makine öğrenmesine dayalı yazılım geliştirmek için en iyi ortamı sağlayan açık kaynaklı bir kütüphanedir. Dlib'in özü, Temel Doğrusal Cebir Alt Programları (BLAS) ile 16 doğrusal cebirdir. Esas olarak Bayes ağlarının ve Kernel tabanlı algoritmaların sınıflandırma, kümeleme, anormallik algılama, regresyon ve özellik sıralaması için uygulanması için kullanılır. Dlib kitaplığının iki ana bileşeni vardır: doğrusal cebir ve makine öğrenimi aracı



CV kullanarak Yüz Algılama a) Giriş resmi b) Yüz hizalama

Veri Tabanına kayıt

Son adım olarak algılanan yüzün özellikleri ve kırıldıldıktan sonra veri tabanına kayıt

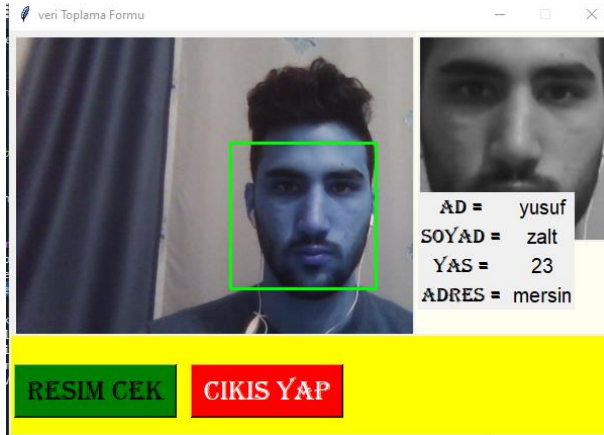


Şekil v.1.2

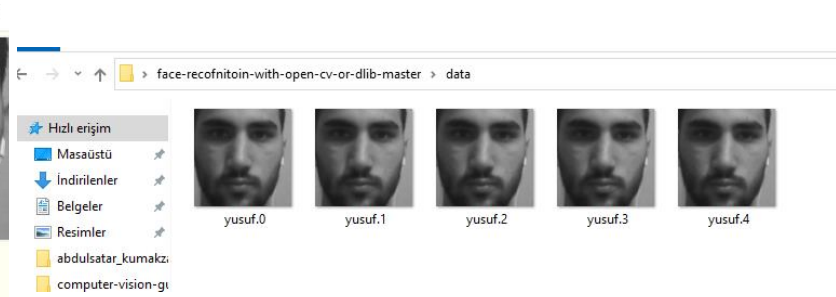
AD :	yusuf
SOYAD:	zalt
YAS :	23
ADRES:	mersin

VERİ TOPLA

Şekil v.1.1



Şekil v.1.3



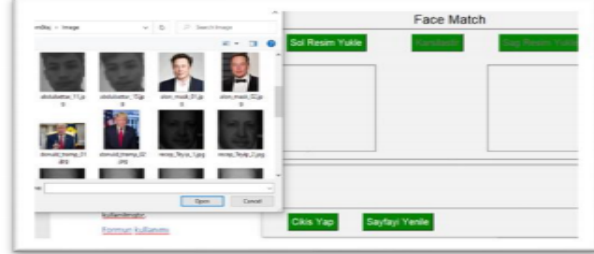
Şekil v.1.4

Şekil v.1.1 de ilk başta kişi bilgileri alınmış şekil v.1.2 ise veri toplama ara yüz formu şekil v.1.3 ise çekilen resimden bir kare ve kişi bilgileri son olarak şekil v.1.4 ise çekilen resimlerin görüntüleri

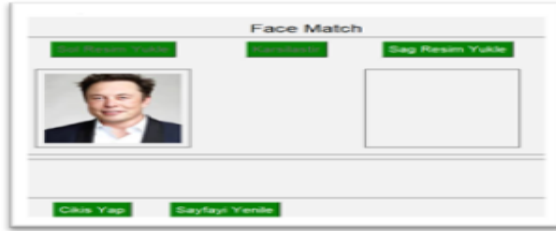
İki Resim Karşılaştırma



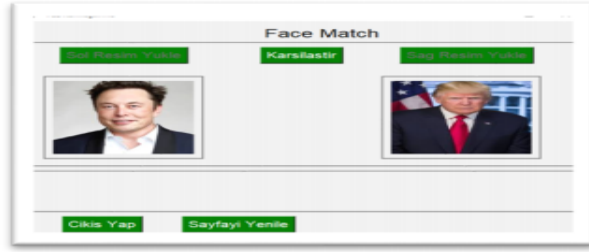
Şekil İ.1.1



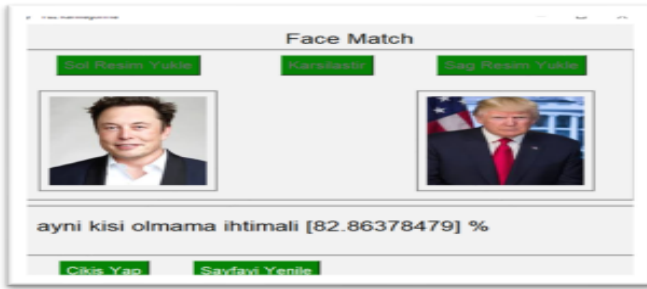
Şekil İ.1.2



Şekil İ.1.3



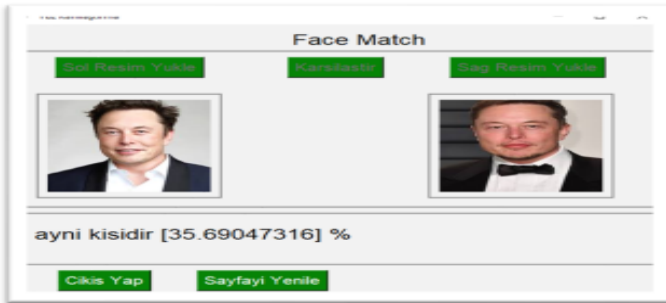
Şekil İ.1.4



Şekil İ.1.5



Şekil İ.1.6

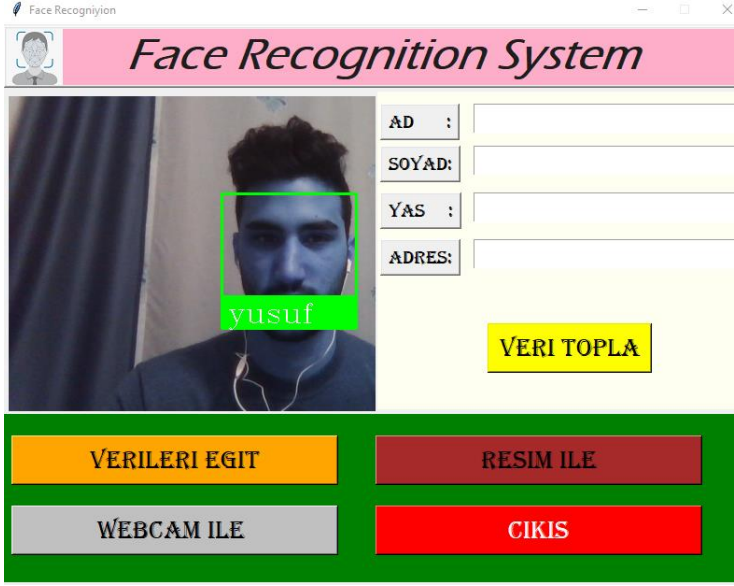


Şekil İ.1.7

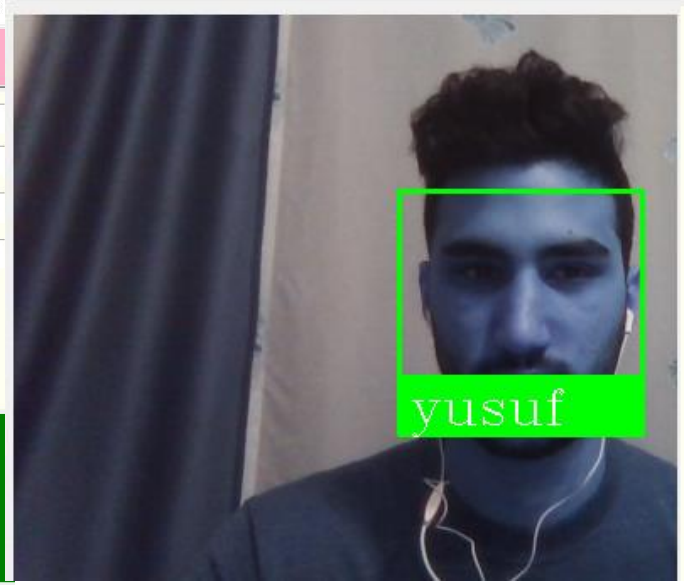
Şekil İ.1.1 yüz karşılaştırma ara yüz formu, Şekil İ.1.2 dosyadan resim yükleme, Şekil İ.1.3 karşılaştırılacak ilk resim, Şekil İ.1.4 karşılaştırılacak ikinci resim, Şekil İ.1.5 iki resim karşılaştırıp çıkan sonuç, Şekil İ.1.6 sayfayı yenileme ve son olarak iki aynı resmi karşılaştırma ve çıkan sonuç.

WebCam ile Yüz Tanıma

webcam dan gelen görüntüden yüz tanıma yapma, tanınan yüzü gray'a dönüştürme, gray resmin özellikleri alma son olarak veri tabanı ile karşılaştırıp resmin kime ait olduğunu yazdırma. Aynı veri toplamada olduğu gibi.



Şekil W.1.2



Şekil W.1.3

Şekil W.1.2 ana form arayüzü ve Şekil W.1.3 ise webcam penceresi canlı yüz tanıma penceresi.

MICROSOFT'UN FACE API İLE YÜZ TANIMA

Projenin tanıma kısmı Microsoft'un Face API kullanılarak yapılmıştır. Bunun nedeni, Microsoft'un API'sinin, listenin dışındaki bir yüzü tümüyle karşılaştırmak için kullanılabilen, içinde yalnızca bir yüzü olması gereken bir grup resmi temsil eden bir yüz listesi oluşturma, silme ve güncelleme yeteneği sunmasıdır. listedeki yüzler ve bir eşleşme bulun. Bu nedenle, bu yüz listeleri, böyle bir ortamda tam potansiyelleriyle kullanılabilir.

Model oluşturma

Bu projede, yüz tanımayı nasıl gerçekleştireceğimizi ve davetsiz misafirlerin girdiğinde bize kısa mesaj uyarıları gönderebilen basit bir güvenlik sistemi oluşturmaya öğreneceğiz.

Aşağıdaki alt bölümlerde ayrıntılı olarak açıklanmaktadır.

Kaynakça

1. Soeren Sonnenburg, 2009,” Dlib-ml: A Machine Learning Toolkit”,
“<https://www.jmlr.org/papers/volume10/king09a/king09a.pdf>”
- 2.wikipedia,”Yüztanım,sistemi,
”https://tr.wikipedia.org/w/index.php?title=Y%C3%BCz_tan%C4%B1ma_sistemi&action=edit§ion=1”
- 3.wikipedia,”Facial,recognition,system”,
https://en.wikipedia.org/wiki/Facial_recognition_system”
4. S. Sharma, Karthikeyan Shanmugasundaram ve Sathees Kumar, 2016, “FAREC — CNN based efficient face recognition technique using Dlib”,
“<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7831628>”
5. Xianghua Fan, Fuyou Zhang, Haixia Wang, Xiao Lu, 2012 “The system of face detection based on OpenCV”, “<https://ieeexplore.ieee.org/document/6242980>”
6. OpenCV, “<https://tr.wikipedia.org/wiki/OpenCV>”
7. Prof.Dr. Muhittin GÖKMEN, Dr. Binnur KURT, Fatih KAHRAMAN ve Abdulkerim ÇAPAR, 2007, “Çok Amaçlı Gürbüz Yüz Tanıma”,
<https://web.itu.edu.tr/~bkurt/Research/eeeag104e121/eeeag.104e121.final.report.pdf>”
8. Selçuk BAŞAK, İrfan GÜNEYDAŞ ve Arzu TUKAN, 1998, “İnsan Yüzü Tanıma”,
“<http://www.selcukbasak.com/download/YuzTanima-Proje.pdf>”
- 9.Ömer F. Yücel, PyCharm programı,2020, “<https://www.mertmekatronik.com/pycharm-nedir-ozellikleri.html>”