

**HAREKETLİ GÖRÜNTÜLERDEKİ İNSANLARIN YÜZ TANIMA METODUYLA
SINIFLANDIRILMASI**

SVETOSLAV MARTINOV ILIYANOV

BİLGİSAYAR MÜHENDİSLİĞİ Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Doç. Dr. Ercan BULUŞ

2022

T.C.
TEKİRDAĞ NAMIK KEMAL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



HAREKETLİ GÖRÜNTÜLERDEKİ İNSANLARIN YÜZ TANIMA METODUYLA
SINIFLANDIRILMASI

SVETOSLAV MARTINOV ILIYANOV

ORCID: 0000-0001-9136-0674

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
YÜKSEK LİSANS TEZİ
Danışman: Doç. Dr. Ercan BULUŞ

HAZİRAN-2022
Her hakkı saklıdır.

ÖZET

HAREKETLİ GÖRÜNTÜLERDEKİ İNSANLARIN YÜZ TANIMA METODUYLA SINIFLANDIRILMASI

Svetoslav MARTINOV ILIYANOV

Bilgisayar Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Doç. Dr. Ercan BULUŞ

Günümüzde insan odaklı işler yavaş yavaş teknolojinin gelişmesiyle birlikte yerini otomatik sistemlere bırakmaya başlamıştır. Bundan dolayı bilgisayarlara ve makinelere verilen görevlerin yerine getirilebilmesi için insanların yapılacak işten ziyade bilgisayarlara yaptıracığı iş gün geçtikçe önem kazanmaktadır. Bir sistemin otomatik çalışması için bir öğrenme mekanizmasının ve karar verme mekanizmasının ardı ardına yerine getirilmesi gerekmektedir. Özellikle siber güvenlik tarafında gelişen sistemler ile birlikte yüz tanımanın önemi de her geçen gün artmaktadır. Özellikle güvenlik güçleri tarafından kullanılan yüz tanıma sisteminin faaliyetleri dünya üzerinde her geçen gün artmaktadır. Bu noktada yüz tanıma sistemlerinin bazı gereksinimler ortaya çıkmaktadır. Bu gereksinimleri karşılamak yüz tanıma tekniklerinin gerçekleştirilebildiği yazılımlar kullanılır. Bu yazılımların temel amacı, yardımcı araçlar ile dijitale dönüştürülen görüntülere anlam kazandırmaktır. Bu çalışma hareketli görüntülerdeki yüzlerin tespiti yapıldıktan sonra, tespit edilen yüzü sınıflandırabilmek için yüz adlandırılarak yüze verilen isim ile yüzün görüntüsü bir klasöre kaydedilerek, insan yüzlerinin isim ile saklandığı bir klasör oluşturur. Bu işlem sonucunda insan yüzlerinden oluşan veri tabanlarına daha kolay veri toplanabilir veya yüz tanıma teknolojisi kullanan uygulamalar için kolay öğrenme yöntemi olabilir.

Anahtar Kelimeler: Yüz Tanıma, Yüz Tanıma Teknolojisi, Siber Güvenlik, Yüz Sınıflandırma, Yüz Tespiti

ABSTRACT

CLASSIFICATION OF PEOPLE IN MOTION IMAGES BY FACE RECOGNITION

Svetoslav MARTINOV ILIYANOV

Department of Computer Engineering

MSc. Thesis

Supervisor: Assoc. Prof. Dr. Ercan BULUŞ

Today, people-oriented jobs have started to give way to automatic systems with the development of technology. For this reason, in order to fulfill the tasks given to computers and machines, the work that people will do to computers rather than the work to be done is gaining importance day by day. In order for a system to work automatically, a learning mechanism and a decision-making mechanism must be followed in succession. The importance of face recognition is increasing day by day, especially with the systems developing on the cyber security side. The activities of the face recognition system, especially used by the security forces, are increasing day by day around the world. At this point, some requirements of face recognition systems arise. To meet these requirements, software that can perform face recognition techniques is used. The main purpose of these software is to give meaning to the images converted to digital with auxiliary tools. In this study, after detecting the faces in the moving images, in order to classify the detected face, the face is named and the image of the face is saved in a folder with the name given to the face and creates a folder where the human faces are stored with the name. As a result of this process, data can be collected more easily into databases consisting of human faces or it can be an easy learning method for applications using face recognition technology.

Keywords: Face Recognition, Face Recognition Technology, Cyber Security, Face Classification, Face Detection

İÇİNDEKİLER

ÖZET	i
ABSTRACT	ii
İÇİNDEKİLER	iii
ÇİZELGELER DİZİNİ	v
ŞEKİLLER DİZİNİ	vi
KISALTMALAR DİZİNİ	vii
TEŞEKKÜR	viii
1. GİRİŞ	1
1.1 Bilgisayar Görüşü	2
1.1.1 Bilgisayar Görüşü Tarihi.....	2
1.1.2 Bilgisayar Görüşü Nasıl Çalışır?	3
1.1.3 Bilgisayar Görüşü Örnekleri	4
1.1.4 Bilgisayar Görüşü Uygulamaları	5
1.2 Görüntü İşleme.....	6
1.2.1 Görüntü İşleme Kütüphaneleri.....	7
1.2.2 Görüntü İşleme İhtiyacı	7
1.2.3 Görüntü İşleme Adımları	8
1.3 Yüz Tanıma Uygulamaları.....	10
1.3.1 Yüz Tanıma Kullanım Alanları	11
1.3.2 Yüz Tanıma Sistemleri Örnekleri	13
1.3.3 Yüz Tanıma Sistemi Avantajları.....	14
1.3.4 Yüz Tanıma Sistemi Dezavantajları	15
1.4 Literatür Özeti	16
1.4.1 Eigenface.....	16
1.4.2 Sinir Ağları.....	18
1.4.3 Grafik Eşleştirme	20
1.4.4 Hidden Markov Model (HMM).....	21
1.4.5 Geometrik Özellik Karşılaştırması	22
1.4.6 Şablon Eşleştirme.....	24
1.4.7 3B Dönüştürülebilir Model	25
2. KULLANILAN YÖNTEMLER	26
2.1 Anaconda	26

2.2 Cuda	27
2.3 Dlib	29
2.4 OpenCV	30
2.5 Haarcascade	31
2.6 Face_recognition.....	34
3. GELİŞTİRİLEN PROGRAM	36
3.1 Akış Şeması	36
3.2 Kaynak Kod	41
4. SONUÇ VE ÖNERİLER.....	47
KAYNAKLAR	48
ÖZGEÇMİŞ	Hata! Yer işareti tanımlanmamış.



ÇİZELGELER DİZİNİ

Çizelge 1.4.1.1 Özyüzler performans değerleri (Kekül, H. vd., 2018).....	18
Çizelge 1.4.2.1 Farklı yüz tanıyıcıların, FERET veri tabanındaki 200 kişi üzerindeki performans değeri. (Lin, S. H. vd, 1997).....	19
Çizelge 1.4.2.2 Cinsiyet tahmini başarımlar yüzdeleri (Buluş, E., 2021)	20
Çizelge 1.4.7.1 CUDA programlama için desteklenen GPU ve Ekran kartı tablosu.	28



ŞEKİLLER DİZİNİ

Şekil 1.1.1 Bilgisayar Görüşü uygulamalarının yıllara göre sektörel ve pazar değerleri.....	5
Şekil 1.2.1 Görüntü İşleme Temeli.....	7
Şekil 1.2.2 Görüntü işleme kontrast değişimi örneği	9
Şekil 1.2.3 Görüntü işleme adımlarından görüntü iyileştirme örneği	10
Şekil 1.4.1 Yüzün çeşitli açılardan grafikleştirilmiş hali (Wiskott, L. ve Von Der Malsburg, C., 1996).....	21
Şekil 1.4.2 Yedi, tek boyutlu HMM modeli(Davari, P. ve Naimi Miar, H., 2008, January) ...	21
Şekil 1.4.3 Yüzün HMM model için yedi parçaya ayrılmış şekli.(Davari, P. ve Naimi, H. M. 2008, January)	22
Şekil 1.4.4 Yüz hatlarının geometrik olarak gösterimi.(Benitez-Garcia, G. , Nakamura, T. and Kaneko, M.,2017).....	23
Şekil 1.4.5 Yüzün belli bölgelerinin şablon eşleştirme yöntemiyle karşılaştırılması. (Brunelli, R. ve Poggio, T., 1993).....	24
Şekil 1.4.6 CMU-PIE veritabanından üç katmanlı yüz tamamlama aşaması. (Blanz, V. ve Vetter, T., 2003)	26
Şekil 2.2.1: Cuda Anahtar Kelimeler Kullanımı Örneği	27
Şekil 2.5.1 Haar Benzerlik özellikleri	32
Şekil 2.5.2 Haar özelliklerinin insan yüzüne uygulanması	33
Şekil 3.1.1 Programın akış şeması.....	37

KISALTMALAR DİZİNİ

OPENCV	Open Source Computer Vision Library
OCR	Optical Character Recognition
ICR	Intelligent Character Recognition
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Network
YSA	Yapay Sinir Ağları
DBNN	Desicion Based Neural Networks
PDBNN	Probabilistic Desicion Based Neural Networks
SOM	Self Organizing Networks
HMM	Hidden Markov Model
CUDA	Compute Unified Device Architecture
ALU	Arithmetic Logic Unit

TEŐEKKÜR

Bilgi Güvenliđinin görsel kısmı olan yüz tanıma tekniklerinden ve yüz tanıma üzerine yazdığım bu tezde karşılaştığım zorluklara karşı yardımlarını esirgemeyen Namık Kemal Üniversitesi, Çorlu Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Bölüm Başkanı Doç. Dr. Ercan BULUŐ' a teşekkürü borç bilirim.

Svetoslav Martinov ILIYANOV

Bilgisayar Mühendisi



1. GİRİŞ

Bu çalışmada, insan yüzünün tespit edilebilmesi için, görüntü işleme ve sınıflandırma yöntemleri kullanılarak çözüme ulaştırılmak için çalışmalar yapılmıştır. Bu çalışmalarda Python dili kullanılmış olup OpenCV kütüphanesinden ve face_recognition kütüphanesinden yararlanılmıştır.

İlerleyen teknoloji ile birlikte insanlar yaptıkları bütün çalışmalarda kendilerine kolaylıklar sağlaması amacıyla değişik yöntemler bulmuşlardır. İnsanlık tarihinde çığır açan bilgisayarın bulunuşu ile birlikte, çalışma hayatında kullanılan sarf malzemeler yerlerini yavaş yavaş dijital dünyanın lideri olan bilgisayarlara bırakmışlardır. Bilgisayarlar ilk başlarda hesaplamalar için kullanılmaya başlanmış daha sonrasında elektronik anlamda gelişen dünya ile beraber insanların vazgeçilmez birer parçası olmuştur. Hatta bazı sektörlerde insanların yerini almaya başlamışlardır. Üretim, ekonomi, savunma sanayii, güvenlik, eğitim gibi sektörlerin en büyük yardımcıları olan bilgisayarlar özellikle günümüzde üretim, güvenlik ve savunma sanayiinde büyük ataklar yapmıştır. İlk yıllarda daha çok veri saklama, veri girişi, veri takibi alanlarında kullanılmıştır. İnsan ile bilgisayar arasındaki arayüzler geliştikçe bilgisayar kullanımı daha da yaygınlaşmıştır.

İnsanların yüz özellikleri ile ayırt etme yolu, günümüzde yüz tanıma olarak bilinir. Yüz tanıma talebinin artmasının arkasındaki en önemli nedenler çeşitli güvenlik önlemleriyle alakalı olarak, erişim kontrolü, kredi kartı doğrulama, suçlu arama gibi olaylardır. Ayrıca çeşitli eğlence uygulamaları ve benzeri kullanım alanlarının da olduğu bilinir. Yüz tanıma uygulama alanları aşağıdaki gibi açıklanabilir.

- Face ID: Normalde kullanılan parola veya anahtar tabanlı tanılama yerine kişileri kimliklerine göre tanılama kullanır. Kimliklerinden kasıt olarak kişinin fiziksel olarak varlığının da önemi vardır. Bu tür uygulamalar genellikle devlet işlerinde, pasaport, ehliyet, kimlik çıkarmada kullanılır.

- Erişim Kontrolü (Access Kontrol) : Bir bilgiye, bir kaynağa veya bir yere erişim kontrolüdür. Bu kontrolde özellikle yüz tanıma kullanılmaktadır. Ülke sınırları, bazı tesislerin erişimi, çevirim içi erişimler, veri tabanı erişimleri, sınavlar bu Alana örneklerden bazılarıdır.

- Güvenlik: yüz tanıma sisteminin en geniş anlamda kullanıldığı alandır. Kimlik, uçağa biniş, suçlu tarama, stadyum, seyirci tarama, suçlu arama gibi genel anlamda her yerde güvenliğin sağlanması için kullanılır.

- Gözetim: Sokaklarda, park alanlarında, apartmanlarda, santrallerde kısacası halkın bulunabildiği her yerde yüz tanıma uygulamaları kullanılmaktadır.
- Emniyet Birimleri: Şüpheli arama, suçlu arama, şüpheli tarama, şüpheli arka planları, kontrol etme, dolandırıcılık, soruşturma alanları da en önemli kullanım alanlarıdır.
- Bilgisayar – İnsan haberleşmesi: İnsan mimiklerine göre bilgisayar kontrolü, oyunlar, eski fotoğrafları doğrulama düzeltme gibi kullanım alanları da vardır.

Yüz tanıma doğru biçimde yapıp garanti sonuçlar elde edildiğinde sayısız pratik uygulama geliştirilebilir.

1.1 Bilgisayar Görüşü

(<https://www.ibm.com/>)Bilgisayar görüşü bir yapay zeka alanı olarak değerlendirilebilir. Dijital ortamda bulunan görüntüler, resimler, videolar ve görsel bilgilerin anlamlandırılmasını sağlamak bilgisayar görüşü konusunun en önemli hedeflerinden biridir. İnsanın duyu organlarının olması gibi yapay zekanın da duyu organları bilgisayar ve onunla yapılacak olan uygulamalardır. Bilgisayar görüşü ise bu konuda en önemli özelliğe yani görme olayına öncülük eden bir teknolojidir.

Bilgisayar görüşünün ana fikri insan görüşü ile aynı temellere dayanır. Nesne ayırt etme, nesnelere nasıl ayırt edileceği, görüntünün doğru veya yanlışlık kavramı, uzaklık, büyüklük, renk gibi nesneye veya varlığa ait özelliklerin insan görüşü tarafından ayırt edilmesi insana doğuştan gelen bir özelliktir. İnsanın doğuştan gelen bu yeteneklerini aktarmak için makineler, bilgisayarla görme becerileriyle eğitilir. İnsan gözünün elemanları yerine algoritmalar, kameralar, sensörler çalıştırılır. Bilgisayarların da hızlı çalışma ve karar yetenekleri kullanılarak bilgisayar görmesi ile birlikte üretim takibi, denetimi, üretim hızları, kalite takibi gibi birçok işte ve sektörde yapılan işleri ve işlemleri kolaylaştırmaktadır.

1.1.1 Bilgisayar Görüşü Tarihi

Mühendisler makineleri daha aktif ve verimli kullanabilmek, görsel verileri anlamlandırmak için çeşitli çalışmalar yapıp her geçen gün ilerleme kaydetmişlerdir. Bunların başlangıcı sayılabilecek deneylerden biri nörofizyologlar tarafından bir kedi üzerinde

yapılmıştır. Kediye gösterilen görüntülerde, kedinin tepkisini ilişkilendirmeye çalışmışlardır. Kedinin çizgilere ve sert kenarlara tepki verdiğini keşfetmişlerdir. Sonuç olarak görüntüleri anlamlandırmanın ilk adımının basit şekillerle başladığı düşünülmüştür (Demush, R., 2019).

Bilgisayarların görüntüleri dijital platformda işlemesi için gerekli teknolojiler 1960'larda geliştirilmeye başlanmıştır. 1963 yılında iki boyutlu görüntüleri 3 boyuta dönüştürülebilen bir gelişme sağlanmıştır. Yapay zeka fikrinin ortaya atılmasıyla bilgisayar görme probleminin çözümü için çaba sarf edilmeye başlanmıştır. 1974 yılında optik karakter tanıma teknolojisinin (OCR) tanıtımı ile beraber basılı metinlerin dijital dönüşümü sağlandı (IBM Cloud Education, 2022). Bununla beraber akıllı karakter tanıma (ICR) ile plaka tanıma, fatura girişi, mobil ödemeler gibi uygulamalarda sıklıkla kullanılmaya başlanmıştır.

Sinir bilimci David Marr, 1982'de yaptığı çalışmada, görüşün hiyerarşik çalıştığını düşünüp, bilgisayar görüşünün temel şekilleri algılayacağı türden algoritmalar geliştirmiştir. Bununla birlikte bilgisayar bilimcisi olan Kunihiko Fukushima basit kalıpları tanıyabilen Neocognitron adında evrimsel bir hücre ağı geliştirdi.

2000 yıllarında araştırma yönü nesne tanıma çözümlerine dönmüştür. Hemen ardından yüz tanıma uygulamaları gerçek zamanlı olarak uygulanmaya çalışılmıştır. Yine 2000'li yıllarda görsellerin dijitalleşmesi adına yapılan çalışmalar için standartlar oluşturulmuş, görsellere nasıl açıklama eklendiği ve nasıl etiketlendiği konusu belli bir kalıba bağlanmıştır. ImageNet veri seti kullanıma açıldığında evrimsel sinir ağlarını bu veri setini öğrenmek için eğittiler. 2012 yılında Toronto Üniversitesinden bir takım, evrimsel sinir ağları(CNN) tabanlı AlexNet isimli modeliyle, görüntü tanıma işlemindeki hata oranlarının düşmesinde önemli bir yere sahiptir (Demush, R., 2019).

1.1.2 Bilgisayar Görüşü Nasıl Çalışır?

Bilgisayar görüşü çalışabilmesi için yüksek miktarda bilgiye ihtiyaç duyar. Algoritması bilgisayara gelen dijital bilgiyi anlamlandırana dek çalışmak için tasarlanmıştır. Bir bilgisayarın belli bir nesneyi tanıyabilmesi için, nesnenin diğer nesnelere farkı, aynı nesnenin farklı açılardan çekilmiş fotoğrafları ve nesnenin farklı türleri gibi parametreler ile beslenmesi gerekir.

Bilgisayar tanınmasının başarılı olabilmesi için en iyi iki yöntem, derin öğrenme ve evrimsel sinir ağı (CNN) isimli makine öğrenmesi yöntemleridir. Makine öğrenmeleri,

bilgisayarlara verilen verileri, algoritma olarak tasarlanmış modellere uygulayarak bilgisayarların öğrenmesini sağlamak için kullanılan teknolojilerdir. Makine öğrenme teknolojisi görüntüler üzerinde kullanıldığında ise bu model yeterli miktarda veri ile beslendiğinde, sistem verileri analiz ederek görüntüleri birbirinden ayırt etmeyi öğrenecektir. Bu algoritmik modeller görüntüyü tanımak yerine, bilgisayarı öğretmek için tasarlanmışlardır.

CNN, Makine veya derin öğrenme modeline, görselleri piksellere bölüp etiketleyerek yardımcı olur. Sonrasında matematiksel fonksiyonları çalıştırabilmek için etiketler kullanılır ve görselin tahminde bulunur. Öncelikle bir dizi tahmin yürüterek tahminlerin doğruluğunu kontrol eder, sonrasında ise insana benzer yüzleri ayırmaya başlar. CNN görüntüleri tanıma kullanılırken, tekrarlayan sinir ağı (RNN) video görüntülerindeki karelerin birbiri arasında ki ilişkinin anlaşılmasına yardımcı olur.

1.1.3 Bilgisayar Görüşü Örnekleri

Günümüzde birçok firma bilgisayar görüşü uygulamalarına maddi yönden kaynak ayırabilecek, gerekli donanımı ve performanslı bilgisayarları kurmak için yeterli finansman sağlayabilecek kapasitede değildir. Şirketlerin bu açığını kapatmak üzere bazı firmalar bulut üzerinden kullanıcılara hazır modeller ve gerekli donanımı sağlamak üzere çalışmalar yapmaktadır. Bu tarz çözümlerle bilgisayar görüşü uygulamalarına kaynak bulmak kolaylaştığı için kullanım alanları da yaygınlaşmaya başlamıştır. Bilgisayar görüşü uygulamalarının görevleri aşağıdaki gibi sıralanabilir:

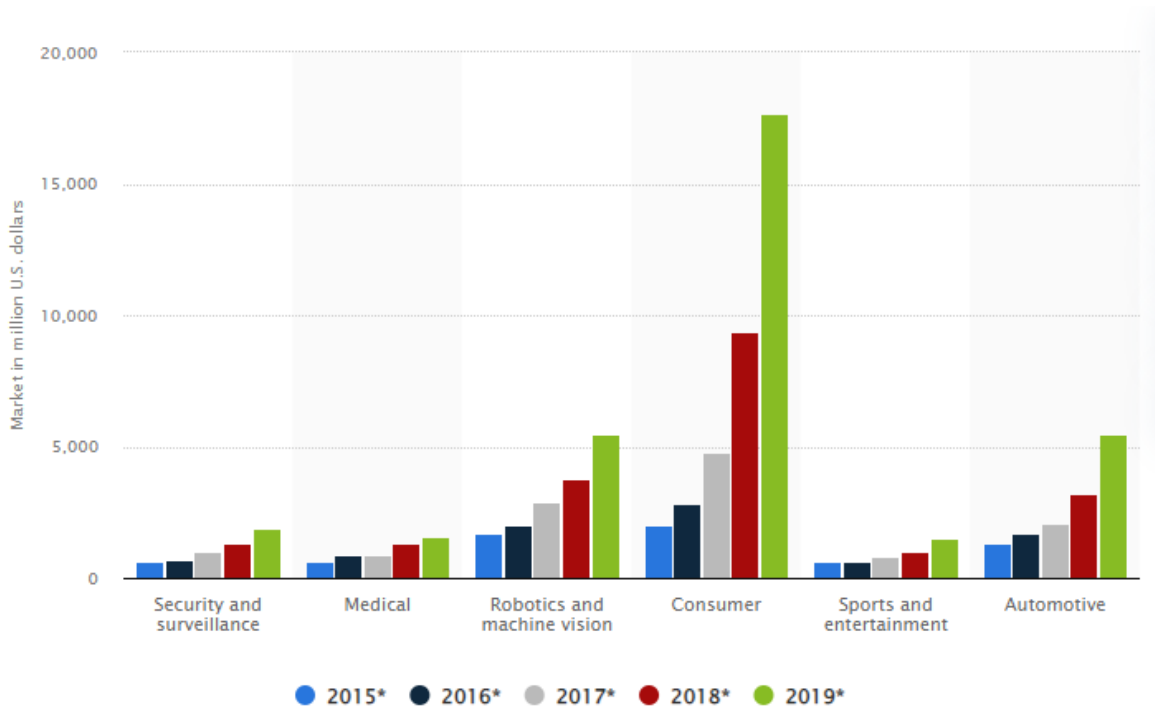
- Görüntü sınıflandırması bir görüntüyü anlamlandırabilmek için çabalar. Görüntünün neye ait olduğunu, hangi sınıfa girdiğini, ait olduğunu tahmin edebilir. Sosyal medya platformlarında kullanıcılar tarafından yüklenen fotoğrafların değerlendirilip yayına alınması işleminde bir görüntü sınıflandırma süzgeci çalışır. Sakıncalı fotoğraflar, düzgün fotoğraflardan ayrılarak sınıflandırma süzgecine takılırlar.

- Nesne algılama, görüntü sınıflandırmasının ardından görseli daha fazla anlamlandırabilmek için nesne algılama yöntemi iyi bir çözümdür. Örnek olarak aynı bantta yürüyen domateslerin ve patateslerin sayısının istendiği bir problemde nesne algılama ile bu sayıyı otomatik olarak saydırabiliriz.

- Nesne izleme, nesne algılandığından itibaren yapılabilecek aksiyonlar için nesne izleme şarttır. Bilgisayar görüşünün önemli adımlarından olan nesne izlemenin kullanıldığı en önemli alanlardan biri, otonom araçların içinde bulunduğu trafiğin akışını sürekli

denetlemesidir. Otonom araçlarda ki çarpışma önleyici sistemlerin tabanın da nesne izleme kullanılmaktadır. Yaklaşan ve uzaklaşan nesnelere sürekli takip edilerek, aracın tehlikeli sayılabilecek mesafelerde durmasını nesne izleme yeteneği tetiklemektedir.

- İçerik tabanlı görüntü alma, görsel verinin büyük olduğu alanlarda ki görüntülere göz atmak, görüntü arama, görüntü etiketleme bilgisayar görüşünü kullanmaktadır. Bu özelliğin en büyük örneği sosyal medya platformlarında, kişi etiketleme esnasında karşımıza çıkan “Bu siz misiniz?” sorunudur. İçerik tabanlı görüntü alma yeteneği bilgisayar görüşünü kullanarak veri tabanından gerekli yüz eşleştirmesini yaparak etiketleme işlemini otomatikleştirmektedir.



Şekil 1.1.1 Bilgisayar Görüşü uygulamalarının yıllara göre sektörel ve pazar değerleri.

1.1.4 Bilgisayar Görüşü Uygulamaları

Bu alanda yapılan çalışmaların tamamı, insanın gerçek hayatında yaptığı işlerin insan olmadan otomatikleştirme çabasıdır. Bunun sonucunda ulaşım, sağlık, eğlence, güvenlik sektörleri ve iş hayatında kullanılması insanların hayatını ciddi derecede kolaylaştırmaktadır. Trafik, mobese kameraları, akıllı telefonlar ve güvenlik sistemlerinden aktarılan görüntü bilgileri, bilgisayar görüşü uygulama alanlarının büyümesini mecbur kılmıştır.

IBM, 2018 yılında yapılan golf turnuvasında My Moments videosunun oluşturulması için bilgisayar görüşü kullanılmıştır. IBM Watson bütün bir golf turnuvasını izleyerek, en

önemli anlarını ve seslerini ayıkladı ve My Moments'i oluşturdu. Bir insanın bu iş için harcayacağı yüzlerce saat harcanmamış oldu.

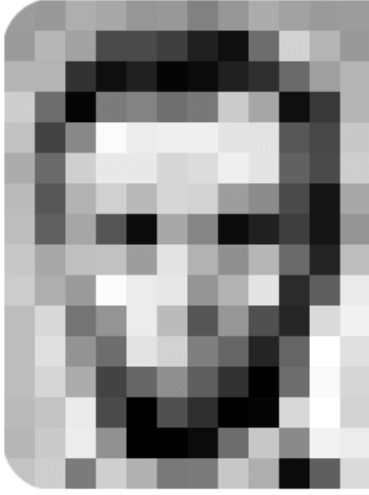
Google çeviri, akıllı telefonun kamerası ile başka dilde yazan bir eserin, yazının, tabelanın görselini, tercih ettiğiniz dile anında tercüme eden bir uygulama geliştirmiştir.

Otonom araçlar yola çıktığında tamamen bilgisayar görüşüyle hareket ediyor denebilir. Bilgisayar görüşü için kullanılan sensörler, kameralar diğer arabaları, yol işaretlerini, yol çizgilerini, tabelaları, yayaları, kısacası trafik akışında meydana gelen her görseli anlamlandırmak için sürekli çalışma halindedirler.

1.2 Görüntü İşleme

(Great Learning Team, 2011)Görüntünün dijital ortama aktarılması ve görüntüden faydalanmak için çeşitli işlemlerden geçirilmesini, görüntü işleme başlığı altında toplanabilir. Hızlı gelişen teknolojilerden biri olarak, günümüzde birçok sektörde çalışma alanı oluşmuştur. Mühendislik, grafiksel işlemler, bilgisayar görüntü işlemlerinde. Yapay zeka (AI) kullanımı gibi önemli alanlarda araştırmaların temelinde bulunur.

Görüntü işleme üç adım olarak değerlendirilebilir. Görüntüyü algılayabilmek için tarayıcılar, kameralar, sensörler gibi cihazlar kullanılarak görüntüyü içeri aktarma ilk adım olarak değerlendirilir. İkinci adım ise çeşitli cihazlardan aktarılan görüntülerin veriye dönüştürülüp, analize ve yönetime hazır hale getirilmesidir. Son adım ise görüntünün gelişimine ve analize dayalı olarak bir sonuca varılarak bir rapor oluşturulmasıdır. Bu raporlarla birlikte, görüntü işlemenin bir diğer yardımcı olduğu konu ise makine öğrenmesine sağladığı veridir. Görüntü işleme, görüntüye göre alınabilecek aksiyonların, öngörülerin tahmin edilebilmesi için kullanıcıya veya algoritmaları, veriler ile besleyen bir yöntem olarak da belirtilebilir.



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	5	10	33	48	105	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	105	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	85	150	79	38	218	241
190	224	147	108	227	210	127	102	35	101	255	224
190	214	173	66	103	143	95	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	95	218

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	5	10	33	48	105	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	105	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	85	150	79	38	218	241
190	224	147	108	227	210	127	102	35	101	255	224
190	214	173	66	103	143	95	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	95	218

Şekil 1.2.1 Görüntü İşleme Temeli

1.2.1 Görüntü İşleme Kütüphaneleri

Python' da görüntü işlemek için kullanılan kütüphaneler aşağıda listelenmiştir;

- Scikit-image
- OpenCV
- Mahotas
- SimpleITK
- SciPy
- Pillow
- Matplotlib

1.2.2 Görüntü İşleme İhtiyacı

Görüntü işleme amacı, görüntü üzerinde değişiklikler yaparak görüntüyü iyi veya kötü yönde manipüle etmektir. Görüntüyü istenen şekle sokmak, çeşitli programlar ile videoyu veya görüntüyü estetik açıdan güzelleştirmek, günümüz popüleritesinde en çok kullanılan yöntemlerdir. Dijital görüntüleme cihazına alınan görüntüler ile dijitalleştirilmiş olan görüntüler de görüntü işlemeye örnek gösterilebilir. Dijital görüntü algılayıcılar dünyayı insan gözü gibi görmezler. Bundan dolayı görüntü işleme bilimsel olarak araştırılmalıdır. Görüntü işlemeye, yapılan işlemin kaydedilmesi, raporlanması ve görüntüye uygulanması dâhildir.

Görüntü işlemenin içeriğinde görüntüyü işleme olduğu gibi ses düzenleme de yer almaktadır. Daha gelişmiş bir görüntü elde etmek veya görüntüden önemli bir kesit almak, görüntülerden yeni görüntü oluşturmak, görüntüleri küçültmek, büyütme, daha anlaşılabilir bir hale getirmek gibi işlemler günümüz dünyasında insanların ihtiyacı olabilir. Bu ihtiyaçta görüntü işlemenin, mühendislik ve bilgisayar bilimleri içinde daha derin bir araştırma konusu olmasına yol açar.

Görüntü işleme 3 adımda anlatılabilir:

- Görüntü algılama araçlarıyla görüntünün içe aktarımı.
- Görüntüyü dönüştürme veya değiştirme.
- Raporlanabilir veya analiz edilebilir sonuç.

Toplanan veya üretilen sonuçlar genel olarak tamamlanmamış veriler olduğundan dolayı, doğrudan uygulamalarda kullanılması doğru olmayabilir. Bundan dolayı çıkan veri önce analiz edilip sonrasında kullanılması uygundur.

Bir örnek ile anlatılacak olursa yüz ayrımı yapmaya çalıştığımızda, görüntüyü algılayan cihazlarımız veri toplamaya başlar ve yüzü gördüğü gibi bize doğru yüz olup olmadığını bildirecektir. Bunu ayırt etmek için öncelik yüzün farklı açılardan görsellerini sisteme öğretmek ile olacaktır. Bu tarz öğrenme durumlarında genel sorun görüntülerin aynı boyutta olmaması olduğu için bütün görüntüleri aynı boyuta getirmek için görüntülerde işlem yapılması gerekir. Bu işlem bilgisayar görüşü uygulamaları için görüntü işlemenin önemini birkez daha ortaya çıkarır.

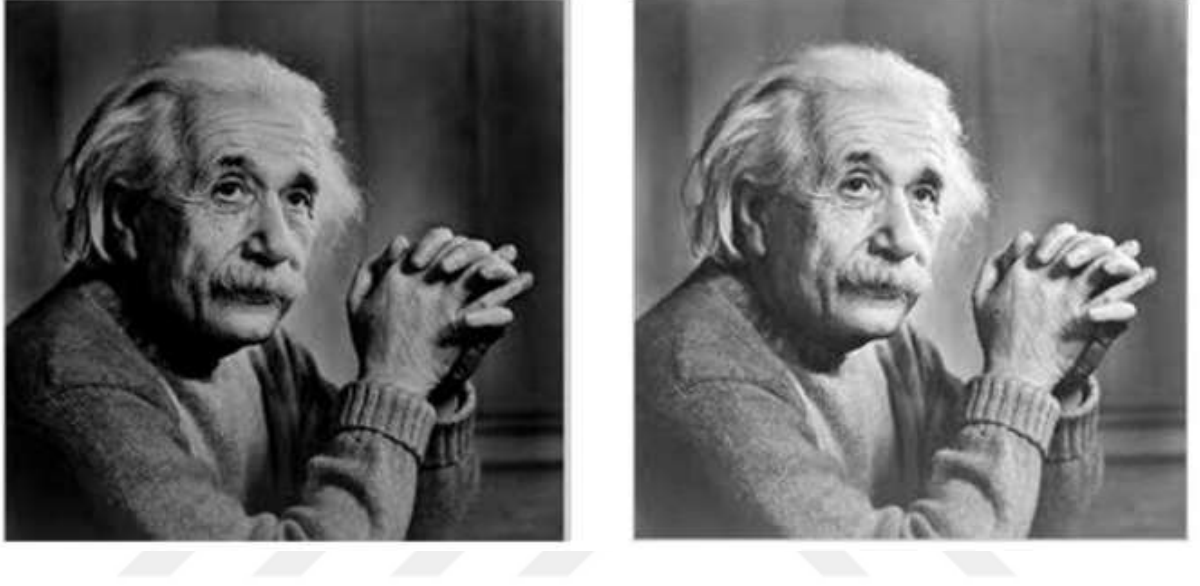
1.2.3 Görüntü İşleme Adımları

Görüntü Toplama: Görüntüyü dijitale aktaracak olan ilk adımdır. Bir nesnenin kendisinin veya durumunun dijital olarak algılanmasıdır. Kelime olarak görüntünün saklanması gibi anlamlar algılanabilir. Fakat genel anlamda görünen bir görüntünün dijitale döndürülmüş hali görüntü toplamanın açıklamasıdır.

Görüntü Geliştirme: Görüntü geliştirme adımı bir görüntünün kullanılabilmesi için yapılan geliştirme iyileştirme çalışmasıdır. Görüntü açık renkli ise koyu renkliye dönüştürme veya görüntünün hatlarını keskinleştirme, görüntünün analiz ve raporlama için uygun hale getirilmesi görüntü geliştirmeye örnektir.

Görüntü İyileştirme: Teknik anlamda gürültülü, kullanılmayacak, anlaşılmayan görüntüleri iyileştirme adıdır. Ses bozukluğu, kamera odaklaması, hareket bulanıklığı gibi anlamayı zorlaştıracak bozuklukları düzeltmek için görüntü iyileştirme teknikleri kullanılır.

Görüntü Renklendirme: Renklendirme genelde görüntüdeki malzemelerin daha iyi ayırt edilebilmesi için kullanılan bir tekniktir. Sınıflandırma işlemlerinde genellikle insanın anlayabilmesi için görüntü renklendirme teknikleri kullanılır.



Şekil 1.2.2 Görüntü işleme kontrast değişimi örneği

Görüntü Sıkıştırma: Dijital fotoğrafçılıkta kullanılan bu yöntem ile maliyetler azaltılabilir. Sıkıştırılan görüntü dosyaları daha az yer kaplayarak toplu şekilde işlenebilir ve saklama koşullarını ekonomik anlamda geliştirebilir.

Karakter Tanıma: Karakter tanıma OCR olarak bilinir. Basılı ve yazılı metinlerin dijital aktarımı probleminin çözümü olarak ortaya çıkmıştır. Optik okuma kısa süre içinde gelişerek karakter tanıma görsel işleme ve bunun yanında sesten metine dönüştürme gibi dallara ayrılmıştır.



Şekil 1.2.3 Görüntü işleme adımlarından görüntü iyileştirme örneği

1.3 Yüz Tanıma Uygulamaları

Yüz tanıma yöntemi bir insanın yüz özelliklerini kullanarak doğru insan olduğunu onaylamanın yöntemidir. Görsel bir yöntem olduğundan dolayı fotoğraflar, videolar veya gerçek zamanlı ortamlarda kameralar yardımıyla kullanılabilir. Yüz tanıma yöntemi, biyometrik ve geometrik olabilen bir güvenlik kategorisidir. Biyometrik kategori içine ses, parmak izi, göz retinası, iris tanıma gibi yöntemlerde girer. Farklı kullanım alanlarına ilgi artarak devam etmektedir, fakat genel olarak güvenlik ve kolluk kuvvetlerinin kullandığı teknolojilerden biridir (Kaspersky Lab, 2022).

İphone mobil telefonların FaceID özelliği yüz tanıma sistemine bir örnek olarak gösterilebilir, fakat çok basit bir türevidir. Sadece sahibinin yüzünü tanır ve kilidi açar. Yapılan işlem veri tabanı gibi bir ortamda çalışmaz, tek kişi ile sınırlıdır ve başka yüzler için kilit açılmaz. Bu teknoloji telefon kilidi kullanımının ötesinde, özellikle görüntü alınan yerlerde insanların yüzlerini karşılaştırarak çalışır. Kişi şüpheli olmasa bile kişinin yüzü internet üzerindeki kaynaklardan, sosyal medya üzerinden, bulut üzerindeki konumlar gibi kaynaklardan gelip herhangi bir lokasyon da tespit çalışması yapılabilir.

Yüz tanıma sistemleri değişebilir fakat genel anlamda aynı şekilde adımlar izlenir. Kamera kalabalık içinde veya tek başına bir yüzü görür ve belirler. Bu yüz yan görünüş veya karşıdan görünüş olabilir. Yüz yakalandığında görüntü analiz edilir. Büyük ölçüde yüz tanıma yöntemlerinde 2 boyutlu eşleştirme kullanılır. Yüzün geometrisini gören program, göz arasındaki mesafe, alın ve çene arasındaki mesafe, kaşlardaki renk farklılığı, dudakların konturu gibi özelliklere odaklanarak bir anahtar oluşturabilmektedir.

Yüz tespit edildikten sonra görüntünün yüz özelliklerini yansıtacak dijital bir veriye dönüştürülmesi gerekir. Yüz görüntüsü bir tür matematik formülüne dönüşerek sayısal veri olarak karşımıza çıkar. Parmak izi gibi yüz izi de eşsiz olmalıdır. Burada başarı oranını artırabilmek için matematik formülünün başarısı önemlidir. Eşsiz yüz diğer yüzlerin dijital verilerinden oluşan veri tabanındaki yüzler ile karşılaştırılır. Özellikle istihbarat birimlerinin milyonlarca kişinin yüz verisi tutulan veri tabanlarına erişimi vardır. Yüz verisi veri tabanında ki başka bir yüz ile eşleşiyorsa başarılı bir eşleşme yapılmış demektir.

Bütün tanıma sistemleri arasında en anlaşılabilir olanı yüzler olarak ortaya çıkar. Sağlama yapması kolaydır. Parmak izi de benzersiz olmasına rağmen dijital araçlar olmadan farklı olduğunu tespit etmek zordur. Yüzler, insan tarafından teyit edilmesi en kolay, dijital hataları algılayabilecek kontrol esnekliğini tanıma sistemlerine sunar. Dünyadaki nüfusun yarısından çoğunun yüz tanıma teknolojilerine maruz kaldığı tahmin ediliyor.

1.3.1 Yüz Tanıma Kullanım Alanları

Çeşitli amaçlarda kullanılan yüz tanıma uygulamalarının kullanım alanı da günlük hayatta epey kullanılmaktadır. İnsanların kullandığı özellikle kimlik kontrol sistemleri üzerinde geniş bir kullanım alanına sahiptir.

Günümüzde bütün verilerimizi sakladığımız telefonların kilidini açmak için neredeyse bütün telefon markaları artık yüz tanıma sistemleri kullanmaktadır. Telefonumuzun içinde birçok kişisel veri barındığından dolayı eşsiz bir koruma yöntemi olan yüz verisi ile korunması, telefonun kayıp olduğunda veya çalınması durumunda kişisel verilerin tam anlamıyla korunmasının yüksek oranda garantisini verebilmektedir. Apple firması telefonun yüz kilidinin rastgele açılmasının 1 milyon ihtimalde 1 olduğunu iddia etmiştir.

Yüz tanıma özellikle güvenlik kuvvetleri tarafından yüksek güvenlik gerektiren alanlarda veya şüpheli aramalarında çoğunlukla kullanılan bir teknolojidir. Özellikle devletin güvenlik güçleri suçlu yüzleri ile veri tabanlarını sürekli genişletmeye devam etmektedir. Bununla birlikte mobil yüz tanıma uygulamaları sahada çalışan güvenlik memurların işini kolaylaştırmaktadır. Bir sürücü veya yayanın fotoğrafı çekilerek veri tabanı üzerinden yüz eşleştirme ve tanıma işlemine hız kazandırmaktadır.

Sınır kapılarında ve hava alanlarında sürekli kullanılmaktadır. Önceden çok büyük iş gücü sarf edilen bir iş olarak görülen sınır memurluğu artık yerini akıllı otomatik kameralara bırakmaya başlamıştır. Uzun kuyruklar oluşan sınır kapılarında yüz tanıma sistemi kullanılarak bu sorunun önüne geçilmiş, artık yolcular pasaportlarını akıllı bir sisteme gösterip yüzlerini gösterdikten sonra sınır kapıları yolculara açılmaktadır. Burada ki işlemde yalnızca kuyruklar azaltılmamış aynı zamanda güvenlikte otomatik hale getirilmiştir. Bunun yanında güvenliğin gerektiği noktalarda, spor müsabakaları, olimpiyatlar, toplantılar gibi etkinliklerde de kullanılmaya başlanmıştır.

Kişilerin güvenliği kapsamında, kayıp kişiler, insan ticareti mağdurlarının bulunabilmesi için kullanılabilir. Kayıp olan kişinin yüzü veri tabanına eklendiğinde, yüz tespitinin yapıldığı konumdan güvenlik güçlerini uyarmak çözüme daha kolay ulaştıracaktır.

Daha önce toplumda suç işlemiş kişilerin veri tabanına eklenmesiyle suçluların gittiği noktaların tespit edilmesi, girdikleri alışveriş mağazaları veya potansiyel işlenecek suçların önüne geçebilmek için yüz tanıma sistemi kullanımı çok uygundur.

Alışveriş merkezlerinde özellikle perakende sektörde müşteri deneyimlerini iyileştirmek amacıyla yüz tanıma sistemiyle müşteri tanınabilir, müşteriye özel seçenekler kampanyalar sunulabilir bunun yanında yüz tanıma sistemiyle ödeme gerçekleştirilebilir. Bununla birlikte ödeme sırasında oluşan kuyrukların azalmasına da yardımcı olunmuş olur.

Bankacılık sektöründe mobil bankacılıkta güvenliğin sağlanması için aktif şekilde yüz tanıma sistemleri kullanılmaktadır. Normal parolalara göre daha güvenli olan yüz tanıma sistemleri parola yerine yüzün biyometrik özelliklerini kullanır. İleriki dönemlerde şifrelerin ve imzaların bankacılıkta ki yerini yüz tanıma sisteminin almasına olası gözle bakılmaktadır.

Reklamcılık alanında yüz tanıma sistemlerinin farklı şekilde kullanım amaçları da oluşmuştur. Bazı yiyecek firmaları yüz tanıma sistemiyle insanların duygusal tepkilerini

sınıflandırmışlar ve bu sınıflandırma sonuçlarını analiz ederek pazarlama kampanyası başlatılmıştır.

Sağlık hizmetlerinde yüz tanıma sistemlerinin test aşamaları hala sürmektedir. Hasta bakımı, hasta kayıtlarına ulaşabilmek için kullanılan yüz tanıma sistemleri aynı zamanda hastanın duygusunu ve acısını tespit etmek için de kullanılmaya başlanmıştır. Doktorların yazdığı reçetelerin artık yüz tanıma sistemleri ile çalışabileceği yazılımlar geliştirilmeye başlanmıştır.

Personel takip sistemleri yerine de yüz tanıma sistemleri kullanılmaya başlanmıştır. Bunun yanında personelin aktif çalışma süresi, giriş, çıkış bilgileri otomatik olarak yazılımda anlam kazanarak ücretlerinin doğru ödenmesine katkı sağlar. Personel örneği gibi eğitim sektöründe de kullanılan yüz tanıma sistemi öğrencilerin devamsızlık yapıp yapmadığını otomatik olarak tespit etmek için kullanılan bir teknolojidir.

Motorlu taşıtların çalışması için anahtar yerine yüz tanıma teknolojilerinin kullanılması testleri yapılmaya başlanmıştır. Özellikle arabalarda kullanılan bu özellik anahtar kullanımı yerine alternatif olarak görülür. Araç yüzü tanıdıktan sonra koltuk, direksiyon, dikiz aynası ayarlarını kişiye göre ayarlayan yardımcı bir sistem olarak kullanılır.

Kumar bağımlılıklarının izlenmesi için yüz tanıma sistemi kullanılabilir. Kumar şirketleri güvenliği sağlamak amacıyla ve müşterilerini daha yüksek derece güvenlik ile koruyabilmek için bu sistemi kullanır. Kumar oynanan alanlara giren kişileri personel tarafından takip etmek zordur. Kumar bağımlısı kişiler yüz tanıma sistemi veri tabanına eklenerek, çalışan personeli kişiye tavsiye vermek için yönlendirebilir. Kara listedeki birinin kumar oynamasını engellemek için de yardımcı bir sistemdir. Böylelikle kumarhaneleri para cezaları ödemekten kurtarabilecek bir sistemdir.

1.3.2 Yüz Tanıma Sistemleri Örnekleri

Apple, telefon kilitlerini hızlı açmak, uygulama içi alışveriş yapabilmek, uygulama yüklemek için yüz tanıma özelliğini kullanır.

British Airways yolcuların uçaklara pasaport veya bilet göstermeden binebilmeleri için yüz tanıma özelliğini aktif olarak kullanılmaktadır. Yolcuların doğru oldukları tespit

edilebilmesi için bir kamera yardımıyla yolcuların yüzleri okunur. İç hatlar ve uluslararası uçuşlarda aktif şekilde kullanılmaktadır.

Amerika Merkezli bir şirket olan Cigna, sağlık sigortasında kullanılan imzalar yerine yüz tanıma sistemlerini kullanarak imza sahtekârlığını engelleyerek sigorta talebinde bulunabilmesine olanak sağlıyor.

Dünya devi Coca-Cola yüz tanıma sistemlerini bazı ülkelerde otomatlara yerleştirerek geri dönüşüme teşvik çalışmaları yapmıştır. Bazı ülkelerde iste kişiselleştirilmiş reklamlar sunmak amacıyla yüz tanıma sistemini kullanmıştır.

Yüz tanıma teknolojisini en etkin biçimde kullanan firmalardan biri de Facebook' tur. Facebook' a yüklenen her fotoğraftaki yüzleri bularak etiketleme amacı ile yüzlerin kime ait olduğu hakkında öneriler sunar. Kişisel veriler kapsamında da yüz tanıma özelliğini devre dışı bırakma hakkını kullanıcıya tanımıştır.

Google, yüz tanıma teknolojisini Google fotoğraflarda kullanarak resimleri sıralayabilir ve tanınan kişileri etiketlemek için kullanmaktadır.

MAC Makyaj ürünleri markası yüz tanıma teknolojisi ile artırılmış gerçekliği kullanarak müşterilerinin, makyaj malzemelerini yüzlerinde sanal bir şekilde denemelerine imkân vererek test için kullanılan malzeme sarfiyatını azaltmıştır.

Mc Donalds ise yüz tanıma teknolojisini insanların tepkilerini değerlendirmek ve çalışma kalitesini artırmak için kullanmıştır. Mc Donalds' ın asıl amacı, çalışanların müşteriler ile ilgilendiği sırada müşterilerin gülümseyip gülümsemediğini tespit ederek bunun üzerinde analizler yapmak olmuştur.

Snapchat firması yüz tanıma sistemlerinin öncü kullanıcılarından biridir. Kullanıcıların yüzlerine uygulanan filtrelerin kullanılmasına imkân sağlayan bir uygulamadır. Bu filtreler çiçekler, köpek yüzleri, kelebekler gibi hayal edilebilecek her şeyi yüze uygulayabilir.

1.3.3 Yüz Tanıma Sistemi Avantajları

Yüz tanıma sisteminin en belirgin özelliği daha sağlam bir güvenlik sistemidir. Kişisel anlamda bireysel cihazları kilitleri, güvenlik kameraları için güvenlik aracı olarak kullanılırken

devlet tarafında suçluları, tehlike yaratacak olayların önüne geçmek için tanımlama yapmaya yardımcı olarak kullanılır. Hırsızları, izinsiz girişler, küçük suçlara olan eğilimi azaltmak için büyük oranda caydırıcılık sağlayabilir. Yüz tanıma sistemleri örnek olarak şirket bilgisayarlarına eklendiğinde teorik olarak çalınması imkânsız olacaktır. Bu da bilgi teknolojisi çalışanlarına bir avantaj olarak geri döner. Teorik olarak parolanın çalınması ve öğrenilebilmesi mümkündür fakat yüz tanıma sistemini kandırmak biraz daha zor olacaktır.

Polisler yaptığı denetimler esnasında vatandaşı durdurup üst arama konusunda ciddi sorunlar yaşayabiliyorlar. Yüz tanıma sistemleri kalabalık arasında şüpheli kişileri ayırarak güvenlik kuvvetlerine ciddi bir avantaj sağlayabilir.

Mağazalardaki insanların denetimine de avantaj sağlayan yüz tanıma sistemleri özellikle ödeme noktalarında oluşan kuyrukların önüne geçebilmek, sıra beklemeyi en aza indirmek bakımından normal ödeme şekillerine göre daha avantajlıdır. Özellikle COVID sonrası şirketlerin değişen politikalarıyla, mağazaya giren müşterinin veri tabanında eşleşip sağlık durumu hakkında bilgi alınması ciddi bir avantaj sağlar.

En büyük avantajının güvenlik olması yüz tanıma sistemi kullanan firmalar için zamandan tasarruf sağlar. Sadece saniyeler içinde yüzü algılar kimliğin doğrulanmasını sağlayarak güvenlik birimlerine cevap verebilir. Hızlı bir yöntem olması da zaman bakımından büyük kazanım sağlar. Firma içi veya dışı bütün sistemlere bütünleşebilir olan yüz tanıma sistemleri firmanın güvenlik alanında yapacağı yatırımları da azaltarak avantaj sağlayabilir.

1.3.4 Yüz Tanıma Sistemi Dezavantajları

Özellikle halka açık yerlerde video kayıtlarında bulunmaktan bütün insanlar memnun olmayabilir. Söz konusu yüz tanıma olduğundan dolayı kişisel veriler ön plana çıkmaya başlar. Aynı zamanda yüz veri tabanları kötü niyetli kişilere de ilham verebilir.

Her yerde bulunan video kameralar ile yapay zekâ sistemlerine ister istemez yardımcı olunabilir. Yüz tanıma sistemleriyle suçlu insanların tespitinin yapılması çok normaldir. Fakat bunun yanında sadece suçlu insanların değil masum insanların tespiti çok kolaylıkla yapılabilir. Bu da insanın bireysel özgürlüğünü kısıtlayan bir durum oluşturur.

Bununla birlikte yüz tanıma sistemi yüksek oranla başarılı çalıştığı gibi hata yapma olasılığı da bulunur. Örneğin bir masumu suçlu gibi gördüğünde veya bir suçluyu masum gibi gördüğünde hata ciddi bir güvenlik açığına sebebiyet verebilir.

Özellikle halka açık alanlarda çalışan yüz tanıma sistemlerinin etik ve mahremiyet tartışmaları devam etmektedir. Bu konuda ki suistimallerin engellenebilmesi için yüz tanıma sistemlerini Avrupa komisyonu 5 yıla kadar durdurmayı planladıkları belirtilmiştir.

Yüz tanıma teknolojisi ile gelişen Deepfake teknolojisinin gelişmesi de yüz tanıma sistemlerine güvenilirliği azaltabilecek etkenlerden biridir. Deepfake videoları gerçeğinden ayırmak için tespit algoritmaları kullanılabilir (Bahar, M. S. ve Buluş, E., 2021).

Yüz tanıma sistemlerinin dezavantajlarından biri de sistemlere yapılacak olan maliyet olarak görülebilir. Güvenliğine önem veren büyük firmalar için bu kalem çok sorun olmaz iken küçük ölçekli firmaların veri depolamaya ve sistem için kullanacağı donanım hızlarına erişebilmek için belli yatırımları yapmaları gerekmektedir. Bu da firmaya hatırı sayılır bir mali yük getirmektedir. Dolayısıyla maliyet yüz tanıma sistemlerinin dezavantajları arasına alınabilir.

1.4 Literatür Özeti

Bir insanın görünümünün en karakteristik özelliği olan yüz, diğer yüzlerden ayırt edilebilirliği özelliğinden dolayı elverişli koşullar oluşturmaktadır. Görüntülerde bir yüzü ayırt edebilmek için çeşitli teknolojiler kullanılmaktadır. Literatür araştırmasında çoğunlukla ön yüzler için geçerli olan araştırmalar incelenmiştir. Yöntemlerin avantaj ve dezavantajlarına değinilmiştir.

1.4.1 Eigenface

Öz yüzler, yüz tanıma için en geniş araştırma alanlarından birine sahiptir. Karhunen Loeve açılımı eigenpicture (öz resim), eigenvector (öz vektör), principal component analysis(temel bileşen analizi) olarak da bilinmektedir. Yapılan araştırmalarda (Sirovich, L. ve Kirby, M., 1987)(Kirby, M. Ve Sirovich, L., 1990) yüz resimlerini test etmek amacıyla temel bileşen analizi kullanılmıştır. Yüz resimlerinin tekrar yapılandırılması hedefiyle, her yüz resminin ortak ağırlık standardı olduğunu savunmuşlardır(öz resim). Her yüz resminin ortak

ağırlığını tanımlayan sekiz öz yüz görüntüsünün öz resme yansıtılmasıyla tanımlama yapılmıştır. (Turk, M. ve Petland, A., 1991)Çalışmada yüz algılama işlemi için Kirby ve Sirovich tekniğiyle desteklenmiş öz yüzler kullanılmıştır.

Öz vektörler, temel bileşen olan öz yüzlerin kovaryans matrisidir. Her bir yüz resmi, öz yüzlerin farklı bir kombinasyonunu tam olarak temsil edebilir. Farklı bir bakış açısıyla en iyi öz vektör kullanılarak da yüz olup olmadığı tahmin edilebilir. Araştırmada yüzler, sırasıyla ışık, yön ve boyut farklılıkları nedeniyle, yüzde 96, yüzde 85 ve yüzde 64 doğru tahmin edilmiştir. Veri tabanlarında ise 16 kişinin 2500 adet resmi vardı.

Görüntülerin büyük kısmında arka plan olduğundan dolayı sonuçlar değişmiştir. Farklı ışık altında sistem performansı önemli şekilde değişmiştir. Bu deneydeki görüntüler arasındaki korelasyon, tanıma performansı adına çok iyi sonuçlar vermemiştir. Işık normalizasyonu (Sirovich, L. ve Kirby, M., 1987) özyüz tanıma için gereklidir.

Bir diğer araştırmada (Pentland, A., 1994) ise özyüz üzerindeki özellikler daha derinlemesine incelendi. Gözler, burun, ağız gibi yüz özelliklerini öz gözler, öz burun, öz ağız gibi özelliklere indirgediler. Bunlardan bir uzay oluşturup kullandılar. Bu yöntemde FERET veri tabanı kullanılarak yüzde 95 lik bir yüz tanıma oranı elde edilmiştir. Fakat bu araştırmada ışığın yüz tanımadaki etkisi değişmemiştir.

(Chang, K. vd., 2003)Bu çalışmada yapılan deneylerde kullanılan temel bileşen analizi sonucunda, kulak görüntüsü tanımlarının yüz tanımlar ile oranının aynı olduğuna rastlanmıştır. Bu kanı ile birlikte kulak ve yüz tanımanın aynı anda kullanılmasının tanıma performansının artması ile sonuçlanmıştır.(Verlinde, P. ve Mayoraz, E., 1998)Yüz ve kulağın birlikte kullanılması gözetim uygulamalarında çok sık rastlanır.

Çizelge 1.4.1.1 Özyüzler performans değerleri (Kekül, H. vd., 2018).

Görüntü Şekli	Test Verisi Sayısı	Doğru sayısı	Hata Sayısı	Başarı Yüzdesi
İdeal Görüntü	80	77	3	96,25
Göz Kısmı Kapalı Görüntü	80	11	69	13,75
Ağız-Çene Kapalı Görüntü	80	6	74	7,5
Burun Kapalı Görüntü	80	25	55	31,25
Gürültü Oluşturulmuş Görüntü	80	75	5	93,75

1.4.2 Sinir Ağları

Sinir ağlarının doğrusal çalışmaması yüz tanıma uygulamalarında tercih edilmesinin en önemli sebeplerinden biri olabilir. Yüz tanıma uygulamalarında kullanılan ilk yapay sinir ağları (YSA) tekniği, her bir yüz için ayrı bir ağ bulunduran WISARD tek katmanlı ağıdır (Stonham, T. J., 1986). Yüz tanımanın başarılı olabilmesi için, yapay sinir ağının yapısı çok önemlidir. (Lawrence, S. vd., 1997)Çok katmanlı algılayıcı ve (Weng, J. J. Ve Huang, T. S., 1993, May)evrişimli sinir ağları denemeleri yapılmıştır. (Lawrence, S. vd., 1997)Bu çalışmada hibrit, kendi kendini organize edebilen bir sinir ağı (SOM) önerildi. Bu ağ ile uzayda bulunan görüntüler çıktı uzayında sayısallaştırılır ve görüntüler üzerindeki küçük değişikliklere

değişmezlik sağlanır. Evrişimsel ağ ise görüntü üzerindeki döndürme, boyut, ölçekleme, deformasyon gibi diğer özellikleri ayırt eder. Yazarlar veri tabanında 400 görüntü içinde 40 kişinin yüzde 96,2 oranında doğru tanındığını test etmiştir.

Sınıflandırma süresi 0,5 saniyeden daha kısa olmasına rağmen eğitime süresi 4 saati bulabilir.(Lin, S. H. vd., 1997) Bu çalışmada karar tabanlı sinir ağı olan (DBNN), olasılıksal karar tabanlı sinir ağı önerildi (PDBNN). PDBNN düzgün bir ağ topolojisi yoktur. Ağ K alt ağına bölünerek her alt ağ bir yüzü tanımak için ayrılmıştır. Ayrılan alt ağ Gaussian aktivasyon fonksiyonunu kullanarak olasılık yoğunluğunu tahmin eder.

Çizelge 1.4.2.1 Farklı yüz tanıyıcıların, FERET veri tabanındaki 200 kişi üzerindeki performans değeri. (Lin, S. H. vd, 1997)

Method	Training Accuracy	Testing Accuracy
Probabilistic DBNN	97 %	99 %
Traditional DBNN	100 %	96 %
Multi-Layer Perceptron	99.5 %	87.5 %

PDBNN bazlı yüz tanıma uygulamaları sinir ağı yapısı dolayısıyla daha avantajlıdır. (Lin, S. H. vd., 1997) PDBNN tabanlı tanıyıcının 200 kişiyi 1 saniye içinde yüzde 96 ya kadar doğru tanıma yapabildiği kanıtlanmıştır. Sonuç olarak sinir ağı tabanlı tanıyıcılar kişi sayısı arttığında sorunlarla karşılaşabilecektir.

Bir diğer çalışmada (Buluş, E., 2021) cinsiyet farklılıklarının sınıflandırılması hedeflenmiştir. Yapay sinir ağları temel alınarak yapılan bu çalışmada, caffemodel ve vgg16 yöntemleri kullanılmıştır. Yaklaşık 57.000 ünlüden oluşan veri seti içinden 15.000 erkek ve 15.000 kadın eğitim için 5.000 erkek ve 5.000 kadın doğrulama için, 8.000 erkek ve 8.000 kadın ise test için seçilmiştir. Resimler 227x227 piksel boyutundadır. Yapay sinir ağları yöntemi olan

CaffeModelin tahmin başarısı %55 olur iken, MVGG-16 yöntemi ile %94 başarı oranına ulaşılmıştır.

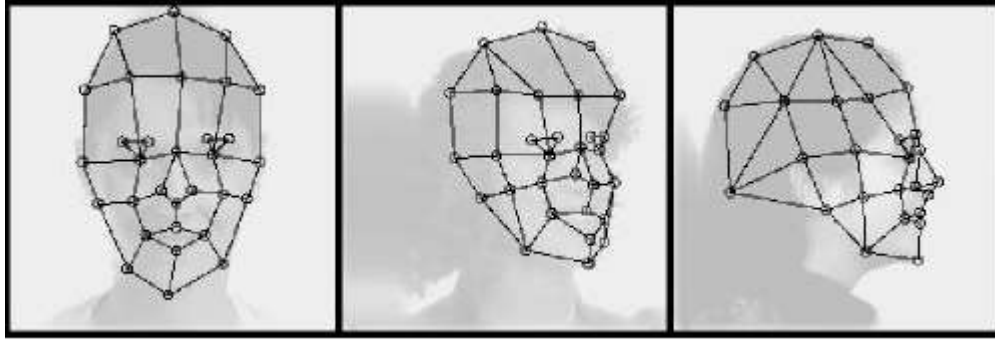
Çizelge 1.4.2.2 Cinsiyet tahmini başarımlar yüzdeleri (Buluş, E., 2021)

	Başarı	Model	Tek resim için tahmin süresi ms
Test verileri ile CaffeModel	%55,5±5	Test için CaffeModel	17
Test verileri ile MVGG-16	%94,5±2	MVGG-16	116

1.4.3 Grafik Eşleştirme

Yüz tanıma için farklı bir seçenekte grafik eşleştirme yaklaşımıdır. (Lades, M. vd., 1993)Bu çalışmada yüzü bulabilmek için, elastik grafik eşleştirmeyi kullanılan dinamik bir bağlantı yapısı sunuldu. Dinamik bağlantı mimarisi, klasik yapay sinir ağlarının bir uzantısı olarak da belirtilebilir. Daha önce öğrenilmiş olan nesnelerin köşeleri güçlü bir tanıma için belirli etiketler oluşturur. Bu etiketler ile elastik grafik arama eşleştirmesi ile tanıma formüle edilebilir. Çalışmada 87 kişilik veri tabanından iyi sonuçlar alındığı bildirilmiştir. Fakat 87 yüzün karşılaştırılması 25 saniye olduğundan dolayı performans açısından yeterli değildir.

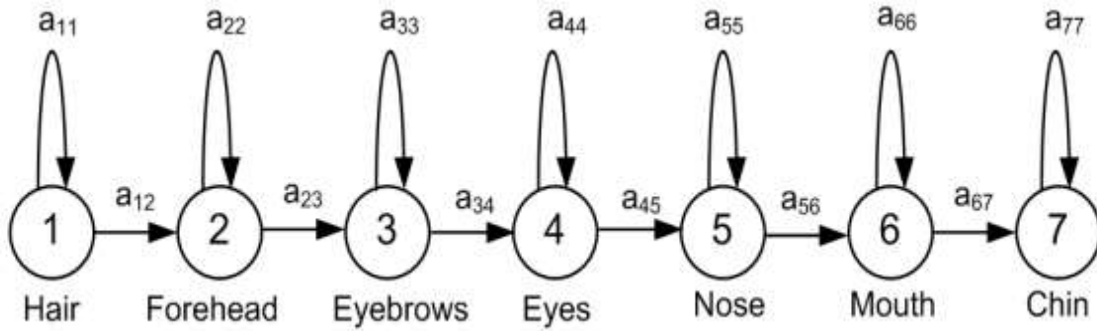
Bu referansta (Wiskott, L. ve Von Der Malsburg, C., 1996) genişletilmiş bir teknik kullanılarak 112 kişinin ön yüzüyle bir galeri oluşturuldu. Bazı değişen yüz ifadeleri nedeniyle araştırma bozulsa da farklı açılarda ki yüzlerde 86,5 ve 66,4 tanıma oranları bildirilmiştir. 15 ve 30 derece döndürülmüş 110 yüzün 112 önyüzden oluşturulmuş bir yüz veri tabanı ile eşleşme testleri dinamik bağlantı mimarisi ile diğer tekniklerinden üstünlük sağlamıştır. Fakat performans açısından pahalı olarak görünmüştür.



Şekil 1.4.1 Yüzün çeşitli açılardan grafikleştirilmiş hali (Wiskott, L. ve Von Der Malsburg, C., 1996)

1.4.4 Hidden Markov Model (HMM)

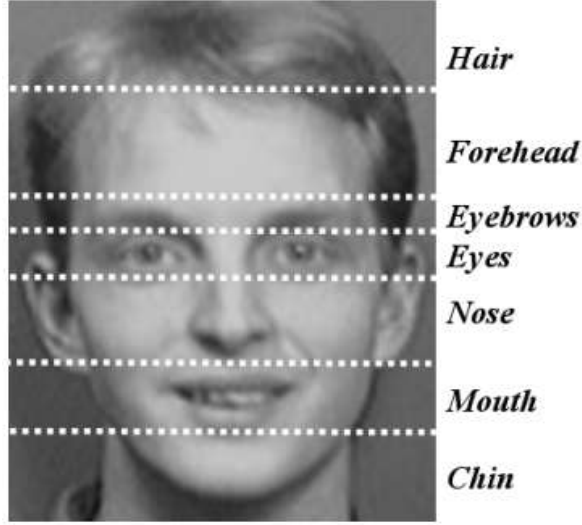
Konuşma uygulamaları için HMM tabanlı vektör zaman serilerinin olasılıksal modellenmesi başarılı olduğu için (Samaria, F. ve Fallside, F., 1993) bu çalışmada yüz tanıma üzerine uygulanması düşünülmüştür. Yüzler, ağız, burun ve gözler gibi bölgelere ayrılarak Markov modeli ile ilişkilendirilebilecek duruma getirildi. Bu model için tek boyutlu girdi gerektiğinden, iki boyutlu görüntüler zamansal ve uzaysal dizilere dönüştürülmüştür.



Şekil 1.4.2 Yedi, tek boyutlu HMM modeli(Davari, P. ve Naimi Miar, H., 2008, January)

(Samaria, F. S. ve Harter, A. C. 1994, Aralık) Bu çalışmada bir bant örnekleme tekniği kullanılarak yüz den uzaysal bir dizi çıkarılmıştır. Vektör serileri pikseller olarak yüzleri temsil etmiştir. Her vektör L çizgisi olarak adlandırılmıştır ve art arda gelen gözlemlerin kesiştiği çizgiye M adı verilmiştir. Sonrasında bir test görüntüsü, gözlem dizisi olarak örneklenir ve veri

tabanındaki bütün HMM ile karşılaştırılır. En iyi eşleşme en yüksek olasılıklı olandır ve test yüzünün kime ait olduğu bulunur.

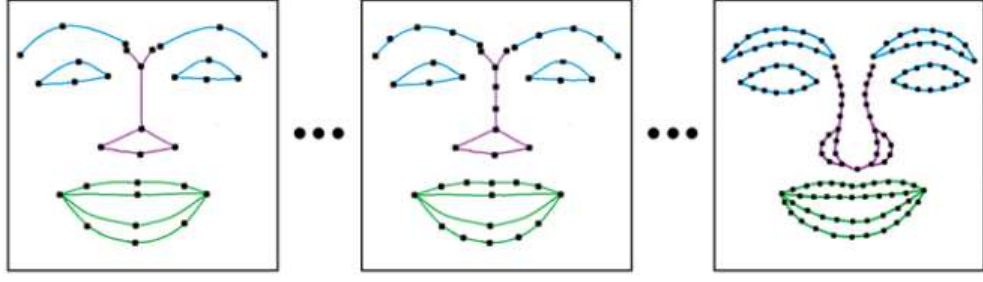


Şekil 1.4.3 Yüzün HMM model için yedi parçaya ayrılmış şekli.(Davari, P. ve Naimi, H. M. 2008, January)

Sonuç olarak ORL veri tabanındaki 40 kişinin 400 görüntüsü arasında tanıma oranı yüzde 87 olarak görülmüştür. Bunun yanında 2 boyutlu HMM taramasının da yüzde 95 tanıma oranına ulaştığı bildirilmiştir. Performans ile ilgili bilgi verilmediği için çözümün pahalı olduğu düşünülmektedir.

1.4.5 Geometrik Özellik Karşılaştırması

Geometrik özellik karşılaştırması yüz geometrisini baz alarak bir yüzün geometrik özelliklerinin hesaplanması ile yüzün tanınmasını hedefleyen bir yaklaşımdır. Çalışmada (Tamura, S. vd, 1996) bir yüzün grafiksel genel özelliklerinin 8 x 6 piksel düşük çözünürlükte dahi tanınabileceği söylenmiştir. Bu grafiksel özellikler gözler, kaşlar, burun, ağız ve yüz hatlarının konumunu ve boyutunu temsil eden bir vektör olarak tanımlanabilir.



Şekil 1.4.4 Yüz hatlarının geometrik olarak gösterimi.(Benitez-Garcia, G. , Nakamura, T. and Kaneko, M.,2017)

(Kanade, T., 1974)Yüz tanıma konusundaki önemli çalışmalarından biri 1973 yılında, yüzün geometrik özellikleri temel alınarak yapıp yüzde 75 tanıma oranına ulaşılabilmiştir. Her yüz başına bir test ve bir model görüntü kullanılmıştır. (Goldstein, A. J. vd, 1971)(Kaya, Y. Ve Kobayashi, K., 1972) Bu çalışmalarda ise el ile bulunan yüz özellikleriyle çalışan bir programın kabul edilebilir bir biçimde tanıma yapabildiği gözlemlenmiştir. (Brunelli, R. ve Poggio, T., 1993)Bu çalışmada ise burun uzunluğu ve genişliği,çenenin şekli gibi yüz özelliklerini otomatik olarak geometrik özelliklerine dönüştürmüşlerdir.47 kişilik veri tabanı içinde Bayes sınıflandırıcısı ile yüzde 90 tanıma oranına ulaşabildiklerini bildirmişlerdir.

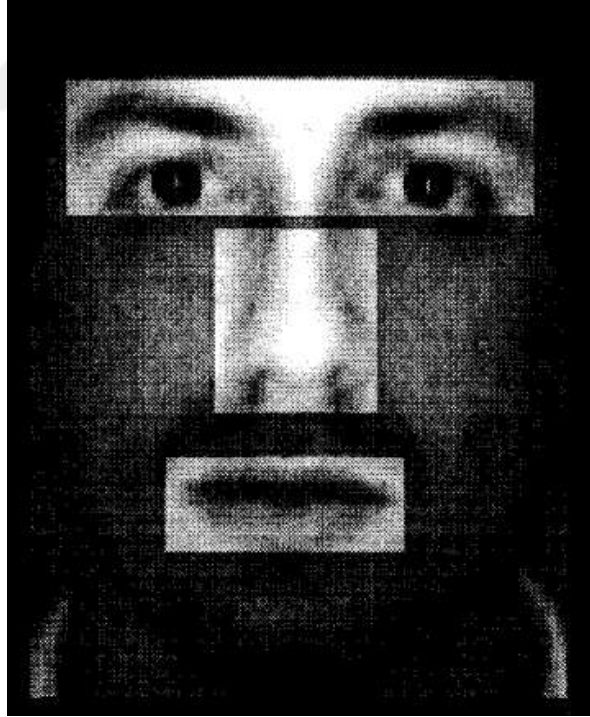
(Cox, I. J. ve Yiannos, P. N., 1996) Bu çalışmada yüzde 95 oranında başarıya ulaşan bir yaklaşımda bulunulmuştur. Bu yaklaşımda her yüz 30 farklı uzaklıkta gösterilmiştir. Veri tabanı 685 kişiden oluşturulmuştu.(Chellappa, R. vd., 1992) Bu çalışmada ise veri tabanı depolama alanı iyileştirmek amacıyla yüzlerin sadece özellik noktaları alınmıştır. Bu noktalar belirlenirken Gabor dalgacık ayrıştırması kullanılmıştır. Her yüz için 35-40 adet özellik noktası oluşturulmuştur. Topolojik olarak özellik noktalarının belli olduğu bir görsel ile eşleştirme işlemi yapıldı. Yüz tanıma doğruluğu oranı yüzde 86 doğru kişiyi bulma oranı ise yüzde 94 olarak gözlemlendi.

Sonuç olarak yüz tanımda kullanılan geometrik özellik karşılaştırılması büyük veritabanlarında kullanılmak için uygun olabilir. Yüz özelliklerinin geometrik konumlarının düzgün eşleştiği durumlarda gayet yüksek tanıma oranları verecektir. Fakat günümüzde otomatikleştirilmiş canlı uygulamalarda yüksek tanıma oranlarına ulaşması zordur ve hesaplama süresi gerektirir.

1.4.6 Şablon Eşleştirme

Şablon eşleştirme geometrik özellik karşılaştırmanın başka bir çeşidi olarak tanımlanabilir. Bir test görüntüsü iki boyutlu yoğunluk değeri olarak tanımlandığında, Öklid mesafesi gibi metrik bir değer kullanılarak ölçülmüş olan bir yüzün, şablon olarak kullanılıp karşılaştırılması yöntemi şablon eşleştirmeye örnektir. Bu yöntemde birden fazla şablon kullanılarak tanıma oranı artırılabilir.

(Brunelli, R. ve Poggio, T., 1993) (Baron, R. J., 1981) Bir yüz birden fazla şablona sahip olabilir. Yüz görüntüleri parçalara bölünerek farklı şablonlar oluşturulabilir. (Brunelli, R. ve Poggio, T., 1993) Bu çalışmada Bruneli ve Poggio, şablon olarak gözleri, burnu ve ağzı seçti. Geometrik karşılaştırma ve şablon eşleştirme algoritmasının yüzler üzerindeki performansını karşılaştırmışlardır. 47 kişinin görüntüsünü içeren veri tabanı içinde, şablon eşleştirme yüzde 100 tanıma oranıyla, yüzde 90 tanıma oranına ulaşabilen geometrik özellik karşılaştırmasının, performans olarak önüne geçmiştir. Eigenfaces yönteminde bahsedilen öz yüzler ve öz özellikler, şablon eşleştirmenin şablonları gibi düşünülebilir.



Şekil 1.4.5 Yüzün belli bölgelerinin şablon eşleştirme yöntemiyle karşılaştırılması. (Brunelli, R. ve Poggio, T., 1993)

Bu yöntemde dezavantaj olarak görülen kısım yüz tanıma esnasında verilmiş olan test görüntüsündeki özelliklerde saklıdır. Her yüzün kendine göre özellikleri olacağından dolayı,

bireysel yüzleri tanımak zor olabilecektir. Bir şablondan yola çıkıldığı için bireysel yüz özellikleri önemini yitirecektir. Şablon tabanlı eşleştirme düşüncesi yüz tanıma tekniklerinin performansını iyileştirmek için kullanılabilir uygun bir teknik olabilir.

1.4.7 3B Dönüştürülebilir Model

Dönüştürülebilir yüz modeli, belli sayıda örneğin, şekil ve doku vektörlerinin, gerçekçi bir insan yüzünü tanımladığı vektör uzayı olarak tanımlanabilir.(Vetter, T. ve Poggio, T., 1997). 3 Boyuta dönüştürülebilir modeli uygulamak için 2 yöntem kullanılır. Yüz modelinin dokusunu ve yapısını belirten katsayılar belirlenir. İkinci yöntem ise yapay yüz oluşturabilmek için 3 boyut oluşturma araçları da kullanılabilir (Beymer, D. Ve Poggio, T., 1995)(Zhao, W. Y. ve Chellappa, R., 2000, Mart).

(Blanz, V. ve Vetter, T., 2003)Bu çalışmada 3 boyutlu şekilleri bozulabilecek olan yüzleri bir bilgisayar simülasyonu ile gerekli şartları ve aydınlatma ayarlanır. Kullanılan algoritma, girdi olarak verilen resimdeki şekilleri, dokuyu ve ortam koşullarını tahmin eder. Bu yapıda yüz döndürme, aydınlatma işlemleri çok kolaydır. Bu yöntem insan yüzü üzerindeki gölgeleri ve yansımaları hesaba katarak işlem yapar. Yüzlerin özellikleri 3 boyutlu bir yüz modeline dayanmaktadır. Bu özellikler yüzlerin 3 boyutlu yüz tarama setinden öğrenilmiştir. Çalışma (Blanz, V. ve Vetter, T., 2003) da sunulan algoritma, kafanın konumu, ve aydınlatma özellikleri yüz tanıma işleminin başlama adımındır ve sistem güvenilirliği büyük ölçüde bu adıma bağlıdır.



Şekil 1.4.6 CMU-PIE veritabanından üç katmanlı yüz tamamlama aşaması. (Blaiz, V. ve Vetter, T., 2003)

3 boyut dönüştürülebilir model tanıma oranları ise CMU-PIE veri tabanında yüzde 95 olarak ölçüldü. FERET veri tabanında ise yüzde 95,9 olarak ölçülmüştür.

2. KULLANILAN YÖNTEMLER

2.1 Anaconda

Python dilinin popülerliğini artırması ile birlikte özellikle bilimsel araştırmalar için yoğun bir şekilde kullanılmaya başlanmıştır. Bütün konular üzerinde kütüphanesinin olması python dilini dahada cazip hale getirmektedir. Python dilini verimli bir şekilde kullanabilmek için Anaconda platformu oluşturulmuştur. 7500 den fazla açık kaynak paketiyle python dilinin en kullanışlı ortamlarından biridir.

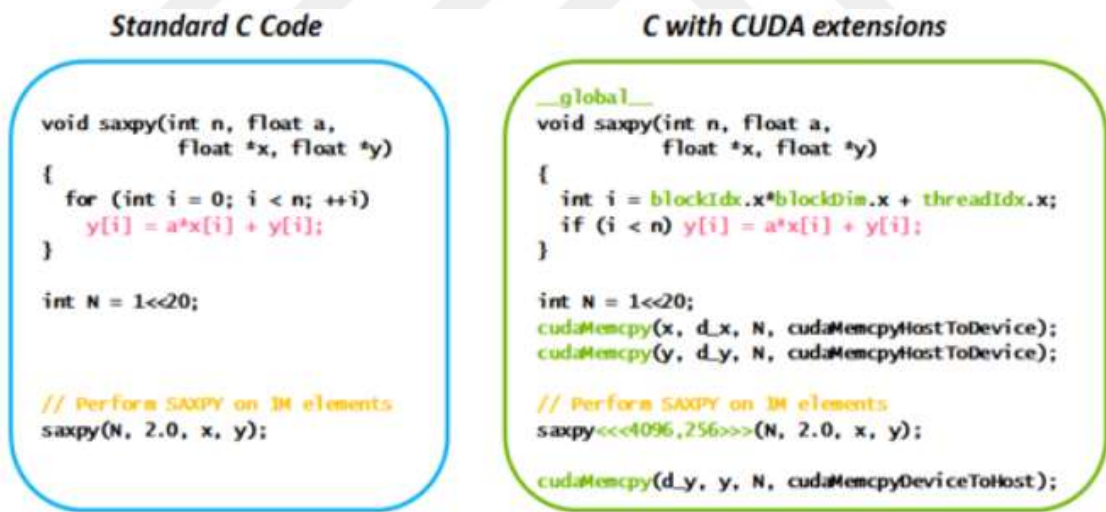
Paket yöneticisi olarak tanımlanabilen Conda, varolan paketleri bulmaya ve yenilerini yüklemeye olanak sağlar. Başka bir pakete ihtiyaç olduğunda farklı bir ortama geçmeye gerek kalmadan Conda üzerinde paketler çalıştırılabilir. Python versiyon farklılıklarından örneklendirilirse, farklı python sürümünü çalıştırmak için ayrı bir ortam Conda üzerinde oluşturulabilir.

Conda'nın yazılan kodun test edilmesi için Travis CI ve AppVeyor gibi entegrasyon sistemleri ile birleştirilip çalışması sağlanabilir.

2.2 Cuda

NVIDIA tarafından üzerinde çalışılan bir programlama modeli olan CUDA, bir programlama modeli ve bilgi işlem platformudur. CUDA kullanıcıların geliştirdikleri programların Graphics Processing Unit (GPU) tarafından desteklenerek daha performanslı çalışmasına destek olmaktadır. Günümüzde her alanda kullanılan bilgisayarların, araştırma ve uygulama projelerinde ekstra performans ihtiyacı duyulan her alanda CUDA kullanılmaktadır. Örnek olarak tıp alanında geliştirilmeye çalışılan tedaviler için, çevirim içi satış alanlarındaki reklamların yerleşimi, satın alınan ürünlerin analizleri, otonom araç çalışmalarının tümünde CUDA kullanılmaktadır.

CUDA ile geliştirilen programlarda, C, C++, Fortran, Python ve Matlab dillerinde yazılan kodlara anahtar kelime biçiminde uzantı eklenir. Bu anahtar kelimeler uygulamanın GPU üzerinde çalışması için yönlendirilmesini sağlar.



Şekil 2.2.1: Cuda Anahtar Kelimeler Kullanımı Örneği

NVIDIA geliştirici web sitesinde yer alan videolarla Cuda paralel programlama dili öğrenilebilir. Geliştiriciler AI' den HPC' ye kadar birden çok uygulamada performansını arttırmak için Cuda içinde oluşturulmuş olan kütüphanelerden faydalanabilir.

Grafik kartları, grafiklerin performanslı görüntülenebilmesi için paralel hesaplamalar yapabilmek için tasarlanmıştır. Bu paralel hesaplama yeteneğinin performansını kullanabilmek

için Cuda bir araçtır. İşlemciler ile karşılaştırıldığında 30 – 100x hız artışı sağlanabilir. GPU' lar daha büyük CPU' lara göre küçük Aritmetik mantık birimlerine (ALU) sahiptirler. Bu ALU' lar ile ekrandaki tüm piksellere giden renkleri hesaplamak için kullanılırlar. Dolayısıyla bu küçük birimler paralel hesaplama yapmak amacı ile kullanıldığında işlem hızlarında büyük yükselişler olacaktır.

CUDA paralel programlama dili için desteklenen GPU ve ekran kartlarının listesi aşağıdadır.

Çizelge 1.4.7.1 CUDA programlama için desteklenen GPU ve Ekran kartı tablosu.

Versiyon	GPU	Ekran Kartı
1.0	G80, G92, G92b, G94, G94b	GeForce 8800GTX/Ultra, 9400GT, 9600GT, 9800GT, Tesla C/D/S870, FX4/5600, 360M, GT 420
1.1	G86, G84, G98, G96, G96b, G94, G94b, G92, G92b	GeForce 8400GS/GT, 8600GT/GTS, 8800GT/GTS, 9600 GSO, 9800GTX/GX2, GTS 250, GT 120/30/40, FX 4/570, 3/580, 17/18/3700, 4700x2, 1xxM, 32/370M, 3/5/770M, 16/17/27/28/36/37/3800M, NVS290, NVS420/50
1.2	GT218, GT216, GT215	GeForce 210, GT 220/40, FX380 LP, 1800M, 370/380M, NVS300, NVS 2/3100M
1.3	GT200, GT200b	GeForce GTX 260, GTX 275, GTX 280, GTX 285, GTX 295, Tesla C/M1060, S1070, Quadro CX, FX 3/4/5800
2.0	GF100, GF110	GeForce (GF100) GTX 465, GTX 470, GTX 480, Tesla C2050, C2070, S/M2050/70, Quadro Plex 7000, Quadro 4000, 5000, 6000, GeForce (GF110) GTX 560 TI 448, GTX570, GTX580, GTX590

Çizelge 1.4.7.2 CUDA programlama için desteklenen GPU ve Ekran kartı tablosu.(Devamı)

2.1	GF104, GF114, GF116, GF108, GF106	GeForce 610M, GT 430, GT 440, GTS 450, GTX 460, GT 545, GTX 550 Ti, GTX 560, GTX 560 Ti, 500M, Quadro 600, 2000
3.0	GK104, GK106, GK107	GeForce GTX 770, GTX 760, GTX 690, GTX 680, GTX 670, GTX 660 Ti, GTX 660, GTX 650 Ti BOOST, GTX 650 Ti, GTX 650, GT 640, GT 630, GeForce GTX 780M, GeForce GTX 770M, GeForce GTX 765M, GeForce GTX 760M, GeForce GT 750M, GeForce GT 745M, GeForce GT 740M, GeForce GTX 680MX, GeForce GTX 680M, GeForce GTX 675MX, GeForce GTX 675M, GeForce GTX 670MX, GTX 670M, GTX 660M, GeForce GT 650M, GeForce GT 645M, GeForce GT 640M, Quadro K600, Quadro K2000, Quadro K4000, Quadro K5000
3.5	GK110	Tesla K20X, K20, GeForce GTX TITAN, GTX 780

2.3 Dlib

Karmaşık yazılımları oluşturmak için kullanılan dlib bir C++ kütüphanesidir. Çeşitli makine öğrenimi algoritmaları kullanımını kolaylaştırmak için oluşturulmuştur. Endüstri de ve akademide sıkça kullanılan dlib kütüphanesi açık kaynak kodludur ve istediğiniz uygulama içinde çalıştırmanıza olanak sağlar. Github üzerinde dlib ile ilgili çoğu bilgiye ulaşılabilir.

2002 yılında geliştirilmeye başlanan dlib birçok aracı içerecek şekilde büyümüştür. Veri madenciliği, istatistik, Bayes ağları, makine öğrenimi, grafik arayüzleri gibi iş kolaylaştıracak bir sürü araçları içinde bulundurur. Kullanım kolaylığına ve taşınabilir çalışmasına odaklanılmış olan kütüphane özellikle kurulum, yükleme gerektirmeyecek şekilde

tasarlanmaktadır. Bundan dolayı dlib, API ler ile çalışır. Şu anda Linux, Windows, Mac OS, Solaris gibi işletim sistemlerinde çalıştığı bilinmektedir.

2.4 OpenCV

(OpenCV, 2022) OpenCV,1999 yılında Gary Bradsky tarafından intel firmasında geliştirilmeye başlanmış, 2000 yılında piyasaya sürülmüştür. OpenCV kütüphanesinin açılımı Open Source Computer Vision Library yani açık kaynak kodlu bilgisayar görüntü kütüphanesidir. Adından da anlaşılacağı üzere bilgisayarın görüş açısının kullanıldığı, kamera veya dijital görme araçlarının yönetilebildiği, bilgisayarın gördüğü görüntülere anlam verilebildiği bir kütüphanedir. Günümüzde bilgisayar kullanımının yaygınlaşmasıyla beraber otomatik yapılması gereken birçok iş ortaya çıkmıştır. Konu ile ilgili olarak teknoloji ilerledikçe bilgisayarların görüşüne ihtiyaç artmaktadır. Yüz tanıma, yüz ifadesi algılama, nesne tanıma, hareket belirleme, hareketli nesne takibi gibi tüm sorulara çözüm olarak OpenCV kullanılabilir. OpenCV kütüphanesi içinde birden fazla modül bulunmaktadır.

- Çekirdek Fonksiyonları (core) OpenCV kütüphanesinin temel matematik özelliklerini ve fonksiyonlarını içinde barındıran ana modüldür. Diğer tüm modüllerin temelini oluşturur.
- Görüntü işleme (imgproc) modülü görüntüleri lineer veya lineer olmadan filtrelemeyi, geometrik olarak görüntü dönüşümlerini, renk değişimlerini içerir.
- Video Analizi (video) nesne izleme algoritmalarının, hareket tahmini ve arka plan çıkarma yeteneklerinin bulunduğu bir modüldür.
- Nesne Algılama (objdetect) nesnelerin daha öncesinde sınıflarının belirlendiği modüldür.
- Video Giriş/Çıkış (videoio) video çekimi ve video kodekleriyle ilgili kullanışlı bir modüldür.

OpenCV yukarıdaki modüller gibi daha birçok görüntü ve hareketli görüntüler ile ilgili modülleri içinde bulundurur.

OpenCV çeşitli programlama dillerini desteklemektedir. C++,Java, Python OpenCV kütüphanesinin kullanıldığı en yaygın programlama dillerinden bazılarıdır. OpenCV ile python dilinin özelliklerini en iyi birleştiren API Opencv-Python API sidir. Python dili kod okunabilirliği ve basit kodlanabilirlik özelliği nedeniyle kullanımı her geçen gün hızla artmakta

ve her geçen gün daha popüler hale gelmektedir. Dilin bu özellikler programcının fikirlerini daha kolay, rahat ve daha az satırla yazmasını destekler.

2.5 Haarcascade

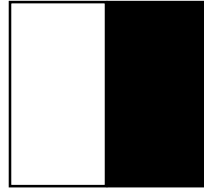
(Viola, P., & Jones, M. (2001, December))Paul Viola ve Michael Jones insan yüzünün dijital olarak tespit edilebilmesi için Viola-Jones nesne algılama sistemi üzerine çalışmalar yapmışlardır. Bu sistem genel anlamda nesnelere için kullanılan bir sınıftır. Fakat yüz tanıma bu sınıfın gelişmesinde büyük bir motivasyon kaynağı olmuştur.

Viola-Jones algoritmasını başarılı bir algoritma yapan özellikler sıralanacak olursa; yüksek algılama oranıyla yanlış-pozitif oranının düşük olması, uygulamaların hızlı kullanılabilmesi için saniyede 2 kare işlenebilmesi ve aslında amacının yüz algılama olmasıdır.

Viola-Jones algoritmasının çalışabilmesi için;

- Haar Özelliği seçimi
- İntegral görüntü oluşturma
- Adaboost öğrenmesi
- Sınıflandırıcıları basamaklama

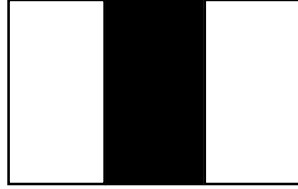
Viola-Jones algoritması ekrandaki tüm görüntünün piksellerini kullanarak görüntü üzerinde nesne aramaya odaklıdır. Şekil 2.5.1 de ekrana gelen görüntü üzerinde kullanılacak olan 4 farklı şablonu göstermektedir. Nesnelerdeki gölgeli pikseller ile açık pikselleri karşılaştırarak bir sonuç elde edilmeye çalışılır. Görüntülerdeki dikey ve yatay piksellerin koyu ve açık renkli olmasına bakılarak karar alır ve cevap verir.



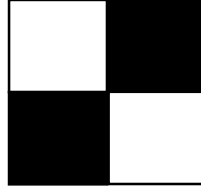
Dikey



Yatay



On - Off - On



apraz

Şekil 2.5.1 Haar Benzerlik özellikleri

(Darshan A., 2019) Şekil 2.5.1 de bulunan Haar dikdörtgen özellikleri orijinal görüntünün herhangi bazı piksellerine denk gelebilir. Ve bu dikdörtgen görünümüleri nesnelere farklı özelliklerini tanımlıyor olabilir. Her özellik, nesnelere üzerindeki kenarları veya üzerinde bulunan dokuyu ya da nesnenin kendisine ait olan parçaların birbiri arasındaki geçişlerini göstererek, nesnenin varlığını veya yokluğunu ispatlayabilir.

İnsanlar ne kadar birbirinden ayırt ediliyor olmasına rağmen yüzde, 2 göz 1 burun ve bir ağız vardır. Bu karakteristik özellikleri Haar dikdörtgen özellikleri ile karşılaştırdığımızda;

Göz çukurları, elmacık kemiklerinden daha koyudur.

Burun yüksekliği gözlerden daha açıktır.



Şekil 2.5.2 Haar özelliklerinin insan yüzüne uygulanması

Haar dikdörtgen özellikleri görüntü üzerindeki belli noktalardaki, aydınlık alanların, beyaz ve siyah dikdörtgenler olarak farklarını ortaya çıkararak görüntüyü anlamlandırmaya çalışır. Beyaz pikseller ve siyah piksellerin farkından oluşan değer daha önceden belirlenmiş olan eşik değerini geçiyor ise nesne vardır. Eşik değeri altında ise nesne yoktur.

HaarCascade eski olmasına rağmen yüz algılamada güçlü algoritmalar arasında yer alır. Haar sınıflandırıcı ile yüz, göz, dudaklar, plakalar, farklı canlılar veya cisimler içinde kullanılabilir. Haar sınıflandırıcı GitHub'da bulunabilir ve OpenCV kütüphanesi ile ulaşılabilir.

2.6 Face_recognition

(Pypi.org, 2020) Python’ da yüzleri tanıma değiştirmek için kullanılan ve basitliği ile öne çıkan bir kütüphanedir. Dlib kütüphanesinin yüz tanıma özellikleri kullanılarak oluşturulmuştur. Bu kütüphane derin öğrenme ile desteklenmiş ve doğruluk oranı yüzde 99,38’dir. Komut ile görüntü klasöründeki yüzleri okuyup, tanıma yapılabilir. Özellikleri arasında resimdeki bütün yüzleri bulma, yüzlerin özelliklerini değiştirebilme, yüzde bulunan göz, ağız, burun gibi duyu organlarını sınıflandırabilme, resimdeki yüzleri sınıflandırabilme, kime ait olduğunu belirtebilme vardır.

Kütüphane indirildikten sonra komut satırından kütüphane çağırılabilir. Kütüphanenin yüzleri tanıyabilmesi için bir klasör içinde kişilere ait yüzlerin görselleri olması yeterlidir. Görsellerin dosya isimleri, kişinin adları olarak kullanılması kişiyi, görüntünün üzerinde isminin görünmesine sebep olur.

Resimdeki yüzlerin bulunabilmesi için aşağıdaki kodları yazmak yeterli olacaktır.

```
Import face_recognition,  
  
image = face_recognition.load_image_file("resim.jpg")  
  
face_locations = face_recognition.face_locations(image)
```

Resimdeki yüz hatlarının, ağız, bunun, göz ve dudakların resimdeki yerlerini işaretlemek için aşağıdaki kodları yazmak yeterli olacaktır.

```
Import face_recognition  
  
Image=face_recognition.load_image_file("resim.jpg")  
  
Face_landmarks_list = face_recognition.face_landmarks(image)
```

Resimdeki yüzlerin tanınması için aşağıdaki kodların yazılması yeterlidir.

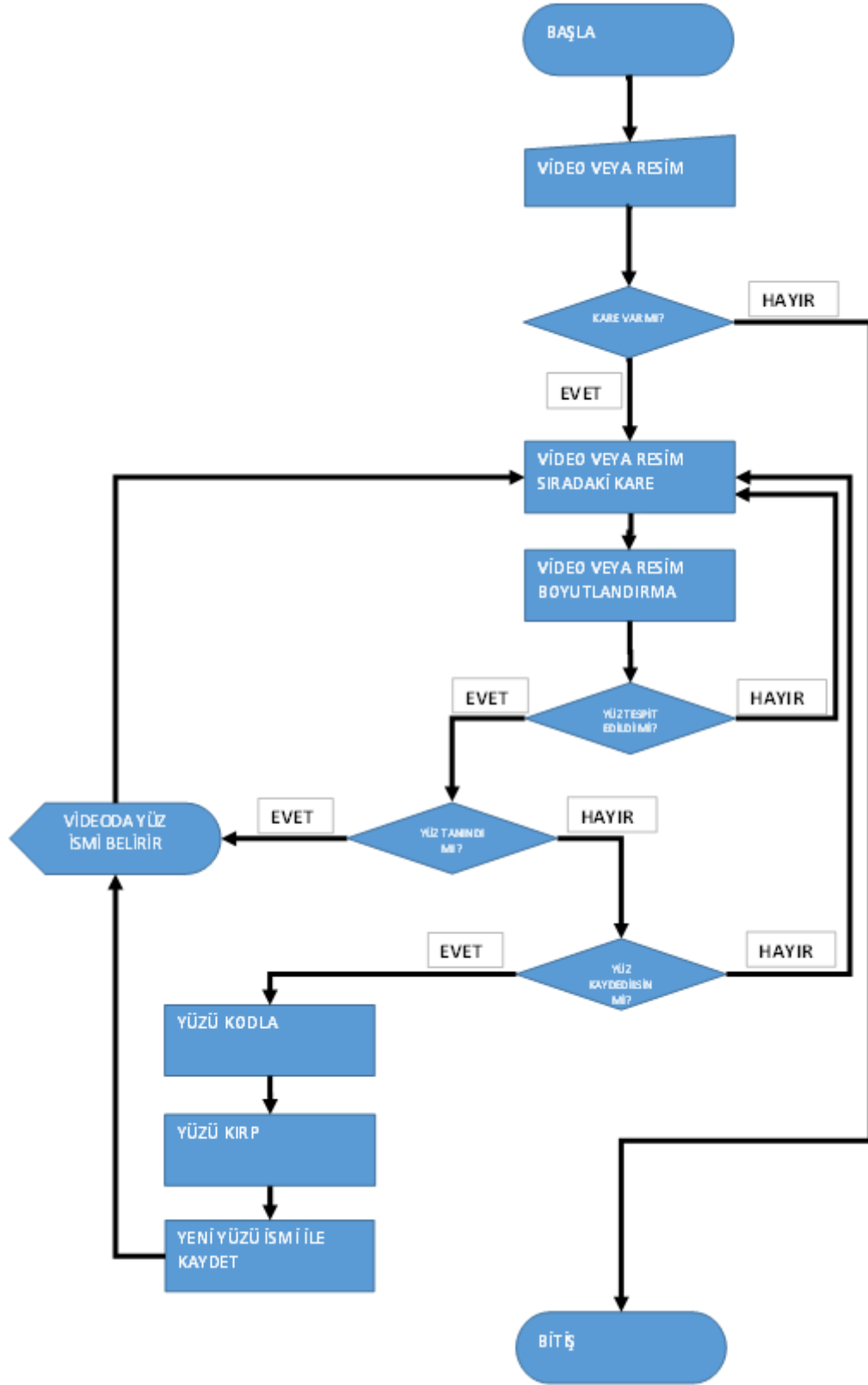
```
Import face_recognition
```

```
Known_image = face_recognition.load_image_file("mehmet.jpg")  
  
Unknown_image = face_recognition.load_image_file("bilinmeyen.jpg")  
  
Mehmet_encode = face_recognition.face_encodings(known_image)[0]  
  
Unknown_encode = face_recognition.face_encodings(unknown_image)[0]  
  
Results = face_recognition.compare_faces([mehmet_encode],unknown_encode)
```

Gerçek zamanlı yüz tanıma uygulamaları yapabilmek için diğer kütüphanelerin yanında da bu uygulama kullanılabilir.

3. GELİŞTİRİLEN PROGRAM

3.1 Akış Şeması



Şekil 3.1.1 Programın akış şeması

Çalışılmış olan programda video kullanılmıştır. Videoyu giriş olarak alan program videonun bütün karelerini gezerek tekrar boyutlandırma işlemi yapmaktadır. Tekrar boyutlandırma işleminin yapılmasının sebebi küçük boyutlu görsellerde çalışmanın performansa olan etkisidir. İşlenen görüntülerde dlib kütüphanesinin frontal_face detector' ü kullanılarak yüzler tespit edilir. Yüzlerin tespit edildiği kare kodlanır (encode) ve yüz karesi kesilir. Tespit edilmiş yüz ile daha önce tespit edilmiş yüzler karşılaştırılır. Bu karşılaştırma dlib kütüphanesinin face_compare fonksiyonunda çalışır. Tespit edilen yüzü sistem daha önce görmediyse yüzü eklemek isteyip istemediğimizi soracaktır. Verilecek evet veya hayır cevabına istinaden işlemler devam eder. Verilen evet cevabından sonraki adım yüz için kullanılacak olan isimdir. Yüze isim verildikten sonra sistem, karşısına çıkan aynı yüzleri tanır. Tespit edilen yabancı yüzde bütün işlemler tekrar eder.

Ubuntu 20.04 LTS işletim sistemi kullanılmıştır.

Ubuntu işletim sistemi kurulduktan sonra Anaconda Linux sürümü kurulumu için aşağıdaki adımlar izlenir.

1. `sudo apt-get update` -- Paket yöneticisi güncellenir
2. `sudo apt-get install curl` -- Curl Linux paketlerini indirmemize yarar.
3. `cd /tmp` -- tmp klasörüne gidilir.
4. `curl -O https://repo.anaconda.com/archive/Anaconda3-2020.02-Linux-x86_64.sh` -- Anaconda paketi indirilir.
5. `bash Anaconda3-2020.02-Linux-x86_64.sh` -- Anaconda kurulumu başlatılır.
6. `source ~/.bashrc` -- Anaconda aktif hale getirilir.
7. `conda create --name test_environment python=3` -- Anaconda ortamı kurulur.
8. `conda activate test_environment` -- Kurulan ortam aktifleştirilebilir.

Anaconda kurulumu yapıldıktan sonra CPU'ya yardımcı olacak olan CUDA paralel programlama platformunun kurulması performansa etki edecektir. Aşağıdaki komutlar ile CUDA için gerekli paketler indirilir.

1. `sudo apt-get update`
2. `sudo apt-get upgrade -ysudo apt-get install -y buildessential cmake unzip pkg-config`
3. `sudo apt-get install -y libxmu-dev libxi-dev libglu1-mesa`
4. `libglu1-mesa-dev`
5. `sudo apt-get install -y libjpeg-dev libpng-dev libtiff-dev`
6. `sudo apt-get install -y libavcodec-dev libavformat-dev`
7. `libswscale-dev libv4l-dev`
8. `sudo apt-get install -y libxvidcore-dev libx264-dev`
9. `sudo apt-get install -y libgtk-3-dev`
10. `sudo apt-get install -y libopenblas-dev libatlas-base-dev`
11. `liblapack-dev gfortran`
12. `sudo apt-get install -y libhdf5-serial-dev graphviz`
13. `sudo apt-get install -y python3-dev python3-tk pythonimaging-tk`
14. `sudo apt-get install -y linux-image-generic linux-imageextra-virtual`
15. `sudo apt-get install -y linux-source linux-headers-generic`
16. `sudo apt-get install nvidia-driver-460 -- Nvidia driver indirilir, sürüm her ekran kartına göre farklı olabilir. Sahip olduğunuz ekran kartına uygun driver indirilmelidir.`
17. `nvidia-smi -- Kurulumlar ve sürücü kontrol edilir.`

18. wget

https://developer.download.nvidia.com/compute/cuda/11.2.2/local_installers/cuda_11.2.2_460.32.03_linux.run -- CUDA kurulum paketi indirilir.

19. sudo sh cuda_11.2.2_460.32.03_linux.run -- Cuda kurulur.

20. sudo nano ~/.bashrc

```
export PATH=/usr/local/cuda-11.2/bin${PATH:+:${PATH}}
```

```
export
```

```
LD_LIBRARY_PATH=/usr/local/cuda11.2/lib64${LD_LIBRARY_PATH:+:${LD_LIBRARY_PATH}}
```

```
export CUDA_HOME=/usr/local/cuda -- Ortam deęişkenleri ayarlanır.
```

21. tar -zvx cudnn-11.2-linux-x64-v8.1.0.77.tgz

```
cd ~
```

```
sudo gedit .bashrc
```

```
export LD_LIBRARY_PATH=xxx/cuda/lib64:$LD_LIBRARY_PATH
```

```
source .bashrc
```

```
cd xxx/cuda/include
```

```
sudo cp *.h /usr/local/cuda/include/
```

```
sudo chmod a+r /usr/local/cuda/include/cudnn.h -- Cudnn kurulur ve ortam deęişkenleri ayarlanır.
```

22. nvidia-smi

```
nvcc -V – Kurulumlar kontrol edilir.
```

CUDA kurulumlarından sonra Dlib kütüphanesinin kurulması gereklidir. Kurulum için aşağıdaki adımlar izlenir.

1. mkdir -p /opt/SP/packages/

2. cd \$_
3. git clone <https://github.com/davisking/dlib.git>
4. cd dlib
5. git submodule init
6. git submodule update
7. cmake -DDLIB_USE_CUDA=1 -DUSE_AVX_INSTRUCTIONS=1 ../
8. cmake --build . --config Release
9. cd ../
10. python setup.py install

Sırayla yukarıda ki komutlar uygulanarak kurulum yapılabilir.

Komut satırında aşağıdaki komutlar yazılarak kurulumun başarısı kontrol edilir. Çıktılar aşağıdaki gibi olmalıdır.

```
>>> import dlib.cuda as cuda;

>>> print(cuda.get_num_devices());

1

>>> import dlib

>>> dlib.DLIB_USE_BLAS

True

>>> dlib.DLIB_USE_CUDA

True

>>> dlib.DLIB_USE_LAPACK
```

```
True
```

Son olarak kullanacağımız face_recognition kütüphanesini kurmak için aşağıdaki komut yazılmalıdır.

1. pip3 install face_recognition -- Pip ile kütüphane indirilebilir.

Face_recognition kütüphanesi dlib kütüphanesini kullandığından dolayı dlib kütüphanesinin kurulumu önceden yapılmıştır (Ageitgey, 2021).

Kurulumlardan sonra conda environment içince main.py dosyası oluşturup programın kodları yazılmaya başlanır.

3.2 Kaynak Kod

Projemizin ana kodunda yaptığımız ilk işlem kütüphaneleri eklemek oldu.

```
import face_recognition
import cv2
import numpy as np
```

OpenCV (Open Source Computer Vision) açık kaynak kodlu görüntü işleme kütüphanesidir. NumPy, dizilerle çalışmak için kullanılan bir Python kütüphanesidir. Ayrıca doğrusal cebir, fourier dönüşümü ve matrisler alanında çalışmak için de gerekli işlemlere sahiptir. Glob modülü, Python'da belirli bir klasör içindeki dosyaları listelememize yardımcı olan bir modüldür.

```
video_capture = cv2.VideoCapture("test5.mp4")
```

Bu satır ile üzerinde test yapacağımız videoyu opencv kütüphanesini kullanarak okuyoruz.

```
#process_this_frame = True
new_person = False
start = True
ask_counter = 0
crop_img = None
```

Kod bloğunda daha sonra kullanacağımız bazı değişkenleri tanımlarız. Bu değişkenlerin karşılaşılabilecek durumlara göre değerleri değiştirilerek istenilen sonuçlar gözlemlenmektedir.

```
while True:
    if new_person == True or start == True:
        i = 1
        face_images_dict = {}
        face_encodings_dict = {}
        known_face_names = []
        known_face_encodings = []
        with open("names.txt") as file:
            for item in file:
                item = item.rstrip('\n')
                known_face_names.append(item)
            file.close()
        try:
            for name in known_face_names:
                face_images_dict["{0}".format(name)] =
                face_recognition.load_image_file("faces/{0}.jpg".format(name))
                face_encodings_dict["{0}".format(name)] =
                face_recognition.face_encodings(face_images_dict["{0}".format(name)])[0]
                known_face_encodings.append(face_encodings_dict["{0}".format(name)])
        except:
            pass
        # Initialize some variables
        face_locations = []
        face_encodings = []
```

```
face_names = []
new_person = False
start = False
```

Programın devamında bir döngü içerisinde test videosu içerisinde daha önce tanımlanmamış bir yüz bulunup bulunmadığı kontrol edilmektedir. Eğer tanımlanmamış bir yüz var ise kişinin ismi listeye eklenmektedir. Bu kısım sadece “new_person” veya “start” değişkenleri True değerini aldığı için çalıştığı için başlangıçta çalışmayacaktır. Video okunup tanıma işlemi yapıldıktan sonra işleme alınır.

```
# Grab a single frame of video
ret, frame = video_capture.read()
frame = cv2.resize(frame, (1280, 720), interpolation=cv2.INTER_AREA)
original_frame = frame
# Resize frame of video to 1/4 size for faster face recognition processing
small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)
# Convert the image from BGR color (which OpenCV uses) to RGB color (which
face_recognition
uses)
rgb_small_frame = small_frame[:, :, ::-1]
```

Bu kısımda video okunur, yeniden boyutlandırılır ve renk formatı opencv kullanımına uygun hale getirilir.

```
# Find all the faces and face encodings in the current frame of video
face_locations = face_recognition.face_locations(rgb_small_frame)
face_encodings = face_recognition.face_encodings(rgb_small_frame,
face_locations)
try:
    for top, right, bottom, left in face_locations:
        # Draw a box around the face
        #cv2.rectangle(img, (left, top), (right, bottom), (0, 0, 255), 2)
        crop_img = rgb_small_frame[top:bottom, left:right]
```

```
except:  
    pass
```

Bir önceki aşamada videodan okunan ve hazırlanan görüntü karesinde face recognition kütüphanesi kullanılarak tanıma işlemi yapılır. Face recognition kütüphanesi yüz tanıma işlemi için “dlib” kullanır. Dlib hakkında detaylı açıklama rapor içerisinde yapılmıştır. Face recognition kütüphanesi kullanılarak tespit edilen yüzün koordinatları kullanılarak yüzün olduğu bölgeyi belirtmek için bir kare çizilir.

```
face_names = []  
for face_encoding in face_encodings:  
    # See if the face is a match for the known face(s)  
    matches = face_recognition.compare_faces(known_face_encodings,  
        face_encoding)  
    name = "Unknown"  
    ## If a match was found in known_face_encodings, just use the first one.  
    # if True in matches:  
    # first_match_index = matches.index(True)  
    # name = known_face_names[first_match_index]  
    # Or instead, use the known face with the smallest distance to the new face  
    face_distances = face_recognition.face_distance(known_face_encodings,  
        face_encoding)  
    try:  
        best_match_index = np.argmin(face_distances)  
        if matches[best_match_index]:  
            name = known_face_names[best_match_index]  
            name = ".join([i for i in name if not i.isdigit()])  
    except:  
        pass
```

Tespit edilen yüzün sistemde daha önce tanımlanmış bir yüz olup olmadığını bu kısımda kontrol ederiz. Face recognition kütüphanesinin “compare_faces” fonksiyonu kullanılarak sisteme kayıtlar olan yüzler ve yeni bulunan yüz karşılaştırılır. Yüzler arasındaki benzerlik

oranı 0.5(default olan değer değiştirilebilir) ten fazlaysa yüzlerin aynı olduğuna karar verilir. Eğer tespit edilen yüz tanınmayan bir yüz ise başlangıç olarak ismi “unknown” olarak tanımlanır.

```
if name == "Unknown":
    ask_counter += 1
    if ask_counter >= 15 and len(face_locations)==1:
        cv2.circle(original_frame, ((left+((right-left)//2))*4,(top+((bottom-top)//2))*4), 2,
(0, 255, 0), 1)
        question = input("Do you want to add new person to system? : ")
        if question == ("yes"):
            print("Please select the face")
            roi = cv2.selectROI(original_frame)
            roi_cropped = original_frame[int(roi[1]):int(roi[1]+roi[3]),
int(roi[0]):int(roi[0]+roi[2])]
            cv2.destroyWindow("ROI selector")
            name_file = open('names.txt', 'r')
            name_list = name_file.readlines()
            print(name_list)
            name_file.close()

            new_name = input("Please enter new person name: ")
            if new_name+"\n" not in name_list:
                name_file = open('names.txt', 'a')
                name_file.write(new_name)
                name_file.write("\n")
                name_file.close()
                new_person = True
                cv2.imwrite("faces/{}.jpg".format(new_name),roi_cropped)
                ask_counter = 0
            elif question == ("no"):
                ask_counter = 0
            face_names.append(name)
```

Bu kısımda sadece tespit edilen yüz tanınmayan bir yüz ise aktif hale gelir. Eğer tespit edilen yüz tanınmıyorsa kullanıcıya tespit edilen yüzü sisteme kayıt etmek isteyip istemediği sorulur. Kullanıcı tespit edilen yüzü sisteme kayıt etmek isterse tespit edilen yüze ait isim bilgisini girer. Daha sonra açılan pencerede tespit edilen yüzü kare içerisine alarak kayıt eder. Bundan sonraki süreçte kayıt edilen yüz tanımlanan isim ile gösterilir.

```
# Display the results
for (top, right, bottom, left), name in zip(face_locations, face_names):
    # Scale back up face locations since the frame we detected in was scaled to 1/4 size
    top *= 4
    right *= 4
    bottom *= 4
    left *= 4
    # Draw a box around the face
    cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)
    # Draw a label with a name below the face
    cv2.rectangle(frame, (left, bottom - 35), (right, bottom), (0, 0, 255), cv2.FILLED)
    font = cv2.FONT_HERSHEY_DUPLEX
    cv2.putText(frame, name, (left + 6, bottom - 6), font, 1.0, (255, 255, 255), 1)
    # Display the resulting image
    cv2.imshow('Video', frame)
    # Hit 'q' on the keyboard to quit!
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
    # Release handle to the webcam
    video_capture.release()
cv2.destroyAllWindows()
```

Son olarak bu kısımda sonuçlarımızı göstermek için opencv kütüphanesini kullanırız. Elde ettiğimiz anlık sonuçlar ekrana basarız.

4. SONUÇ VE ÖNERİLER

Yazılan programda hareketli görüntülerde bütün kareler yorumlanarak yüz tespitleri yapılmaya çalışılmıştır. Yapay sinir ağları yöntemini kullanarak yüzler face_recognition kütüphanesi ile tespit edilmiştir. Program tespit edilen yüzleri kırpma imkanı verdiği için dolayı anlamsız görüntülerin kırpılmasının önüne geçilebilir. Görüntünün karesinde kırpılan yüzlerden sonra kırpılan yüzün adlandırılması istenir. Böylelikle kaydedilmiş olan yüzler daha sonraki tanıma işlemleri için kullanılmaya hazır halde bekleyebilecektir.

Yüz tanıma uygulamalarının sağladığı avantajlar gibi dezavantajları da vardır. Günlük hayatta kişinin kişisel alanı söz konusu olduğunda, bu alanı korumanın kişinin hayat konforuna kesinlikle iyi etki edeceği gerçektir. Bu durum söz konusu olduğunda kişi kendine ait hesaplar, projeler, çalışmaları kendi yetkisi dâhilinde görebilmek için bazı engeller koyması gerekmektedir. Çoğunlukla koyulan bu engeller zaman içinde bir zaman kaybı olarak görülmeye başlanacaktır. Bu oluşan zaman kaybını engellemek için biyometrik kimlik doğrulama yöntemleri çok doğru bir seçim olacaktır. Sıradan bir kişi için biyometrik kimlik doğrulaması çok teknik olabilir fakat işleyişte çok iyi bir konfor sağlayacaktır. Örneğin bir hesabın birden fazla özel karakterden oluşan parolasının yazılması zaman alırken biyometrik yöntemler ile saniyeler sürecektir.

Başka bir açıdan güvenliğe en büyük desteği kolluk kuvvetlerinin uygulamalarında ki kişi karşılaştırması olarak görülebilir. Kolluk kuvvetlerinin suçlu, şüpheli sınıfında bulunan kişileri daha hızlı tespit ettikleri yüz tanıma sisteminin getirdiği kolaylıklardan biridir.

Geliştirilmiş olan programda, program yüzü tespit ettikten sonra kişinin kayıt altına alınıp alınmayacağını sorguluyor ve evet cevabını aldığı anda ismi ve yüz çevresinin programı kullanan tarafından belirlenmesine olanak sağlıyor. Belirlenen yüz bir klasöre isim ile beraber yazılıyor. Bu uygulama ile beraber işletmelerin personel takip sistemleri, kolluk kuvvetlerinin suçlu veritabanının genişletilmesi, hükümetlerin nüfus ve vatandaşlık işlerinin hızlandırılması, seyahat firmalarının yolculardan oluşan veritabanları hazırlaması ve bunun gibi birçok işlemin yapılıp kontrol noktalarında zamandan tasarruf sağlanabilir.

Yüz tanıma sistemlerinin doğru şekilde kullanımı bütün sektörler için yararlı olabileceği gibi toplanmış olan yüz verisinin aksi yönde kullanımı da insanın özgürlüğünü kısıtlayan kolay bir yöntem olarak da rahatsızlık verebilir. Yüz tanıma sisteminin popülerliği arttıkça, yasalar ile desteklenmesi, toplanan verinin doğru kullanımına yol açacağı bir gerçektir.

Çalışmanın devamında bir videoda bulunan yüzlerin listesini yapıp bu yüzler arasında istenmeyen veya şüpheli yüzleri tespit etmeye yönelik çalışmalara devam edilecektir.

KAYNAKLAR

IBM (2022). What is computer vision? <https://www.ibm.com/topics/computer-vision#citation2>

Demush, R. (2019). A Brief History of Computer Vision (and Convolutional Neural Networks). <https://hackernoon.com/a-brief-history-of-computer-vision-and-convolutional-neural-networks-8fe8aacc79f3>

IBM Cloud Education (2022). Optical character recognition. <https://www.ibm.com/cloud/blog/optical-character-recognition>

Great Learning Team (2011). Introduction to Image Processing. <https://www.mygreatlearning.com/blog/introduction-to-image-processing-what-is-image-processing/>

Kaspersky Lab., (2022). KasperskyWhat is Facial Recognition – Definition and Explanation. <https://www.kaspersky.com/resource-center/definitions/what-is-facial-recognition>

Bahar, M.S. & Buluş, E.,(2021). Derin öğrenme teknikleri kullanılarak yüz fotoğrafları içeren sahte fotoğraf ve video sentezi.

Sirovich, L., & Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *Josa a*, 4(3), 519-524.

Kirby, M., & Sirovich, L. (1990). Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern analysis and Machine intelligence*, 12(1), 103-108.

Kekül, H. , Bircan, H. & Arslan, H. (2018). YÜZ TANIMA UYGULAMALARINDA ÖZYÜZLER VE YAPAY SİNİR AĞLARININ KARŞILAŞTIRILMASI . *Uluslararası Yönetim Bilişim Sistemleri ve Bilgisayar Bilimleri Dergisi* , 2 (1) , 51-59 . Retrieved from <https://dergipark.org.tr/tr/pub/uybisbbd/issue/37798/430914>

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1), 71-86.

Pentland, A., Moghaddam, B., & Starner, T. (1994). View-based and modular eigenspaces for face recognition.

Chang, K., Bowyer, K. W., Sarkar, S., & Victor, B. (2003). Comparison and combination of ear and face images in appearance-based biometrics. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9), 1160-1165.

- Verlinde, P., & Mayoraz, E. (1998). Decision fusion using a multi-linear classifier.
- Buluş, E., UBMK'2021, "Gender determination from pictures with CNN models"(online), Ankara-TÜRKİYE, September 2021
- Stonham, T. J. (1986). Practical face recognition and verification with WISARD. In Aspects of face processing (pp. 426-441). Springer, Dordrecht.
- Davari, P., & Naimi, H. M. (2008, January). A New Face Recognition System-Using HMMs along with SVD Coefficients. In International Conference on Computer Vision Theory and Applications (Vol. 2, pp. 200-205). SCITEPRESS.
- Lawrence, S., Giles, C. L., Tsoi, A. C., & Back, A. D. (1997). Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1), 98-113.
- Weng, J. J., Ahuja, N., & Huang, T. S. (1993, May). Learning recognition and segmentation of 3-D objects from 2-D images. In 1993 (4th) International Conference on Computer Vision (pp. 121-128). IEEE.
- Lin, S. H., Kung, S. Y., & Lin, L. J. (1997). Face recognition/detection by probabilistic decision-based neural network. *IEEE transactions on neural networks*, 8(1), 114-132.
- Lades, M., Vorbruggen, J. C., Buhmann, J., Lange, J., Von Der Malsburg, C., Wurtz, R. P., & Konen, W. (1993). Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on computers*, 42(3), 300-311.
- Wiskott, L., & Von Der Malsburg, C. (1996). Recognizing faces by dynamic link matching. *Neuroimage*, 4(3), S14-S18.
- Samaria, F., & Fallside, F. (1993). Face identification and feature extraction using hidden markov models (pp. 295-298). Olivetti Research Limited.
- Samaria, F. S., & Harter, A. C. (1994, December). Parameterisation of a stochastic model for human face identification. In Proceedings of 1994 IEEE workshop on applications of computer vision (pp. 138-142). IEEE.
- Tamura, S., Kawai, H., & Mitsumoto, H. (1996). Male/female identification from 8×6 very low resolution face images by neural network. *Pattern recognition*, 29(2), 331-335.
- Benitez-Garcia, G. , Nakamura, T. and Kaneko, M. (2017) Facial Expression Recognition Based on Local Fourier Coefficients and Facial Fourier Descriptors. *Journal of Signal and Information Processing*, 8, 132-151.
- Kanade, T. (1974). Picture processing system by computer complex and recognition of human faces.
- Goldstein, A. J., Harmon, L. D., & Lesk, A. B. (1971). Identification of human faces. *Proceedings of the IEEE*, 59(5), 748-760.
- Kaya, Y., & Kobayashi, K. (1972). A basic study on human face recognition. In *Frontiers of pattern recognition* (pp. 265-289). Academic Press.

- Brunelli, R., & Poggio, T. (1993). Face recognition: Features versus templates. *IEEE transactions on pattern analysis and machine intelligence*, 15(10), 1042-1052.
- Cox, I. J., Ghosn, J., & Yianios, P. N. (1996). Feature-based face recognition using mixture-distance, *Computer Vision and Pattern Recognition*.
- Chellappa, R., Manjunath, B. S., & Malsburg, C. V. D. (1992). A feature based approach to face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 373-378).
- Baron, R. J. (1981). Mechanisms of human facial recognition. *International Journal of Man-Machine Studies*, 15(2), 137-178.
- Vetter, T., & Poggio, T. (1997). Linear object classes and image synthesis from a single example image. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7), 733-742.
- Beymer, D., & Poggio, T. (1995). Face recognition from one model view. In *Proc. Fifth Int'l Conf. Computer Vision*.
- Zhao, W. Y., & Chellappa, R. (2000, March). SFS based view synthesis for robust face recognition. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition* (Cat. No. PR00580) (pp. 285-292). IEEE.
- Blanz, V., & Vetter, T. (2003). Face recognition based on fitting a 3d morphable model. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9), 1063-1074.
- Anaconda (2022). Anaconda Documentation. <https://docs.anaconda.com/>
- Umangshrestha (2021). Introduction to CUDA. <https://medium.com/geekculture/introduction-to-cuda-7bf6909ea57c>
- Dlib (2015). Dlib Introduction. <http://dlib.net/intro.html>
- OpenCV (2022). Open Source Computer Vision. <https://docs.opencv.org/>
- Viola, P., & Jones, M. (2001, December). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001* (Vol. 1, pp. I-I). Ieee.
- Darshan A. (2019). What are Haar Features used in Face Detection ? <https://medium.com/analytics-vidhya/what-is-haar-features-used-in-face-detection-a7e531c8332b>
- Pypi.org (2020). Face-recognition 1.3.0. <https://pypi.org/project/face-recognition/>
- Mahedi-61 (2022). Cuda 11.2 installation on Ubuntu 20.04. <https://gist.github.com/Mahedi-61/2a2f1579d4271717d421065168ce6a73>
- Ageitgey (2021). Face_recognition github repository. https://github.com/ageitgey/face_recognition