



T.C.
SELÇUK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ORB Özelliklerine Dayalı Yüz Tanıma Sistemi

Abdullah Mohammed NOORI

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Mart-2019
KONYA
Her Hakkı Saklıdır

ÖZET

YÜKSEK LİSANS TEZİ

ORB ÖZELLİKLERİNE DAYALI YÜZ TANIMA SİSTEMİ

Abdullah Mohammed NOORI

Selçuk Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr.Öğr. Üyesi Murat SELEK

2019,70 Sayfa

Jüri
Dr. Öğr. Üyesi Murat SELEK
Prof. Dr. Fatih BAŞÇİFTÇİ
Doç. Dr. Halife KODAZ

Yüz tanıma, içinde barındırdığı ışıklandırma, ölçek değişkeni, döndürme değişkeni, çakışma alanları gibi karmaşık konular nedeniyle araştırmaya açık bir husus olma özelliğine sahiptir. Bu nedenle çalışmada, yüz tanıma amacıyla kullanılan Yönelimli FAST ve Döndürülmüş BRIEF (Oriented FAST and Rotated BRIEF-ORB) algoritması, Temel Bileşen Analizi (Principal Component Analysis-PCA) ve Yerel İkili Örüntü (Local Binary Patterns-LBP) teknikleri gibi metotlar kullanılmıştır. Elde edilen sonuçlar mukayese edilerek hangi tekniğin daha başarılı olduğu ortaya konulmaya çalışılmıştır.

Bu amaçla, 30 kişiden oluşan bir veri seti oluşturulmuştur. Yüz tanıma aşamasında birbirinden farklı durumları da göz önüne alabilmek için 2 farklı yüz detayı, 3 farklı ışık seviyesi ve 4 farklı mesafe dikkate alınmıştır. Her bir farklı durum için her bir kişinin 5 farklı açıdan görüntüleri çekilmiştir. Bunun sonucunda toplamda 3600 görüntüden oluşan bir veri seti elde edilmiştir. Bu görüntülere ORB, PCA ve LBP algoritmaları uygulanarak yüz tanıma testleri yapılmıştır. Bu testlerin sonucunda ORB tekniğinin diğer metotlara göre %86.29'lük bir başarı ile daha yüksek bir yüz tanıma oranına sahip olduğu tespit edilmiştir.

Anahtar Kelimeler: : FAST köşe noktaları, Yüz tanıma, Özellik çıkarımı, LBP algoritması, ORB algoritması, PCA algoritması.

ABSTRACT

MS THESIS

FACE RECOGNITION SYSTEM BASED ON ORB FEATURES

Abdullah Mohammed NOORI

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF SELÇUK UNIVERSITY
THE DEGREE OF MASTER OF SCIENCE OF PHILOSOPHY
IN COMPUTER ENGİNEERİNG

Advisor: Assit. Prof. Dr. Murat SELEK

2019,70 Pages

Jury
Assit. Prof. Dr. Murat SELEK
Prof. Dr. Fatih BAŞÇİFTÇİ
Assoc. Prof. Dr. Halife KODAZ

Face recognition has the feature of being open to research due to complex issues such as illumination, scale variant, rotation variant, and collision areas. For this reason, in our study, we applied new methods on the images for the purpose of face recognition such as ORB (Oriented FAST and Rotated BRIEF) algorithm, PCA (Principal component analysis) and LPB (Local binary patterns) techniques. We aimed to identify the more successful ones by comparing the results obtained.

For this purpose, a 30-person database set has created in this study. In order to be able to take into account different situations in the face recognition phase, 2 different facial details, 3 different light levels and 4 different distances have been taken into account. For that purpose, images of each person were taken from 5 different angles for each different situation. Consequently, a database consisting of a total 3600 images has obtained. Face recognition tests have been performed on these images by applying ORB, PCA and LBP algorithms. As a result of these tests, ORB has been found to have a higher face recognition rate with a 86.29% success rate than the others.

Keywords: FAST corner points, Face recognition, Feature extraction, LBP algorithm, ORB algorithm, PCA algorithm.

ÖNSÖZ

Bu tez çalışmasında ORB özelliklerine dayalı bir yüz tanıma sisteminin tasarlanması ve uygulanması gerçekleştirilmiştir. Çalışma donanım ve yazılım olarak tasarlanmış ve bir kapı geçiş sisteminde yüz tanımaya dayalı geçiş kontrolü olarak gerçekleştirilmiştir.

Çalışmamda bana her türlü yardımcı olan, bilgilerinden faydalandığım, seçtiğim konu hakkında bana desteğini esirgemeyen değerli danışman hocam Sayın Dr. Öğr. Üyesi Murat SELEK'e teşekkürlerimi sunarım. Tüm bu çalışmalar süresinde bana yardımcı olan, her türlü destek ve moral veren aileme ve özellikle canım annem'e FATIMAH A.ALHAMID, beni her zaman ve her konuda desteklediği için teşekkürlerimi sunuyorum.

Abdullah Mohammed NOORI
KONYA-2019

İÇİNDEKİLER

ÖZET ..... **i**
ABSTRACT ..... **ii**
ÖNSÖZ ..... **iii**
İÇİNDEKİLER ..... **iv**
SİMGELER VE KISALTMALAR ..... **vi**
1. GİRİŞ **1**
2. KAYNAK ARAŞTIRMASI **3**
2.1. Özyüzler 3
2.2. Sinir Ağları 5
2.3. Grafik Eşleme 6
2.4. Gizli Markov Modelleri 7
2.5. Geometrik Özellik Eşleştirme 7
2.6. Şablon eşleme 9
2.7. 3B morphable model 9
2.8. Hat Kenar Haritası 10
2.9. Destek Vektör Makinesi (SVM) 13
3. MATERYAL VE YÖNTEM **22**
3.1. Malzemeler 22
3.1.1. Lattepanda Bilgisayar Modülü 22
3.1.2. Kamera 23
3.1.3. Ekran 24
3.1.4. 3B baskılı Muhafaza 24
3.1.5. Sistemin Diğer Donanımları 24
3.2 Yöntem 25
3.2.1. Yönelimli FAST ve Döndürülmüş BRIEF (Oriented Fast and Rotated Brief-ORB) 25
3.2.2. Temel bileşenler analizi (PCA) 30
3.2.3. Yerel ikili örüntü (LBP) 33
3.2.4. Yüz veri seti 37
3.2.5 Linde–Buzo–Gray algoritması (LBG) 40
3.2.6 K-en yakın komşu (KNN) 41
3.2.7 Haar Cascade sınıflandırıcı 42

3.2.8. Önerilen sistem 43
4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA 50
5. SONUÇLAR VE ÖNERİLER 57
5.1 Sonuçlar 57
5.2 Öneriler 58
KAYNAKLAR 60
EKLER 64
ÖZGEÇMİŞ 70

SİMGELER VE KISALTMALAR

Simgeler

X	: Veri Matrisi
W	: Katsayılar Vektörü
XTX	: X Kovaryans Matrisi
Q	: İki Farklı Bileşen Arasındaki Kovaryans
L	: İlk Temel Bileşen
∑	: Sayı dizisinin toplamını gösteren sembol
Hi	: Yerel İkili Örüntü Histogram
Ni	: Normalize Edilmiş Hi Histogramı
I_P	: İlgi noktası

Kısaltmalar

3B	: 3 Boyutlu
AWGN	: Toplanır Beyaz Gausian Gürültüsü (Additive White Gaussian Noise)
ANN	: Yapay Sinir Ağları (Artificial Neural Networks)
BOVW	: Görsel Kelime Çantası (Bag Of Visual Words)
BRIEF	: İkili Gürbüz Bağımsız Temel Özellikler (Binary Robust Independent Elementary Features)
CCA	: Kanonik Korelasyon Analizi (Canonical Correlation Analysis)
DBNN	: Karar Tabanlı Sinir Ağları (Decision-Based Neural Networks)
EBCM	: Elastik Demet Grafik Eşleme (Elastic Bundle Chart Matching)
EOF	: Deneysel Ortogonal Fonksiyonlar (Empirical Orthogonal Functions)
EVD	: Eigenvalue Ayrışımı (Decomposition)
FAST	: Hızlandırılmış Segment Testi Özellikleri (Features from Accelerated Segment Test)
FERET	: Yüz Tanıma Teknolojisi (Face Recognition Technology)
HMMs	: Gizli Markov Modelleri (Hidden Markov Model)

ICA : Bağımsız Bileşen Analizi (Independent Component Analysis)
IPS : Düzlem İçi Anahtarlama (In Plane Switching)
KNN : K- En Yakın Komşu (K-Nearest Neighbors)
KLT : Karhunen–Loève Dönüşümü (Karhunen–Loève Transformation)
LBP : Yerel İkili Örüntü (Local Binary Patterns)
LD : Doğrusal Ayırma (Linear Discriminant)
LBG : Linde–Buzo–Gray
LEM : Hat Kenar Haritası (Line Edge Map)
NCC : En Yakın Merkez Sınıflandırması (Nearest Center Classification)
ORB : Yönelimli FAST ve Döndürülmüş BRIEF (Oriented FAST and Rotated BRIEF)
O-PWC : Optimal İkili Bağlanma Sınıflandırıcısı (Optimal Pairwise Coupling Classifier)
ORL : Olivetti Araştırma Laboratuvarı (Olivetti Research Laboratory)
PCA : Temel Bileşenler Analizi (Principal Component Analysis)
PDBNN : Kalan olasılıksal karar temelli Sinir Ağı (Probabilistic Decision-Based Neural Networks)
POD : Uygun Ortogonal Ayrıştırma (Proper Orthogonal Decomposition)
QP : Kuadratik Programlama (Quadratic Programming)
RAM : Rastgele Erişimli Hafıza (Random Access Memory)
RBF : Radyal Tabanlı Fonksiyon (Radial Basis Functions)
SIFT : Ölçek Değişmezliği Özellik Dönüşümü (Scale Invariant Feature Transform)
SOM : Kendi Kendini Organize Eden Harita (Self-Organizing Map)
SRM : Yapısal Risk Azaltma (Structural Risk Minimization)
SURF : Hızlandırılmış Gürbüz Özellikler (Speeded Up Robust Features)
SVD : Tekil Değer Ayrıştırma (Singular Value Decomposition)
SVM : Destek Vektör Makinesi (Support Vector Machine)

1. GİRİŞ

Akıllı davranış biçimlerinin en önemlilerinden biri de şüphesiz model tanımlama olup, bunu hem yapay, hem de biyolojik sistemlerle uygulamak mümkündür. Model tanımlama, makinelerin etrafında olup bitenleri nasıl anlamlandırdığını tespit etme ve geçmişlerinden yola çıkarak ilgi sahalarını öğrenme tekniği olup, bu modelleri sınıflandırmak için anlamlı ve makul kararlar verilebilmesi sürecidir. Bu süreç veri toplama, öznitelik bulma, model sınıflandırma ve performans değerlendirme şeklinde dört basamaktan oluşmaktadır. Bazı uygulamalarda, söz konusu model bir parmak izi, bir yüz ya da bir konuşma sinyali olabilmektedir. Model tanımlama, örneklerin eşleştirilmesi, istatiksel model tanımlama, yapay sinir ağları, sözdizimsel model tanımla vb. şekillerde uygulanabilir.

Yüzleri sınıflandırma alanındaki en eski teknik Francis Galton tarafından geliştirilmiştir. Yazar 1888 yılında yüz profillerini bir eğri olarak toplamış, ardından araştırmacılar yüz modellerini hesaplamış ve modelden sapmalara göre sınıflandırma yapmışlardır. Bu sınıflandırma, veri tabanında diğerleriyle karşılaştırılabilecek bir hesaplama vektörü elde edilmesini sağlamıştır (Galton, 1878).

İlk, yarı otomatik yüz tanıma sistemi 1960'lı (Ballantyne ve ark., 1996) yıllarda geliştirilmiştir; bu metot görüntülerde yer alan yüzlerin göz, burun, kulak ve ağız gibi bazı karakteristik niteliklerinin saptanması ilkesine dayanıyordu. Sonrasında, belirli noktalar ve işaretleyiciler arasındaki mesafeler hesaplanmıştır. 1989 yılına gelindiğinde, Kohonen (Kohonen, 1989) tarafından standart hale getirilen yüzleri tanımlamak için bir sinir ağı geliştirilmiştir; 1990 yılında Kirby ve Sirovich, özgün yüzleri saptamak için cebirsel hesaplama tekniğini kullandılar (Kirby ve Sirovich, 1990). Bu alanda en dikkat çekici çalışma, 1999 yılında W. Zaho tarafından geliştirilmiştir. Zaho (Zhao ve ark., 2003), yüz tanıma sistemlerini gerçek dünya düzlemine uyarlamayı başarmıştır. Günümüze gelince, Borra Surekha ve Kanchan Jayant Nazare, 2016 yılında kısmi yüz tanıma sistemi geliştirmişlerdir (Jayant ve Borra, 2016).

Yüz tanıma, bilgisayar dünyasında özellikle son on yılda oldukça popüler bir araştırma alanı haline gelmiştir.. Görüntülerin analiz edilmesi ve anlamlandırılmasına yönelik olarak başarıyla kullanılmıştır. Doğası gereği bilgisayar mühendisleri, psikologlar ve nörologlar konuyla yakından ilgilenmişlerdir.

Yüz tanıma tekniğine yönelik olarak yapılan bir tarif, durumu şu şekilde özetlemektedir: Tanımlanmış bir sahneden video veya fotoğraf almak veya daha önce yüz veri tabanında kullanılan sahnedeki bir veya daha fazla kişiyi doğrulamaktır. Yüz tanıma teknolojisini, kişisel kimlik saptama, güvenlik sistemleri, görüntü film işleme, psikoloji, bilgisayar etkileşim, akıllı kart, hukuki yaptırımlar ve benzeri birçok alanda uygulamak mümkündür.

Bu çalışmada, yüz tanıma problemine yönelik olarak ORB (Rublee ve ark., 2011) (Oriented FAST and Rotated BRIEF - Yönelimli FAST ve Döndürülmüş BRIEF) tekniğini uygulayıp, bu tekniği LBP (Local Binary Patterns - Yerel İkili Örüntü) ve PCA (Principal Component Analysis - Temel Bileşen Analizi) gibi yaygın olarak kullanılan diğer tekniklerle mukayese ettikten sonra, ORB tekniğinin, diğer algoritmalara göre üstün yanı ortaya konulmaya çalışılmıştır.

Tez çalışması 5 bölümden oluşmaktadır. 1. Bölüm'de yüz tanıma üzerine yapılmış çalışmalardan bahsedilmektedir. 2. Bölüm'de konu ile ilgili kaynak araştırmaları sunulmaktadır. 3. Bölüm'de tez çalışmasında kullanılan materyaller ve metotlar açıklanmaktadır. 4 Bölüm'de tez çalışmasında yapılan testler ve sonuçları verilmektedir. 5. Bölüm'de ise sonuçlar ve öneriler sunulmaktadır.

2. KAYNAK ARAŞTIRMASI

Yüz tanıma, birçok alanı ve disiplini kapsayan önemli bir araştırma problemidir. Bu, yüz tanıma, banka kartı tanımlama, erişim kontrolü, güvenlik izleme ve gözetleme sistemi gibi çok sayıda pratik uygulamaya sahip olmanın yanı sıra, insanlar arasında etkili iletişim ve etkileşimler için gerekli olan temel bir insan davranışıdır. İlk olarak yüzleri sınıflandırmak için resmi bir yöntem önerilmiştir (Galton, 1878). Yazar, yüz profillerini eğri olarak toplama, normlarını bulma ve ardından diğer profilleri normdan sapmalarıyla sınıflandırmayı önermiştir. Bu sınıflandırma çok modludur, yani bir veri tabanındaki diğer vektörlerle karşılaştırılabilen bağımsız bir ölçüm vektörü ile sonuçlanır. Bu ilerlemeler, yüz tanıma sistemlerinin gerçek dünya ortamlarında gösterildiğine kadar uzanmıştır (Zhao ve Chellappa, 1999). Yüz tanımanın hızlı gelişimi, bazı faktörlerin birleşiminden kaynaklanmaktadır, bunlar: Algoritmaların aktif gelişimi, yüz görüntülerine ait büyük veritabanlarının kullanılabilirliği ve yüz tanıma algoritmalarının performansını değerlendiren bir yöntemdir. Literatürlerde, yüz tanıma problemi şu şekilde formüle edilebilir: Bir sahnenin statik (durağan) veya video görüntüleri verildiğinde, bir veritabanında saklanan yüzlerle karşılaştırarak sahnedeki bir veya daha fazla kişiyi tanımlama veya doğrulama şeklindedir. Kişinin doğrulanmasını yüz tanıma ile karşılaştırırken, farklı olan birçok yön vardır. Öncelikle, kişisel kimlik sisteminin bir müvekkili - yetkisiz kullanıcısı - kooperatif olması ve bir kimlik iddiası olması şarttır. Yüz tanıma için birçok yöntem önerilmiştir.

2.1. Özyüzler

Özyüz, tanıma ile yüzleşmek için en kapsamlı olarak araştırılan yaklaşımlardan biridir. Karhunen–Loève Dönüşümü (Karhunen–Loève Transformation-KLT), özresim, özvektör ve ana bileşen olarak da bilinir. (Sirovich ve Kirby, 1987; Kirby ve Sirovich, 1990), yüzlerin resimlerini verimli bir şekilde temsil etmek için temel bileşen analizi kullanılmıştır. Her bir yüz görüntüsünün, her yüz için küçük bir ağırlık topluluğu ve standart bir yüz resmi (Eigenface) ile yaklaşık olarak yeniden oluşturulabileceğini iddia etmişlerdir. Her bir yüzü tanımlayan etki, yüz görüntüsünü özresim üzerine yansıtmak suretiyle elde edilir. (Turk ve Pentland, 1991), yüz tanıma ve tanımlama için Kirby ve Sirovich tekniği ile desteklenen özyüzleri kullanmıştır.

Matematiksel olarak özyüzler, yüzlerin dağılımının temel bileşenleri veya yüz görüntü kümesinin kovaryans matrisinin özvektörleridir. Özvektörler, yüzler arasında sırasıyla farklı miktarlarda varyasyonları temsil edecek şekilde düzenlenir. Her bir yüz, özyüzlerin doğrusal bir kombinasyonu ile tam olarak gösterilebilir. Aynı zamanda sadece en büyük öz değerlere sahip en iyi özvektörler kullanılarak yaklaştırılabilir. En iyi M özdeğerleri M boyutlu bir alan, yani "yüz alanı" oluşturur. Yazarlar sırasıyla yüzde 96, yüzde 85 ve yüzde 64 doğru sınıflandırma, ortalama aydınlatma, yönelim ve büyüklük varyasyonlarını rapor etmiştir. Veritabanları, 16 kişiden 2.500 görüntü içermiştir.

Görüntüler büyük miktarda arka plan alanı içerdiğinden, yukarıdaki sonuçlar arka plandan etkilenmiştir. Yazarlar, aydınlatmadaki değişiklikler ve görüntüler arasında önemli bir korelasyon ile sistemin farklı aydınlatma koşullarında sağlamlık performansı açıklanmıştır. Bununla birlikte, (Grudin, 1997), tüm yüzlerin görüntüleri arasındaki korelasyonun, tatmin edici bir tanıma performansı için etkili olmadığı gösterilmiştir. Özyüz yaklaşımı için aydınlatma normalizasyonu (Kirby ve Sirovich, 1990) genellikle gereklidir.

(Zhao ve Yang, 1999), objektif Lambertian ise, rastgele aydınlatma etkilerini hesaba katarak, her biri farklı aydınlatma koşullarında çekilmiş üç görüntü kullanarak kovaryans matrisini hesaplamak için yeni bir yöntem önermiştir. (Pentland ve ark., 1994), gözler, burun ve ağız gibi yüz bileşenlerine karşılık gelen özdefektasyonlara, özyüzler üzerindeki erken çalışmalarını uzatmıştır. Yukarıdaki özdeğerlerden (yani özler, eigennose ve eigenmouth) oluşan bir modüler öz uzay kullanmışlardır. Bu yöntem, görünümdeki değişikliklere standart özyüz yönteminden daha az duyarlıdır. Sistem, yaklaşık 3000 kişiye ait 7.562 görüntü Yüz Tanıma Teknolojisi (Face Recognition Technology–FERET) veritabanında yüzde 95'lik bir tanıma oranı elde edilmiştir. Özet olarak, özyüz, hızlı, basit ve pratik bir yöntem olarak görünür. Bununla birlikte, genel olarak, ölçek ve aydınlatma koşullarında değişikliklere karşı değişmezlik sağlamaz.

Son zamanlarda, standart temel bileşen analizi yaklaşımı kullanılarak, kulak ve yüz tanıma ile yapılan 32 deneyde, tanıma performansının esas olarak, kulak görüntüleri veya yüz görüntüleri kullanılarak özdeş olduğu ve iki modlu tanıma sonuçları için istatistiksel olarak anlamlı bir performans geliştirilmesiyle birleştirdiği gösterilmiştir. Örneğin, 197-görüntülü eğitim setlerini kullanarak gün değişimi deneyinin tanıma

oranındaki fark, multimodal biyometrik için %71.9, kulak için %71.6 ve yüz için %70.5'dir. Multimodal biyometri ile ilgili önemli çalışmalar da vardır. Örneğin, (Hong ve Jain, 1998) multimodal biyometrik tanımlamada yüz ve parmak izi kullandı ve (Verlinde ve Mayoraz, 1998) yüz ve ses kullanılmıştır. Bununla birlikte, yüz ve kulağın kombinasyon halinde kullanımı, gözetim uygulamaları ile daha alakalı görünmektedir.

2.2. Sinir Ağları

Sinir ağları kullanmanın cazip olması, ağın doğrusal olmamasından kaynaklanabilir. Bu nedenle, özellik çıkarma adımı, doğrusal KLT yöntemlerinden daha verimli olacaktır. Yüz tanıma için kullanılan ilk yapay sinir ağları (Artificial Neural Networks-ANN) tekniklerinden biri, WISARD olarak adlandırılan ve her depolanmış birey için ayrı bir ağ içeren tek katmanlı uyarlamalı bir ağdır (Stonham, 1986). Başarılı bir tanıma için bir sinir ağı yapısının oluşturulmasının yolu çok önemlidir ve daha çok uygulamasına bağlıdır. Yüz tanısı için çok katmanlı perceptron (Sung ve Poggio, 1995) ve evrişimli sinir ağı (Lawrence ve ark., 1997) uygulanmıştır. Yüz doğrulama için, (Weng ve ark., 1993) çok çözünürlüklü bir piramit yapıdır. (Lawrence ve ark., 1997), yerel görüntü örneklemesini, Kendi Kendini Organize Eden Harita (Self-Organizing Map -SOM) sinir ağını ve evrişimli sinir ağı ile birleştiren hibrid bir sinir ağı önermiştir. SOM, görüntü örneklerinin, orijinal uzayda bulunan giriş ve çıkış alanlarında da yakın olduğu ve böylece görüntü örneğindeki değişikliklere boyut küçültme sağlayan bir topolojik uzay içine nicelleştirilmesini sağlar. Evrişim ağı, hiyerarşik bir dizi katmanında ardışık olarak daha büyük özellikleri çıkarır ve çeviriye, dönmeye, ölçeğe ve deformasyona kısmi değişmezlik sağlar. Yazarlar 40 bireyden 400 görüntü Olivetti Araştırma Laboratuvarı (Olivetti Research Laboratory-ORL) veritabanı üzerinde %96.2 'lik doğru tanımlamayı bildirmiştir. Sınıflandırma süresi 0.5 saniyeden daha az, ancak eğitim süresi 4 saat kadar uzundur. (Lin ve ark., 1997), Karar Tabanlı Sinir Ağları (Decision-Based Neural Networks-DBNN), Olasılıksal Karar Temelli Sinir Ağı (Probabilistic Decision-Based Neural Networks-PDBNN) kullanmıştır (Kung ve Taur, 1995). PDBNN, 1) Yüz detektörüne etkili bir şekilde uygulanabilir: Bir insan görüntüsünün, dağınık bir görüntüde yerini bulur, 2) Göz lokalizörü: Anlamlı özellik vektörleri oluşturmak için her iki gözün pozisyonlarını belirler ve 3) Yüz tanıyıcıdır. PDBNN, tamamen bağlı bir ağ topolojisine sahip değildir. Bunun yerine ağı, K alt ağlarına böler. Her alt küme, veritabanındaki bir kişiyi tanımlamaya adanmıştır. PDNN, nöronları için Guassian aktivasyon fonksiyonunu kullanır ve her "yüz alt ağının" çıkışı,

nöron çıkışlarının ağırlıklı toplanmasıdır. Başka bir deyişle, yüz alt ağı, popüler Guassian modelini kullanarak olasılık yoğunluğunu tahmin eder. Toplanır Beyaz Gausian Gürültüsü (Additive White Gaussian Noise-AWGN) şemasına kıyasla, Guassian'ın karışımı, yüz uzayındaki zaman olasılık yoğunluklarına yaklaşmak için çok daha esnek ve karmaşık bir model sağlar.

Birinci aşama; Her alt ağ kendi yüz görüntüleri ile eğitilir. Karar temelli öğrenme olarak adlandırılan ikinci aşamada, alt ağ parametreleri diğer yüz sınıflarından bazı özel örneklerle eğitilebilir. Karar temelli öğrenme şeması eğitim için tüm eğitim örneklerini kullanmaz. Sadece yanlış sınıflandırılmış desenlerde kullanılır. Numune, yanlış alt ağa sınıflandırılmışsa, alt ağ parametrelerini ayarlayacaktır. Böylece karar bölgesi yanlış sınıflandırılmış numuneye daha yakın olarak hareket ettirilebilir.

PDBNN tabanlı biyometrik tanımlama sistemi, hem sinir ağlarının hem de istatistiksel yaklaşımların yararına sahiptir ve dağınık hesaplama prensibinin paralel bilgisayarda uygulanması nispeten kolaydır. (Lin ve ark., 1997) PDBNN yüz tanıma sisteminin 200 kişiye kadar tanıma yeteneğine sahip olduğu ve yaklaşık 1 saniyede %96'ya kadar doğru tanıma oranına ulaşabileceği bildirilmiştir. Ancak, kişi sayısı arttığında, hesaplama gideri daha zorlu hale gelecektir. Genel olarak, sinir ağı yaklaşımları, sınıfların sayısı arttığında sorunlarla karşılaşır. Ayrıca, tek bir model görüntü tanıma testi için uygun değildir. Çünkü sistemlerin “optimal” parametre ayarına alıştırılması için kişi başına çoklu model görüntüleri gereklidir.

2.3. Grafik Eşleme

Grafik eşleştirme, yüz tanıma için başka bir yaklaşımdır. (Lades ve ark., 1993), en yakın depolanmış grafiği bulmak için Elastik Demet Grafik Eşleştirme (Elastic Bundle Chart Matching-EBCM) kullanılan değişmez nesne tanıma için dinamik bir bağlantı yapısı sunulmuştur. Dinamik bağlantı mimarisi, klasik yapay sinir ağlarının bir uzantısıdır. Ezberlenmiş nesneler, köşeleri, bir yerel güç spektrumu açısından birçok biçimli tanımlama ile etiketlenen ve kenarları geometrik uzaklık vektörleri ile etiketlenmiş olan seyrek grafiklerle temsil edilir. Nesne tanıma, eşleşen bir maliyet fonksiyonunun stokastik optimizasyonu ile gerçekleştirilen elastik grafik eşleştirme olarak formüle edilebilir. 87 kişilik bir veritabanında ve 15 derecelik bir rotasyonla farklı ifadeler içeren küçük bir dizi ofis ögeleri kümesinde iyi sonuçlar bildirilmiştir.

Eşleşen işlem, yaklaşık 30 saniye süren ve 23 aktarıcı ile paralel bir makinedeki 87 depolanmış objeyle karşılaştırmak için maliyetlidir. (Wiskott ve Von Der Malsburg, 1996) bu tekniği uzattı ve insan yüzlerini 112 nötr önden görünüm yüzüne sahip bir galeriye karşı eşleştirilmiştir. Derinlik rotasyonu ve değişen yüz ifadesi nedeniyle prob görüntüleri bozulmuştur. Büyük rotasyon açılarına sahip yüzler üzerinde teşvik edici sonuçlar elde edilmiştir. 15 derece rotasyonun 111 yüzü ve 30 derece rotasyonun 110 yüzünün eşleştirilmesi için 112 nötr frontal görüntüden oluşan bir galeriye %86.5 ve %66.4'lük tanıma oranları bildirilmiştir. Genel olarak dinamik bağlantı mimarisi, rotasyon değişmezliği açısından diğer yüz tanıma tekniklerinden üstündür; Bununla birlikte eşleştirme işlemi, hesaplama açısından maliyetlidir.

2.4. Gizli Markov Modelleri

Statik olmayan zaman serilerinde Gizli Markov Modelleri (Hidden Markov Model-HMM) stokastik modelleme, konuşma uygulamaları için çok başarılı olmuştur. (Samaria ve Fallside, 1993) bu yöntemi insan yüzü tanıma yöntemine uygulamıştır. Yüzler, gizli bir Markov modelinin durumları ile ilişkilendirilebilen gözler, burun, ağız vb. bölgelere sezgisel olarak bölünmüştür. Görüntüler iki boyutlu olduğundan ve HMM'ler tek boyutlu bir gözlem dizisi gerektirdiğinden görüntüler ya 1B geçici dizilere ya da 1B uzaysal dizilerine dönüştürülmelidir.

(Samaria ve Harter, 1994), bir bant örnekleme tekniği kullanılarak bir yüz görüntüsünden uzamsal bir gözlem dizisi çıkarmıştır. Her bir yüz görüntüsü bir 1B vektör serisi piksel gözlemiyle temsil edilmiştir. Her gözlem vektöründe bir L çizgisi bloğu ve birbirini izleyen gözlemler arasında bir M çizgisi üst üste gelir. Bilinmeyen bir test görüntüsü ilk önce bir gözlem dizisine örneklenir. Bu, model yüz veritabanındaki her HMM'e karşılık gelir her HMM farklı bir konuyu temsil eder. En iyi eşleşme olarak kabul edilir ve ilgili model, test yüzünün kimliğini ortaya çıkarır. HMM yaklaşımının tanıma oranı, 40 kişiden ve 400 görüntüden oluşan ORL veritabanı kullanılarak %87'dir. Bir sahte 2B HMM'nin (Samaria ve Harter, 1994), ön deneylerinde %95'lik bir tanıma oranı elde ettiği bildirilmiştir. Sınıflandırma süresi ve eğitim zamanı verilmemiştir (çok maliyetli olduğuna inanılmaktadır). Parametrelerin seçimi öznel sezgiye dayanmıştır.

2.5. Geometrik Özellik Eşleştirme

Geometrik özellik eşleme teknikleri, bir yüzün resminden bir dizi geometrik özelliklerin hesaplanmasına dayanır. Yüz tanıma, tek yüz özellikleri ayrıntılı olarak

açıklanmadığında 8x6 piksel (Tamura ve ark., 1996) gibi düşük çözünürlükte bile mümkün olsa da, yüz özelliklerinin genel geometrik konfigürasyonunun tanıma için yeterli olduğunu gösterir. Genel konfigürasyon, gözler ve kaşlar, burun, ağız ve yüz hatları şeklindeki ana yüz özelliklerinin konumunu ve boyutunu temsil eden bir vektör ile tarif edilebilir.

Geometrik özellikleri kullanarak otomatik yüz tanıma konusundaki öncü çalışmalardan biri, 1973'te (Kanade, 1974) tarafından yapılmaktadır. Sistem, kişi başına iki görüntü kullanan 20 kişilik bir veritabanında %75'lik bir tanıma oranı elde etti. (Goldstein ve ark., 1971; Kaya ve Kobayashi, 1972) el ile çıkarılan özniteliklerle sağlanan bir yüz tanıma programının tatmin edici sonuçlarla görünüşte farkedilebileceğini göstermiştir. Referans (Brunelli ve Poggio, 1993), burun genişliği ve uzunluğu, ağız konumu ve çene şekli gibi bir yüzün resminden bir dizi geometrik özelliği otomatik olarak çıkarmıştır. 35 boyutlu vektörden elde edilen 35 özellik mevcuttu. Tanıma daha sonra Bayes sınıflandırıcı ile gerçekleştirilmiştir. 47 kişilik bir veri tabanında %90'lık bir tanınma oranı bildirilmiştir.

(Cox ve ark., 1996), 685 bireyden oluşan bir sorgu veritabanında %95 tanıma oranı elde eden bir karışım-mesafe tekniğini ortaya koymuştur. Her yüz, elle çizilmiş 30 mesafeyle temsil edilmiştir. (Manjunath ve ark., 1992), her bir yüz görüntüsü için veri tabanındaki depolama gereksinimini büyük ölçüde azaltan özellik noktalarını tespit etmek için Gabor dalgacık ayrışmasını kullanmıştır. Tipik olarak, yüz başına 35-45 özellik noktası oluşturulmuştur. Eşleştirme süreci, özellik noktalarının topolojik grafik sunumundaki bilgilerini kullanmıştır. Farklı merkez konumlarını telafi ettikten sonra iki maliyet değeri, topolojik maliyet ve benzerlik maliyeti değerlendirilmiştir. Doğru kişiye en iyi eşleşme anlamında tanıma doğruluğu %86 ve doğru kişinin yüzlerinin %94'ü ilk üç aday içindedir.

Özet olarak, özellikler arasındaki kesin ölçülen mesafelere dayanan geometrik özellik eşleştirmesi, geniş bir veritabanında olası eşleşmeleri bulmak yararlı olabilir. Bununla birlikte, özellik konum algoritmalarının doğruluğuna bağlı olacaktır. Mevcut otomatik yüz özellik konum algoritmaları, yüksek bir doğruluk derecesi sağlamaz ve önemli bir hesaplama süresi gerektirir.

2.6. Şablon eşleme

Şablon eşleştirmesinin basit bir versiyonu, iki boyutlu bir yoğunluk değerleri dizisi olarak temsil edilen bir test görüntüsünün, tüm yüzü temsil eden tek bir şablon ile, Öklid mesafesi gibi uygun bir metrik kullanılarak karşılaştırılmasıdır. Yüz tanımada şablon eşleştirmenin daha çok karmaşık versiyonları vardır. Bir kişinin yüzünü temsil etmek için birden fazla yüz şablonu farklı bakış açılarından kullanılabilir.

Tek bir bakış açısındaki bir yüz, aynı zamanda, çok sayıda farklı küçük şablonlarla da temsil edilebilir (Baron, 1981; Brunelli ve Poggio, 1993). Gri seviyelerin yüz görüntüsü, eşleşmeden önce uygun şekilde işlenebilir (Bichsel, 1991). (Brunelli ve Poggio, 1993) 'da Bruneli ve Poggio, mevcut tüm yüzler için otomatik olarak dört özellik şablonu; Gözler, burun, ağız ve tüm yüzü seçti. Geometrik eşleştirme algoritmalarının ve şablon eşleştirme algoritmalarının performansını, 47 bireyden oluşan ve 188 görüntü içeren yüzleri, aynı veritabanında karşılaştırdılar. Şablon eşleşmesi, tanımada (yüzde 100 tanıma oranı) geometrik eşleştirmeden (yüzde 90 tanıma oranı) üstündür ve aynı zamanda daha basittir. Başlıca bileşenler (özyüzler veya özdeğerler olarak da bilinir), veri tabanındaki şablonların doğrusal kombinasyonları olduğu için, bu teknik korelasyondan daha iyi sonuçlar elde edemez (Brunelli ve Poggio, 1993), fakat daha az hesaplama gerektirebilir.

Şablon eşlemenin bir dezavantajı hesaplama karmaşıklığıdır. Başka bir sorun da bu şablonların tanımında yatmaktadır. Tanıma sistemi, şablon ve test görüntüsü arasındaki belirli farklılıklara karşı tolerans göstermesi gerektiğinden, bu tolerans bireysel yüzleri benzersiz kılan farklılıkları ortaya çıkarabilir.

Genel olarak, özellik eşleme ile karşılaştırıldığında şablon tabanlı yaklaşımlar daha mantıklı bir yaklaşımdır. Özetle, mevcut hiçbir teknik sınırsız değildir. Yüz tanıma tekniklerinin performanslarını, özellikle gerçek dünyada karşılaşılan çok çeşitli ortamlarda geliştirmek için daha fazla çaba gerekmektedir.

2.7. 3B morphable model

Morphable yüz modeli, bir dizi; örneğin şekil ve doku vektörlerinin herhangi bir dışbükey kombinasyonunun, gerçekçi bir insan yüzünü tarif edecek şekilde oluşturulmuş yüzlerin (Vetter ve Poggio, 1997) vektör uzay temsiline dayanmaktadır.

3B morphable modelinin resimlere uydurulması, farklı görüntüleme koşullarında tanıma için iki yolla kullanılabilir. Paradigma 1) Modeli taktıktan sonra, tanıma,

yüzlerin içsel şeklini ve dokusunu temsil eden model katsayılarına dayanabilir, Paradigma 2) Üç boyutlu yüz rekonstrüksiyonu, galeri prob görüntülerinden sentetik görünümler oluşturmak için de kullanılabilir (Beymer ve Poggio, 1995; Vetter ve Blanz, 1998; Zhao ve Chellappa, 2000; Georghiades ve ark., 2001). Sentetik görünümler, ikinci bakış açısı-bağımlı tanıma sistemine aktarılır.

Daha yakın zamanda, (Blanz ve Vetter, 2003) deforme olabilen 3B modellerini bilgisayar grafikleri ile projeksiyon ve aydınlatma simülasyonuyla birleştirmiştir. Buna gore bir kişinin tek bir görüntüsü verildiğinde, dokuyu ve tüm ilgili 3B sahne parametrelerini otomatik olarak tahmin eder. Bu çerçevede, derinlikteki dönüşler veya aydınlatmadaki değişiklikler çok basit işlemlerdir ve tüm pozlar ve aydınlatmalar tek bir model ile kaplıdır. Aydınlatma, Lambertian yansımasıyla sınırlı değildir, ancak insan derisinin görünümü üzerinde önemli etkiye sahip olan speküler yansımaları ve gölgeleri dikkate alır.

Bu yaklaşım, yüzlerin sınıfa özgü özelliklerini yakalayan mor yüzlü bir 3B yüz modeline dayanmaktadır. Bu özellikler otomatik olarak 3B taramalarından oluşan bir veri kümesinden öğrenilir. Morphable modeli, yüksek boyutlu bir yüz uzayında vektörler olarak yüzlerin şekillerini ve dokularını temsil eder. Yüz alanı içindeki doğal yüzlerin olasılık yoğunluk fonksiyonunu içerir. (Vetter ve Blanz, 1998) 'da sunulan algoritma, kafa konumu ve yönü, kameranın odak uzaklığı ve aydınlatma yönü dahil olmak üzere tüm 3B sahne parametrelerini otomatik olarak tahmin eder. Bu, sistemin dayanıklılığını ve güvenilirliğini önemli ölçüde artıran yeni bir başlatma prosedürüyle başarılır. Yeni başlatma, altı ve sekiz özellik noktası arasındaki görüntü koordinatlarını kullanır.

Yan görünüm galerisine dayalı CMU-PIE veritabanında doğru tanımlama yüzdesi %95 ve FERET setindeki karşılık gelen yüzde, önden görünüm galeri görüntülerine dayanarak, montajdan elde edilen tahmini kafa pozları ise %95.9 dur.

2.8. Hat Kenar Haritası

Kenar bilgisi, aydınlatma değişikliklerine belirli ölçüde duyarlı olmayan kullanışlı bir nesne gösterim özelliğidir. Kenar haritası çeşitli desen tanıma alanlarında yaygın olarak kullanılmasına rağmen, (Takacs, 1998) 'da bildirilen son çalışmalar dışında yüz tanımada ihmal edilmiştir.

Nesnelerin kenar görüntüleri, nesne tanıma ve gri düzeydeki resimlerle benzer doğruluk elde etmek için kullanılabilir. (Takacs, 1998), yüz görüntülerinin benzerliğini ölçmek için kenar haritalarından yararlanmış ve %92'lik bir doğruluk sağlanmıştır. Takács, yüz tanıma sürecinin daha erken bir aşamada başlayabileceğini ve üst düzey bilişsel işlevlerin katılımı olmadan yüzlerin tanınması için kenar görüntülerinin kullanılabileceğini savunmaktadır.

(Gao ve Leung, 2002) tarafından önerilen bir Çizgi Kenar Haritası yaklaşımı, bir yüz kenarı haritasından çizgiler olarak özellikler çıkarılmaktadır. Bu yaklaşım, şablon eşleştirme ve geometrik özellik eşleştirmesinin bir kombinasyonu olarak düşünülebilir. Hat Kenar Haritası (Line Edge Map-LEM) yaklaşımı, yalnızca aydınlatmaya ve düşük bellek gereksinimine karşı değişme gibi özellik tabanlı yaklaşımların avantajlarına sahip olmakla kalmaz, aynı zamanda şablon eşleştirmesinin yüksek tanıma performansına da sahiptir.

Çizgi Kenar Haritası, yüz yüze ait pikselleri çizgi parçalarına gruplayarak yapısal bilgileri bir yüz görüntüsünün uzamsal bilgisiyle birleştirir. Kenar haritasının inceltilmesinden sonra, bir yüzün LEM'ini oluşturmak için çokgen bir çizgi yerleştirme işlemi (Leung ve Yang, 1990) uygulanır. Bir insan ön yüzü LEM'nin bir örneği Şekil 2.l'de gösterilmektedir. LEM temsili, sadece eğriler üzerindeki çizgi parçalarının uç noktalarını kaydettiği için depolama gereksinimini azaltır. Ayrıca, LEM'in, düşük seviyeli kenar haritası sunumundan türetilen orta düzeyde bir görüntü temsili olması nedeniyle, aydınlatma değişikliklerine karşı daha az duyarlı olması beklenmektedir. LEM'in temel birimi, kenar haritasının piksellerinden gruplanan çizgi segmentidir.



Şekil 2.1. Bir yüz lem'inin çizimi

Yüz tanımlama uygulamasında LEM eşleştirmesinin ön-işlemi olarak kullanılabilen bir yüz önleme algoritması önerilmiştir. Ön ısıtma işlemi, aday sayısını azaltarak aramayı hızlandırabilir ve gerçek yüz LEM eşleşmesi sadece kalan modellerin bir alt kümesi üzerinde gerçekleştirilebilir.

Kontrollü ideal koşullar altında ön yüzler üzerinde yapılan deneyler, önerilen LEM'in tutarlı bir şekilde kenar haritasına üstün olduğunu göstermektedir. LEM sırasıyla yüz veri tabanlarındaki giriş ön yüzlerinin %100 ve %96.43'ünü doğru bir şekilde tanımlamaktadır. Özyüz yöntemi ile karşılaştırıldığında, LEM ideal koşullar altında yüzler için özyüz yöntemi olarak ve hafif görünüm varyasyonları olan yüzler için özyüz yönteminden önemli ölçüde daha üstün performans göstermiştir (Çizelge 2.1). Ayrıca, LEM yaklaşımı, özyüz yöntemine ve kenar haritası yaklaşımına göre boyut değişimlerinde çok daha sağlamdır (Çizelge 2.2).

Çizelge 2.1. Kenar haritası yüz tanıma sonuçları. eigenface (20-eıgenvectors) ve lem (Gao ve Leung, 2002)

	Bern veritabanı			AR veritabanı		
Yöntem	EM	özyüz	LEM	EM	özyüz	LEM
Tanıma oranı	97.7%	100%	100%	88.4%	55.4%	96.4%

Çizelge 2.2. Boyut değişikliği ile rekor sonuçları (Gao ve Leung, 2002)

	Top 1	Top 2	Top 3
Kenar haritası	43.3%	56.0%	64.7%
Özyüz (112-özvektörler))	44.9%	68.8%	75.9%
LEM (pLHD)	53.8%	67.6%	71.9%
LEM (LHD)	66.5%	75.9%	79.7%

(Gao ve Leung, 2002) 'de, LEM yaklaşımının, değişen aydınlatma koşullarında yüzleri tanımlamak için özyüz yaklaşımından önemli ölçüde üstün olduğu gösterilmiştir. LEM yaklaşımı aynı zamanda, özyüz yönteminden farklı pozlamalara karşı daha az duyarlıdır, ancak büyük yüz ifadesi değişikliklerine daha duyarlıdır.

2.9. Destek Vektör Makinesi (SVM)

Destek Vektör Makinesi (Support Vector Machine-SVM) yüksek genelleştirme performansından ötürü, başka bilgi eklemeye gerek kalmadan, genel amaçlı örüntü tanıma için etkili bir yöntem olarak kabul edilen bir öğrenme tekniğidir (Vapnik, 2013). Sezgisel olarak, iki sınıfa ait bir dizi nokta verildiğinde, bir SVM, aynı sınıftaki noktaların mümkün olan en büyük kısmını aynı tarafta ayıran hiper düzlemi bulur ve her iki sınıftan da hiper düzlemden uzaklığı en üst düzeye çıkarır. (Vapnik, 2013) 'e göre, bu hiperdüze, Optimal Ayırma Hiperdüzeni (Optimal Separating Hyperplane-OSH) olarak adlandırılır ve bu da sadece eğitim setindeki örneklerin yanlış tanımlanması riskini en aza indirir, aynı zamanda test kümesinin görülmemiş bir örneğini de oluşturur.

SVM ayrıca polinomiyal sinir ağlarını veya Radyal Temel işlevi sınıflandırıcılarını eğitmenin bir yolu olarak da görülebilir. Burada kullanılan eğitim teknikleri, genelleştirme hatasının sınırlarını en aza indirerek daha iyi genelleme kabiliyetlerinin elde edildiğini ifade eden Yapısal Risk Azaltma (Structural Risk Minimization-SRM) prensibine dayanmaktadır. Gerçekten de, bu öğrenme tekniği, doğrusal olarak kısıtlı bir Kuadratik Programlama (Quadratic Programming-QP) problemini çözmekle eşdeğerdir. SVM, ortalama boyuttaki yüz tanıma sistemleri için uygundur. Çünkü normal olarak bu sistemlerin sadece az sayıda eğitim örneği vardır.

Fakat çok sayıda QP probleminde, (Lin, 2001) global optimizasyonu garanti eden bir ayrıştırma algoritması sunmuş ve SVM'leri çok büyük veri kümeleri üzerinde eğitmek için kullanmıştır.

Özet olarak SVM'lerin ana özellikleri şunlardır: 1) Genelleme hatası üzerinde resmi olarak kanıtlanmış bir üst sınırı en aza indirgemesi; 2) Çekirdekler açısından ikili bir formülasyon vasıtasıyla yüksek boyutlu özellik uzaylarında çalışması; 3) Bu özellik alanlarındaki tahminin, girdi verileri üzerindeki oldukça ilgili sınıflandırma kriterlerine tekabül eden hiperplanlara dayanması; ve 4) Eğitim veri kümesindeki aykırılıkların yumuşak kenar boşlukları ile ele alınabilmesidir.

Yakın zamanda bilgisayarlı görme problemine SVM'lerin uygulanması önerilmiştir. (Guo ve ark., 2000), yüz tanıma problemini çözmek için SVM'leri ikili ağaç tanıma stratejisiyle kullanmıştır. Özellikler çıkarıldıktan sonra, her bir çift arasındaki ayrımcılık işlevleri SVM'ler tarafından öğrenilmiştir. Daha sonra, ayrık test seti tanıma için sisteme girmiştir. Test örneklerini tanımak için ikili ağaç yapısı oluşturmayı önerdiler. İki deney seti sunulmuştur. İlk deney, 40 kişiden ve 400 görüntüden oluşan Cambridge Olivetti Araştırma Laboratuvarı ORL yüz veritabanı üzerindedir. İkincisi 137 bireyden ve 1079 görüntüden oluşan daha büyük bir veri setindedir. SVM tabanlı tanıma, En Yakın Merkez Sınıflandırması (Nearest Center Classification-NCC) kriteri kullanılarak standart özyüzler yaklaşımı ile karşılaştırılmıştır. Her iki yaklaşım da özyüz özelliği ile başlar, ancak sınıflandırma algoritmasında farklıdır. Hata oranları, özyüz sayısının, yani özellik boyutunun işlevi olarak hesaplanır. SVM'nin minimum hatası %8.79, bu da NCC'nin %15.14'ünden daha iyidir.

(Phillips, 1999) 'de, yüz tanıma problemi, iki yüz görüntüsü arasındaki farklılıkları modelleyen, farklılık uzamında bir problem olarak formüle edilir. Farklı alanlarda, yüz tanıma iki sınıf problem olarak formüle edilir. Vakalar 1) Aynı kişinin yüzleri arasındaki benzerlikler ve 2) Farklı insanların yüzleri arasındaki benzerliklerdir. Karar yüzeyinin yorumlanmasını değiştirerek, yüzler arasındaki farkların örneklerinden öğrenilen, yüzler arasında bir benzerlik ölçüsü oluşturulmuştur. SVM tabanlı algoritma, FERET veri tabanından gelen zor bir görüntü dizisinde Temel Bileşenler Analizi (Principal Component Analysis-PCA) tabanlı algoritma ile karşılaştırılmıştır. Hem doğrulama hem de tanımlama senaryoları için performansı ölçülmüştür. SVM için

kimlik performansı %77-78, PCA için %54'tür. Doğrulama için, eşit hata oranı SVM için %7 ve PCA için %13'tür.

(Heisele ve ark., 2001), bileşen tabanlı bir teknik ve yüz tanıma için iki küresel teknik sunmuş ve poz değişikliklerine karşı sağlamlık açısından performanslarını değerlendirmiştir. Bileşen tabanlı sistem, bir dizi 10 yüz bileşenini tespit edip çıkarır ve bunları, doğrusal SVM'ler tarafından sınıflandırılan tek bir özellik vektörüne yerleştirir. Her iki küresel sistemde de tüm yüz algılanır, görüntüden çıkarılır ve sınıflandırıcılara girdi olarak kullanılır. İlk küresel sistem, veritabanındaki her kişi için tek bir SVM'den oluşmuştur. İkinci sistemde, her bir kişinin veritabanı kümelenmiş ve bir dizi görüntüye özgü SVM sınıflandırıcısı üzerinde eğitilmiştir. Sistemler, yaklaşık 400'e kadar derinlikte döndürülmüş yüzleri içeren 8.593 gri yüzlü görüntüden oluşan bir veri tabanı üzerinde test edilmiştir. Tüm deneylerde, bileşen tabanlı sistem, küresel sistem için daha güçlü bir sınıflandırıcı kullanılmasına rağmen yani doğrusal SVM'ler yerine doğrusal olmayan küresel sistemleri geride bırakmıştır. Bu, tüm yüz deseni yerine yüz bileşenlerinin giriş özellikleri olarak kullanılmasının yüz tanıma testini önemli ölçüde basitleştirdiğini göstermektedir.

(Huang ve ark., 2002), eğitim sürecine bir 3B morphable modelin dahil edilmesiyle bileşen tabanlı yüz tanımada yeni bir gelişme sunmuştur. Bir kişinin iki yüz görüntüsüne ve 3B morfasyon modeline dayanarak, veritabanındaki her bir kişinin 3B yüz modelini hesaplamıştır. 3B morphable değişken pozlar ve aydınlatma koşulları altında sunarak, bileşen bazlı tanıma sistemini eğitmek için çok sayıda sentetik yüz görüntüsünü kullanmıştır. 360 derinliğe kadar döndürülen yüzler için %98 civarında bileşen bazlı tanıma oranları elde edilmiştir. Sistemin önemli bir dezavantajı, bakış açılarından ve farklı aydınlatma koşulları altında alınan çok sayıda eğitim görüntüsüne ihtiyaç duymasıdır.

(Jonsson ve ark., 2000) 'de, destek vektörlerinin öğrenilmesini gerektiren alıcıya özel bir çözüm benimsenmiştir. Bu temsil (Phillips, 1999) 'de verilenlerden farklıdır. Daha önce bahsedildiği gibi, (Phillips, 1999) 'de, SVM, alıcı-içi ve alıcı-arası farklılık görüntüleri arasındaki popülasyonları ayırt etmek için eğitilmiştir. Dahası, temsil ve ön işlemden bağımsız olarak eğitim verilerinden ilgili ayrımcı bilgileri çıkarmak için SVM'nin içsel potansiyelini araştırırlar. Bu amaca ulaşmak için, yüzlerin hem Temel Bileşen hem de Doğrusal Ayırma (Linear Discriminant-LD) alt uzayında temsil edildiği deneyler tasarlamışlardır. Sonuncusu (Fisherfaces), ayrımcı özellik çıkarımı üzerine

odaklanarak bir yüz temsilinin örneği olarak kullanılırken, birincisi basitçe veri sıkıştırmasına ulaşır. Ayrıca, görüntü fotometrik normalizasyonun SVM yönteminin performansı üzerindeki etkisini inceler ve deney sonuçları kıyaslama yöntemleri ile karşılaştırıldığında üstün performans gösterir. Bununla birlikte, temsil alanı zaten ayrımcı bilgiyi yakalayıp vurguladığında, SVM'ler üstünlüklerini kaybeder. Sonuçlar ayrıca SVM'lerin, eğitim verilerinde yeterince temsil edildiği sürece aydınlatmadaki değişikliklere karşı güçlü olduğunu göstermektedir. Önerilen sistem, son derece rekabetçi sonuçlar elde eden 295 kişilik geniş bir veri tabanında değerlendirilmektedir. Doğrulama için %1 ve tanıma için %2'lik bir sıralı hata oranı görülmektedir.

(Guo ve ark., 2001) 'de, K sınıfı bir sınıflandırma testi için çok vasıflı sınıflandırma problemini ele almak amacıyla yeni bir yapı önermiştir. Her biri en güvenilir ve en uygun olan Optimal İkili Bağlanma Sınıflandırıcısı (Optimal Pairwise Coupling Classifier-O-PWC) dizisi oluşturulmuştur. Son karar, bu O-PWC'nin sonuçları birleştirilerek elde edilecektir. Bu algoritma, ifade, poz ve yüz detaylarında oldukça yüksek değişkenlik derecesine sahip 40 kişiden ve 400 görüntüden oluşan ORL yüz veri tabanı üzerinde uygulanmaktadır. Eğitim seti 200 örnek (her birey için 5) içermektedir. Kalan 200 örnek test seti olarak kullanılır. Sonuçlar, hesaplama maliyetinin çok fazla artmayacağını, doğruluk oranının arttığını göstermektedir. Çizelge 2.3'te ORL veri tabanı üzerinde farklı tanıma yöntemlerinin karşılaştırmasını göstermektedir.

Çizelge 2.3. Doğruluk oranı karşılaştırma

Yöntem	Maksimum Oylama	PWC	O-PWC (çapraz entropi)	O-PWC (Kare hatası)
Oran	94%	95.13%	96.79%	98.11%

(Déniz ve ark., 2003) yüz tanıma problemi için SVM ve Bağımsız Bileşen Analizi (Independent Component Analysis-ICA) tekniklerini birleştirmektedir. ICA, temel Bileşen Analizi'nin genelleştirilmesi olarak düşünülebilir. (Şekil 2.2) PCA ve ICA baz görüntüleri arasındaki farkı göstermektedir.



Şekil 2.2. Yale yüz veritabanı için bazı orijinal (solda), pca (orta) ve ıca (sağda) temel görüntüleri

İki farklı yüz veritabanında (Yale ve AR veri tabanları) deneyler yapılmıştır. Elde edilen sonuçlar Çizelge 2.4'te görülmektedir. SVM sadece polinom (derece 3'e kadar) ve Guassian çekirdekleri (çekirdek parametresi σ'nu değiştirirken) ile kullanılmıştır.

Çizelge 2.4. En Yakın ve En İyi Sınıflandırıcı (NMC) Ve SVM İle YALE Ve AR Görüntüleri İçin Alan Tanıma Oranları. SVM İçin, Yanlış Sınıflandırma Ağırlığı Olarak 1000'in Değeri Kullanıldı. Son Kolon Temsilcileri

Veri kümesi	NMC Öklid mesafesini kullanarak		SVM			Gaussian
			P=1	P=2	P=3	
Yale	PCA	92.73%	98.79%	98.79%	98.79%	99.39%
	ICA	95.76%	99.39%	99.39%	99.39%	99.39%
AR	PCA	48.33%	92%	91.67%	91%	92.67%
	ICA	70.33%	93.33%	93.33%	92.67%	94%

SVM tabanlı çoklu görüşlü yüz algılama ve tanıma çerçevesi (Li ve ark., 2000) 'te açıklanmıştır. Yüz algılama, her biri belirli bir görüşten sorumlu olan birkaç dedektör inşa edilerek gerçekleştirilir. Yüz görüntülerin simetrik özelliği, modellemenin karmaşıklığını basitleştirmek için kullanılır. Destek Vektör Regresyon tekniğini kullanarak elde edilen kafa pozunun tahmini, uygun yüz dedektörünü seçmek için çok önemli bilgiler sağlar. Bu, diğer yöntemlerle karşılaştırıldığında, çoklu görüntü yüz algılamada doğruluğu geliştirmeye ve hesaplamaların azaltılmasına yardımcı olur.

Video dizileri için pürüzsüzleştirme stratejisi değiştirerek daha fazla hesaplama azaltımı sağlanabilir. Yüz dedektörleri önden görünümde bir yüz bulduğunda, yüz tanıma için bir SVM tabanlı çok sınıflı sınıflandırıcı etkinleştirilir. Yukarıdaki tüm sorunlar bir SVM çerçevesinde entegre edilmiştir. Bu yaklaşımın önemli bir özelliği, özellikle düşük çözünürlükte, büyük ölçekli değişikliklerde ve derinlikteki rotasyonda, kısıtlı bir ortamda güçlü bir performans elde edebilmesidir. Dört video dizisindeki test sonuçları ile ilgili algılama oranı %95 ve tanıma doğruluğu %90'ın üzerindedir. Tam algılama ve tanıma hızı, Pentium II300 PC'de 4 saniyedir. Tam algılama ve tanıma hızı PentiumII300 PC'de 4 kare / saniye kadardır.

(Kim ve ark., 2002) 'te, birkaç SVM sınıflandırıcısı ve bir NN hakemini birleştiren yeni bir yüz tanıma yöntemi sunulmuştur. Önerilen yöntem, herhangi bir açık özellik çıkarma şeması kullanmamıştır. Bunun yerine SVM'ler, giriş deseni olarak ham piksellerin gri seviye değerlerini almıştır. Bu yapılandırmanın mantığı, bir SVM'nin gri boyutlu yüz görüntü alanı gibi yüksek boyutlu alanda öğrenme yeteneğine sahip olmasıdır. Ayrıca, bir yerel korelasyon çekirdeği (polinom çekirdeği yönteminin değiştirilmiş formu) ile SVM'lerin kullanımı, özellik çıkarma ve sınıflandırma için etkin bir kombinasyonu sağlar, böylece dikkatle tasarlanmış bir özellik çıkarıcı ihtiyacını ortadan kaldırır.

Birden fazla SVM tahkim edildiğinde ortaya çıkan ölçekleme problemi, bir NN'yi eğitilebilir bir ölçekleyici olarak kabul edilmesiyle çözülür. Deneysel sonuçlardan ORLveri tabanını kullanarak (Şekil 2.3), önerilen yöntem 40 sınıflı bir yüz deseni için ortalama işlem süresi 0.22 saniyede %97.9'luk bir tanıma oranı ile sonuçlanmıştır. (çizelge 2.5), ORL veri tabanını kullanan sonuçların elde edilebileceği çeşitli sistemlerin performansının bir özetini göstermektedir. Önerilen yöntem, en iyi performans ve ikinci en iyi performans gösteren sistemden - konvolüsyon NN'den (% 44.7) önemli oranda azalma göstermiştir.



Şekil 2.3. Orl veri tabanından alınan örnek görüntüler

Çizelge 2.5. Çeşitli sistemlerin hata oranları

Yöntem	Hata oranı (%)
Özyüzler	10.0
Psudo-2DHMM	5.0
Konvolüsyon NN	3.8
Yerel korelasyon çekirdeği olan SVM'ler	2.1

Öte yandan, (Jonsson ve ark., 2002) yüz kimlik doğrulama (doğrulama) bağlamında SVM'leri incelemiştir. Çalışmaları, SVM yaklaşımının, eğitim verilerinden ilgili ayrımcı bilgileri çıkarabildiği hipotezini desteklemektedir ve kıyaslama yöntemleri üzerindeki üstün performansının ana sebebidir. Temsili alan, su yüzüne çıkmış bilgi içeriğini zaten yakalar ve vurgularsa, SVM'ler üstünlüklerini kaybeder. SVM'ler ayrıca, eğitim verilerinde yeterince temsil edildiği sürece, aydınlatma değişiklikleri ile başa çıkabilirler. Bununla birlikte, özellik çıkarma (Fisherfaces) veya normalizasyon ile arındırılan veriler üzerinde, SVM'ler aşırı eğitimli olabilir ve bu da genelleme yeteneğinin kaybolmasına neden olabilir.

Bunlardan aşağıdaki sonuçlar çıkarılabilir:

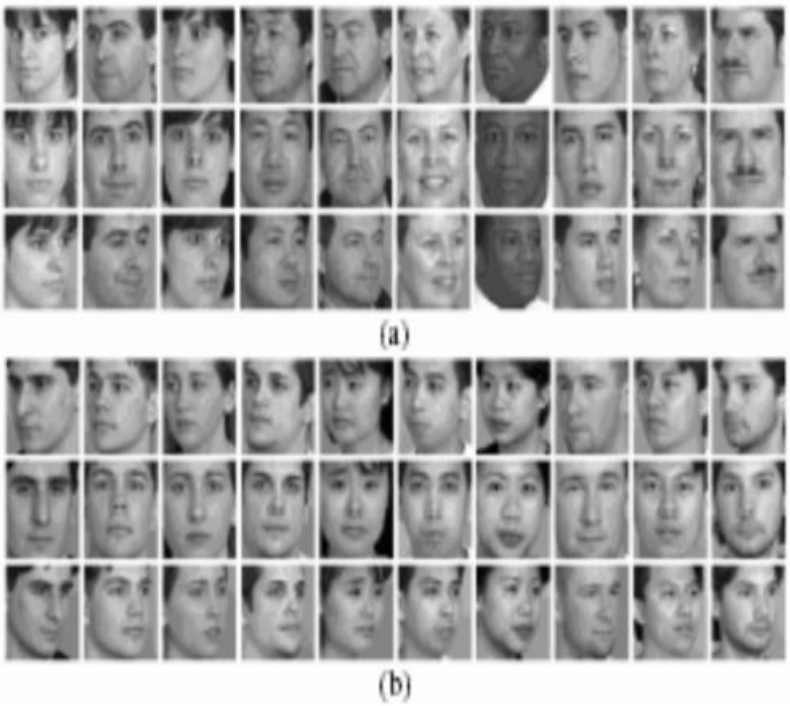
- SVM yaklaşımı, ilgili ayrımcı bilgileri tam otomatik olarak verilerden çıkarabilmektedir. Ayrıca aydınlatma değişiklikleri ile başa çıkabilir. Bu özellikteki en önemli rol, SVM'lerin doğrusal olmayan karar sınırlarını öğrenebilme yeteneği tarafından oynanır.

- Öznitelik çıkarma (Fisherfaces) veya normalizasyon ile arındırılan verilerde SVM'ler aşırı eğitimli olabilir ve bu da genelleme yeteneğinin kaybolmasına neden olabilir.

- SVM'ler birçok parametreyi içerir ve farklı çekirdekler kullanabilir. Bu, en iyi çözümü bulmak için tamamen araştırıldığını garanti etmeden optimizasyon alanını oldukça kapsamlı hale getirir.

- Bir SVM alıcı başına eğitim almak için yaklaşık 5 saniye sürer (Sun Ultra Enterprise 450 cihazında). Bu, oklit ve korelasyon katsayısı sınıflandırıcıları için alıcıya özgü eşiklerin belirlenmesinden daha büyük bir büyüklük sırasına dayanmaktadır. Ancak pratik açıdan bakıldığında, fark önemsizdir.

(Huang ve ark., 1998), SVM kullanarak yüz poz ayrımcılığı problemi için bir yaklaşımı tarif etmektedir. Yüze dayalı ayrımcılık, yüz görüntüsünü bilinen birkaç pozdan biri olarak etiketleyebilir. Yüz görüntüleri standart FERET veri tabanından çizilir, bkz. (Şekil 2.4).



Şekil 2.4. (a) Eğitim ve (b) test görüntüleri örnekleri

Eğitim seti ön ve eşit olarak dağıtılmış 150 görüntüden, sırasıyla yaklaşık 33.750 sol ve sağ pozlardan içeren testlerden oluşmuştur ve test seti yine üç farklı poz tipi arasında eşit olarak dağıtılan 450 görüntüden oluşmuştur. SVM mükemmel doğruluk elde etmiştir %100 - görünmeyen test verilerindeki olası üç yüz pozu arasında, derece 3'ün polinomları ya da çekirdek yaklaşma işlevleri olarak Radyal Tabanlı Fonksiyon Radial Basis Functions-RBF) kullanılarak ayırt edilmiştir. Polinom çekirdeği ve RBF çekirdekleri kullanılarak deney sonuçları sırasıyla Çizelge 2.6'da verilmiştir.

Çizelge 2.6. Polynomıal kerneller kullanarak deneysel sonuçla

Sınıflandırıcıların tipi	Destek vektörlerinin sayısı	150 örnekte eğitim doğruluğu	450 örneğe test doğruluğu	Maks. üç sınıflandırıcıdan çıkan test doğruluğu
Ön ve diğerleri sol	33	100%	99.33%	
33.750 ve diğerleri sağ	25	100%	99.56%	
33.750 ve diğerleri	37	100%	99.78%	100%

(Pang ve ark., 2003) bireyleri ifşa etmeden ve grup büyüklüğünü veya grubun üyelerini kısıtlamadan bireyin dinamik gruba üyeliğini doğrulamak için bir yöntem sunar. Küçük boyutlu bir kümeyi (üyelik) evrensel kümedeki tamamlayıcı kümesinden (üyeliksiz) ayırmak için üyelik kimlik doğrulamasını iki sınıf bir yüz sınıflandırma sorunu olarak ele alırlar. Doğrulamada, yanlış pozitif hata en kritik olanıdır. Neyse ki hata, SVM topluluğu kullanılarak,her SVM'nin bağımsız bir üyelik ve üyelik dışı sınıflandırıcı olarak davrandığı ve birkaç SVM'nin, SVM'lerin yarısından fazlası

tarafından yapılan sınıflandırmayı seçen çok sayıda oylama şemasında birleştirildiği durumlarda geçerli olarak kaldırılabilir. Çizelge 2.7'de Rbf kernelleri kullanılarak deneysel sonuçlar verilmiştir.

Çizelge 2.7. Rbf kernellerini kullanarak deneysel sonuçlar

Sınıflandırıcılar tipi	Destek Vektör Sayısı	Eğitim Hassasiyeti 150 örnek üzerinde	Test Hassasiyeti 450 örnek üzerinde	Maks. Kullanarak Test Doğruluk üç sınıflandırıcıdan çıktı
Ön ve diğerleri sol	47	100%	100%	
33.750 ve diğerleri sağ	38	100%	100%	
33.750 ve diğerleri	43	100%	100%	100%

Yüz görüntülerinin iyi bir şekilde kodlanması için, Gabor filtreleme, temel bileşen analizi ve doğrusal ayrımcılık analizi, etkin yüz ifadesi elde etmek, veri boyutunun etkin bir şekilde düşürülmesi ve farklı yüzlerin ayrılması için sırasıyla giriş yüz görüntüsüne uygulanmıştır. Daha sonra, SVM topluluğu, üyelik grubuna dâhil edilmiş olsun ya da olmasın, bir giriş yüzü görüntüsünün doğrulanması için uygulanmıştır. Deney sonuçları, SVM topluluğunun, farklı grup büyüklükleri veya farklı grup üyelerinin varyasyonlarıyla başa çıkabilmek için üyelik dışı ve istikrarlı bir sağlamlığı tanıma yeteneğine sahip olduğunu göstermiştir. Aynı grup büyüklüğündeki gruptaki üyelerin varyasyonuna bakılmaksızın doğru kimlik doğrulama oranı neredeyse %97 ile %98,5 arasında değişmektedir.

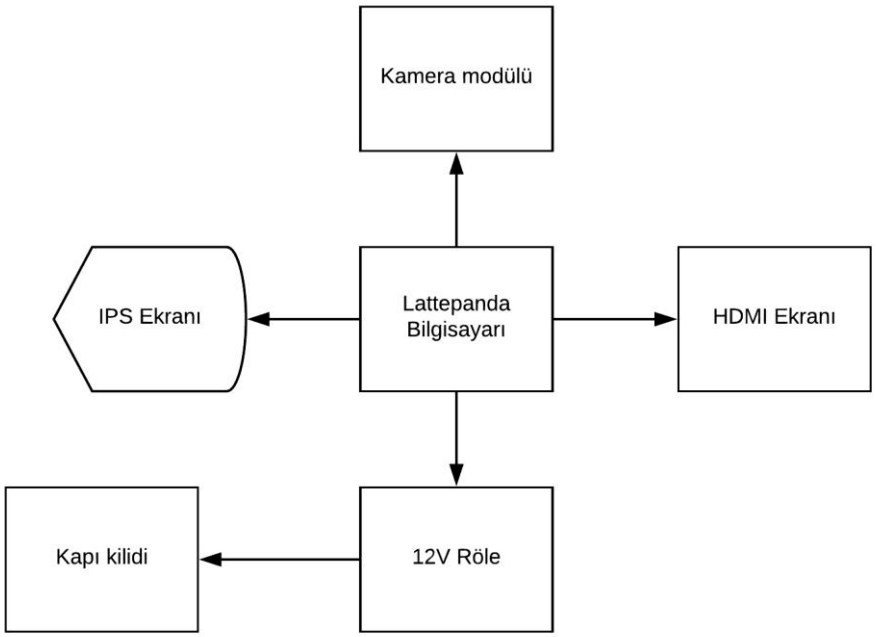
Bununla birlikte, önerilen kimlik doğrulama yöntemiyle ilgili bir problem, sınırlı eğitim verisi kümesine bağlı olarak, üyelerin boyutu küçük olduğunda (<20), üyelik için doğru sınıflandırma oranının oldukça düşmesidir. Yine de, simülasyon sonuçları, önerilen yöntemin kimlik doğrulama performansının, üye grubu için 50 kişiden az bir büyüklükte durabileceğini göstermektedir.

3. MATERYAL VE YÖNTEM

3.1. Malzemeler

Bu bölümde tez çalışmasında kullanılan materyaller ve yöntemlerden bahsedilmiştir. Yöntem olarak (ORB , LBP ve PCA) algoritması kullanılmıştır.

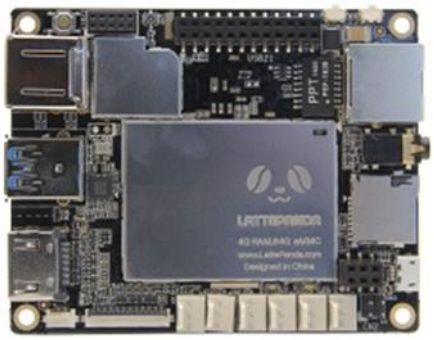
Tez çalışmasında kamera modülü, IPS ekranı, HDMI ekranı, Lattepanda bilgisayarı, 12V Röle ve kapı kilidi kullanılmıştır. Şekil 3.1'de sistem blok diyagramı gösterilmiştir.



Şekil 3.1. Sistem blok diyagramı

3.1.1. Lattepanda Bilgisayar Modülü

Lattepanda, windows işletim sistemini çalıştıran bir mini bilgisayar ve geliştirme kartıdır. Dört çekirdekli bir Intel Atom işlemciden oluşan Lattepanda kartı Şekil 3.2'de görülmektedir. Çizelge 3.1'de ise Lattepanda kartının özellikleri verilmiştir.



Şekil 3.2. Lattepanda bilgisayar modülü
Çizelge 3.1. Lattepanda bilgisayar modülü özellikleri

Lattepanda Özellikleri	
İşlemci	Intel® Cherry Trail Z8350 Quad Core
İşletim sistemi	Windows 10
RAM	4GB
Depolama	64GB eMMC
GPU	Intel HD Graphics, 12 EUs @200-500Mhz, single-channel memory
Portlar	USB3.0 port and two USB 2.0 ports
Bağlantı	Wi-Fi 802.11n 2.4G, Bluetooth 4.0
Diğer Özellikler	Built-in Arduino Leonardo

3.1.2. Kamera

Tez çalışmasında yüzler için net görüntüler yakalamak amacıyla Logitech 720p Webcam kullanılmıştır. Şekil 3.3'te görülen kameranın özellikleri Çizelge 3.2'de verilmektedir.



Şekil 3.3. Kamera modülü
Çizelge 3.2. Logitech HD 720p kamera özellikleri

Kamera Özellikleri	
Bağlantı Tipi:	USB kablosu
USB Tipi:	USB 2.0
Mikrofon:	Built-in, Gürültü bastırma
Lens ve Sensör Tipi:	Plastik
Odak Tipi:	Sabit
Görüş Alanı (FOV):	60°
Odak Uzunluğu:	4.0 mm
Optik Çözünürlük:	1280 x 960 1.2MP
Görüntü Yakalama (4:3 SD):	320x240, 640x480 1.2 MP, 3.0 MP
Görüntü Yakalama (16:9 W):	360p, 480p, 720p
Video Yakalama (4:3 SD):	320x240, 640x480, 800x600
Video Yakalama (16:9 W):	360p, 480p, 720p,
Frame Oranı (max):	30fps @ 640x480
Cable Length	5 Feet or 1.5 Meters

3.1.3. Ekran

Tez çalışmasında kullanılan 7 inç HD Düzlem İçi Anahtarlama (In Plane Switching- IPS) ekran Şekil 3.4'te görülmekte ve özellikleri Çizelge 3.3'te verilmektedir.



Şekil 3.4. Lattepanda 7 inç ıps ekran modülü
Çizelge 3.3. Lattepanda ıps ekran özellikleri

7 inç IPS Ekran Özellikler	
Boyut	7 inch IPS Görüntüle
çözüm	1024*600 yüksek çözünürlük
boyutlar	164.0mm x 96.0mm x 2.6mm
Bağlayıcı	10cm FPC

3.1.4. 3B baskılı Muhafaza

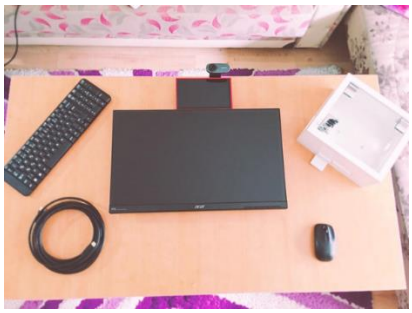
Solidworks ile tasarlanan 3 boyutlu yazıcı makinede basılan plastik muhafaza. Şekil 3.5'te gösterilmiştir.



Şekil 3.5. 3B baskılı muhafaza

3.1.5. Sistemin Diğer Donanımları

Tez çalışmasında kullanılan diğer donanımlar olan ekran, klavye, fare, koruyucu kılıf Şekil 3.6'da gösterilmiştir.

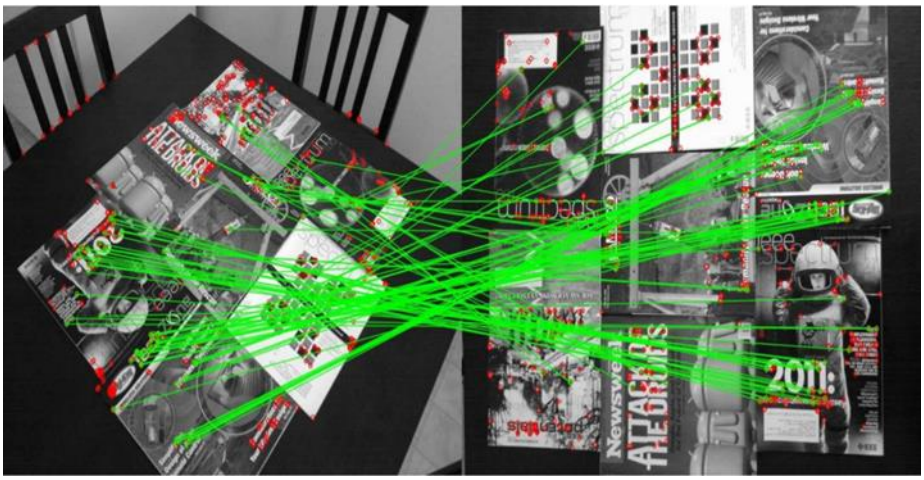


Şekil 3.6. Ekran, klavye, fare ve koruyucu kılıf

3.2 Yöntem

3.2.1. Yönelimli FAST ve Döndürülmüş BRIEF (Oriented Fast and Rotated Brief-ORB)

Yönelimli FAST ve Döndürülmüş BRIEF (Oriented Fast and Rotated Brief-ORB) (Rublee ve ark., 2011), 2011 yılında Ethan Rublee tarafından ortaya konmuş bir görsel öznitelik çıkarımı algoritmasıdır. Performansı ve eşleştirme verimliliğini arttırmak için değişikliklere sahip İkili Gürbüz Bağımsız Temel Özellikler (Binary Robust Independent Elementary Features-BRIEF) tanımlayıcısına ve Hızlandırılmış Segment Testi Özellikleri (Features from Accelerated Segment Test-FAST) köşe algılama algoritmasına dayanır. Değişmez Özellik Dönüşümü Ölçeği (Scale Invariant Feature Transform -SIFT). Yazarların bahsettiği gibi Hızlandırılmış Gürbüz Özellikler (Speeded Up Robust Features-SURF)'e etkili bir alternatiftir. Şekil 3.7 ORB eşleştirmesinin performansını göstermektedir.



Şekil 3.7. Orb öznitelik eşleştirme

3.2.1.1. Hızlandırılmış segment testi özellikleri (FAST)

FAST algoritması, 2006 yılında Edward Rosten ve Tom Drummond(Rosten ve Drummond, 2006) tarafından ileri sürülmüş öznitelik çıkarımı yapmak ve sonrasında bilgisayarla görme görevlerindeki nesneleri izlemek için kullanılabilen bir köşe algılama yöntemidir. En parlak özelliği hesaplama verimliliği ve Gaussians, SIFT, SUSAN ve HARRIS farkı gibi iyi bilinen algoritmalardan daha hızlı olmasıdır. Performansı artırmak ve hesaplama süresini kısaltmak için makine öğrenme tekniklerini kullanır. Bu özellikler onu yüz tanıma gibi gerçek zamanlı uygulamalar için uygun hale getirir.

a. Segment testi dedektörü

Algoritma, p noktasının bir köşe olup olmadığını sınıflandırmak için 3 yarıçaplı 16 piksellik bir Bresenham çemberi kullanır. 1'den 16'ya kadar saat yönünde etiketli çemberin içerisindeki her piksel için, bir N piksel kümesinin hepsi p pikselinin yoğunluğu Ip artı bir t eşik değerinden daha parlaksa veya p pikselinin yoğunluğu eksi bir t eşik değerinden daha koyuysa p köşe olarak sınıflandırılır. Durumlar aşağıda yazıldığı gibidir:

- Durum 1: N piksel kümesi S, $\forall x \in S$, x yoğunluğu (Ix) > Ip + t eşiği
- Durum 2: N piskel kümesi S, $\forall x \in S$, Ix < Ip - t

Karşılaşılan her iki durumda da p pikseli köşedir. N ve t eşik değerinin seçilmesinde bir denge vardır. Saptanan köşelerin sayısı çok fazla olmamalıdır ve yüksek performans hesaplama verimliliğinin düşürülmesiyle elde edilmemelidir, makine öğrenimi olmadan N genellikle 12 olarak seçilir. Köşe olmayan noktaları elemek için bir yüksek hız testi uygulanabilir.

b. Yüksek hız testi

4 örnek pikseli (piksel 1,9,5 ve 13) kontrol ederek tespit edilen köşe olmayan noktaları çıkarmak için yüksek hız testi kullanılır. Çünkü köşeden daha parlak ya da daha koyu olan en az 12 piksel olması gerekmektedir. Böylece bu 4 pikselden en az 3'ü köşeden daha parlak veya daha koyu olmalıdır. İlk olarak piksel bir ve dokuz incelenir, eğer I1 ve I9'un her ikisi de [Ip - t, Ip + t] içerisinde ise p pikseli köşe değildir. Aksi takdirde bu piksellerden üçünün Ip artı t eşiğinden daha parlak olup olmadığını veya Ip

eksi t eşiğinden daha koyu olup olmadığını kontrol etmek için 5 ve 13 pikselleri ayrıca test edilir.

Bu piksellerden 3 tanesi daha parlak veya daha koyu ise diğer pikseller daha sonra nihai saptama için değerlendirmeye alınır. Makalesinde algoritmanın yazarına göre köşe pikselini kontrol etmek için ortalama 3.8 piksel gerekir. Her bir köşe için 8.5 piksel ile kıyaslandığında 3.8, performansı büyük ölçüde artırabilecek önemli bir eksiltmedir. Fakat bu yöntemde çok sayıda zayıflık vardır:

1. Yüksek hız testi iyi genelleme yapamaz. N < 12 için. Eğer N <12 ise, bir p pikselinin köşe olması ve 4 test pikselinin sadece 2'sinin Ip artı t eşiğinden daha parlak olması veya Ip eksi t eşiğinden daha koyu olması mümkün olabilir.
2. Performansı metin piksellerinin seçimine ve sıralamasına bağlıdır
3. Birbirine yakın birden çok özellik tespit edilmektedir.

c. Makine öğrenme iyileştirmeleri

Yüksek hız testin ilk iki zayıf noktasını ele almak için makine öğrenme yaklaşımı, özellik algılama algoritmasını geliştirmek için kullanılırdı. Bu yöntem iki adımda çalışmaktadır:

İlk adım, hedef uygulama alanından daha iyi olan bir dizi alıştırma görüntüsü üzerinde işlenen bilinen bir N ile bir köşe tespitidir. Köşeler, 16 piksellik bir daire çıkarımı yapan ve uygun bir eşikle yoğunluk değerlerini karşılaştıran basit bir uygulama aracılığı ile tespit edilir.

P pikseli için çember $x \in)1, 2, 3, ..., 16 ($ üzerindeki her bir lokasyon $p \rightarrow x$ ile ifade edilebilir. Her bir pikselin statüsü, $Sp \rightarrow x$ aşağıdaki üç durumdan birinde olmalıdır:

1. d, $Ip \rightarrow x \leq Ip - t$ (daha koyu) (3.1)
2. s, $Ip - t \leq Ip \rightarrow x \leq Ip + t$ (aynı) (3.2)
3. b, $Ip \rightarrow x \geq Ip + t$ (daha açık) (3.3)

P_d, P_s, P_b'nin olduğu 3 ayrı alt kümede bir x (tüm p'lere eşdeğer) P bölümleri (tüm alıştırma görüntülerinin tüm piksellerinin kümesi) seçerek:

1. $P_d = \{p \in P : S_p \rightarrow x = d \}$ (3.4)
2. $P_s = \{p \in P : S_p \rightarrow x = s \}$ (3.5)
3. $P_b = \{p \in P : S_p \rightarrow x = b \}$ (3.6)

İkinci adım karar ağacı algoritmasıdır; En yüksek bilgi kazanımına ulaşmak için 16 lokasyonda ID3 algoritması uygulanır. Kp, p'nin köşe olup olmadığını gösteren bir Boolean değişkeni olsun ve sonrasında köşe olan p bilgisini ölçmek için Kp entropisi uygulansın. Q piksel kümesi için, KQ toplam entropisi (normalleştirilmemiş) şu şekildedir:

$H(Q) = (c + n) \log2(c + n)- c\log2c - n\log2n$ (3.7)

Koşul c = $| \{i \in Q$: Ki doğru$\}|$ (köşe sayısı) (3.8)

Koşul n = $| \{i \in Q$: Ki yanlış$\}|$ (köşe olmayanların sayısı) (3.9)

Bilgi kazancı daha sonra aşağıdaki gibi ifade edilebilir:

$Hg= H (P) - H (Pb) - H (Ps) - H (Pd)$ (3.10)

Bilgi kazanımını en üst düzeye çıkaracak her bir x'i seçmek amacıyla belirli bir alt kümeye özyinelemeli bir yöntem uygulanır. Örneğin; İlk önce bir x'in en fazla bilgi ile P'yi Pd, Ps, Pb'ye bölmesine karar verilir, sonrasında belirli bir alt küme, Pd, Ps, Pb için en fazla bilgi kazancını üretmek amacıyla başka bir y seçilir. Bu alt kümedeki piksellerin hepsi ya köşe olanlar ya da köşe olmayanlar haline gelsin diye bu özyinelemeli işlem entropi sıfır olduğunda sona erer.

d. Zayıf kenar noktalarını bastırma (Non-maximum suppression)

Bitişik lokasyonlardaki çeşitli ilgi noktalarının belirlenmesi başka bir konudur. Bu konu Zayıf Kenar Noktalarını Bastırma kullanılarak çözülmektedir. Bu aşağıda verilen kaba kod üzerinde gösterilmiştir.

	Non-Max suppression
1	Procedure NMS (B,c)
2	$\boldsymbol{B_{nms} \leftarrow \theta}$
3	for $\boldsymbol{b_i} \in B$ **do**
4	discard ← False
5	for $b_j \in B$ **do**
6	if same(b_j,b_i) > λ_{nms} then
7	If score(c,b_j) > score(c,b_i) then
8	discard ← True
9	If not discard then
10	$B_{nms} \leftarrow B_{nms}$ U b_i
11	Return B_{nms}

e. Yönelimli FAST (oFAST)

FAST algoritması kötü performans göstermektedir ve yönelimi hesaplamaz, bu yüzden bazı değişiklikler yapılmıştır ve yeni algoritma (oriented FAST-oFAST) olarak adlandırılmıştır. Bir yamanın yoğunluk merkezini merkezde bulunan bir köşe ile hesaplamaktadır. Bu köşeden merkeze doğru olan yön, yönelimi vermektedir. Yamanın boyutunun r olduğu dairesel bir r bölgesi içinde olması gereken x ve y ile momentler hesaplanır ve bu Denklem (3.11)'de verilmiştir:

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y) \tag{3.11}$$

Momentleri hesapladıktan sonra sendroidi Denklem (3.12) ile hesaplayabiliriz:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \tag{3.12}$$

Daha sonra köşe merkezi O'dan sendroid OC'ye kadar bir vektör oluşturarak yönelim ancak Denklem (3.13)'te verilir:

$$\theta = atan2(m_{01}, m_{10}) \tag{3.13}$$

3.2.1.2 İkili gürbüz bağımsız temel özellikler (BRIEF)

(BRIEF) tanımlayıcısı, bir dizi ikili yoğunluk testinden oluşturulan bir görüntünün bit dizgisi tanımıdır. Algoritma 2006 tarihli Lepetit, V., Fua, P.'ye ait "Randomize Edilmiş Ağaçlar Kullanan Anahtar Nokta Tespiti" ve 2010 tarihli Ozuysal, M., Calonder, M., Lepetit, V., Fua, P.'ye ait "Rasgele Fern'ler Kullanan Hızlı Anahtar Nokta Tespiti" çalışmalarından esinlenmiştir. Test sonuçları, çeşitli görüş noktalarından görülen yamaları tanımlamak için hem randomize edilmiş sınıflandırma ağaçları (Lepetit ve Fua, 2006) hem de Naive Bayesian sınıflandırıcı (Ozuysal ve ark., 2010) alıştırması yapmak için kullanılmıştır. Algoritma, ya sınıflandırıcıyı ya da ağaçları yürürlükten kaldırır ve görüntü yamasını düzleştirdikten sonra hesaplama yapan test

yanıtlarından ikili bir vektör oluşturur. Test, SxS boyutundaki p yamasında τ olarak tanımlanmakta ve Denklem (3.14)'te verilmektedir:

$$\tau(p;x,y) := \begin{cases} 1 : p(x) < p(y) \\ 0 : p(x) \geq p(y) \end{cases} \quad (3.14)$$

p(x)'in x= (u,v)⊤'de p versiyonunun düzleştirilmiş versiyonundaki piksel yoğunluğu olduğu koşulunda.

Bir nd (x, y) pozisyon çiftinin seçilmesi, bir dizi ikili testi benzersiz şekilde saptar ve BRIEF tanımlayıcısını n boyutlu bir bit dizgisi olarak alır:

$$f_{nd}(p) := \sum_{1 \leq i \leq nd} 2^{i-1} \ \tau(p; x_i, y_i) \quad (3.15)$$

BRIEF algoritmasının yazarı, ölçüyü 128,256 ve 512 olarak kabul eder. Bu opsiyonlar, hız, depolama verimliliği ve tanıma oranı arasındaki uzlaşıları, ödünleşmeleri açığa çıkarır.

Makaledeki yazarlar k = nd/8 'in tanımlayıcıyı depolamak için gereken bayt sayısını gösterdiği yerdeki BRIEF tanımlayıcılarını BRIEF-k olarak adlandırırlar. Bu tür tanımlayıcıları oluştururken gereken tek seçenek, (x, y) çiftlerinin uzamsal düzenlemesi ve yoğunluk farklarından önce yamaları düzeltmek için kullanılan çekirdeklerdir.

3.2.1.3 Döndürülmüş BRIEF (rBRIEF)

ORB, BRIEF tanımlayıcısını kullanmaktadır fakat BRIEF rotasyon ile yetersiz performans göstermektedir. ORB yazarları bazı değişiklikleri ele almışlardır. ORB, oFAST kullanarak çıkarım yapan anahtar noktaların yönelimine göre BRIEF'i döndürür. (x, y) konumundaki n ikili testlerin herhangi bir özellik kümesi için piksel koordinatlarını içeren S'yi, $2\,x\,n$ matrisini tanımlar ve sonrasında θ yamasının yönelimini kullanarak Sθ döndürülmüş versiyonunu elde etmek için S'yi döndürür. ORB, açıyı 2π/30'luk artış değerlerine ayırır ve yeniden hesaplanan BRIEF modellerini içeren bir başvuru çizelgesi oluşturur. Anahtar noktaların yönelimi θ görünümler boyunca uyumlu olduğu sürece doğru nokta kümesi Sθ, tanımlayıcısını hesaplamak için uygulanacaktır.

3.2.2. Temel bileşenler analizi (PCA)

Temel Bileşenler Analizi (Principal Component Analysis-PCA), değişiklik algılama konuları, görüntü iyileştirme görevleri ve daha fazlası gibi Uzaktan Algılama çalışmalarında yaygın olarak kullanılan bir boyutluluk azaltma yöntemidir. Aslında

PCA son derece ilişkili çok boyutlu veriler üzerinde uygulanan doğrusal bir dönüşümdür. Girdi boyutları, verilen boyutların (temel bileşenler olarak adlandırılan) azalan sırayla kalıcı en-boy özellikleriyle ilgili en önemli çeşitliliği içerdiği yeni bir koordinat sistemine dönüştürülür. PCA algoritması iki cebirsel açıklamaya sahiptir, bunlar Kovaryans Özvektörler ve verilen veri matrisin sayısal kesinliği için kullanılan SVD tekil değer çıkarımıdır. PCA 1901'de Karl Pearson tarafından (Pearson, 1901) mekanikte asal eksen teoreminin bir analoğu olarak icat edilmiş ve daha sonra 1930'ların başında Harold Hotelling tarafından geliştirilmiştir (Hotelling, 1933). Amaç alanına bağlı olarak sinyal işlemede ayrık KLT, çok değişkenli kalite kontrolde Hotelling dönüşümü, makine mühendisliğinde Uygun Ortogonal Ayrıştırma (Proper Orthogonal Decomposition-POD), doğrusal cebirde X'in SVD, XTX'in Eigenvalue Ayrışımı (Eigenvalue Decomposition-EVD), meteoroloji biliminde faktör analizi, Eckart-Young teoremi veya ampirik dikgen fonksiyonlar Deneysel Ortogonal Fonksiyonlar (Empirical Orthogonal Functions-EOF) gürültü ve titreşimde ampirik özişlev ayrışımı (Sirovich ve Kirby, 1987), ampirik bileşen analizi, yarı harmonik modlar, spektral ayrışım ve yapısal dinamiklerde ampirik modal analiz olarak adlandırılır.

Keşifsel veri analizinde ve öngörücü kodlamaların oluşturulması için PCA'dan bir araç olarak sıklıkla faydalanılmaktadır. Genellikle gruplar arasındaki genetik mesafeyi ve ilişkiyi görselleştirmek için kullanılmaktadır. PCA, bir veri kovaryans matrisinin özdeğer ayrışımı veya bir veri matrisinin tekil değer ayrışımı ile genellikle Z-skorlarını ortalama olarak topladıktan ve normalleştirdikten veya her özellik için veri matrisine uyguladıktan sonra gerçekleştirilebilir. Bir PCA'nın sonuçları, nadiren faktör puanları olarak adlandırılan bileşen puanları ile ilgili olarak ve bileşen puanını elde etmek için her bir orijinal standardize değişkenin çarpılması gereken ağırlığı depolayarak düzenli bir şekilde ele alınmaktadır. Bileşen puanları birim varyasyonuna standardize edilmişse dolgular bunlardaki veri varyasyonunu içermelidir ve bu özdeğerlerin büyüklüğüdür. Bileşen puanları standardize edilmemişse sonuç olarak bunlar veri çeşitliliğini ve sonrasında birim ölçekli olması gereken dolguları içerir ve bu ağırlıklar özvektörler olarak ifade edilmektedir, bunlar başlıca olana değişkenlerin dikgen rotasyonunun kosinüsleridir.

PCA, doğru özvektör bazlı çok değişkenli analizlerin basite indirgenmişidir. Bunun işleyişi sıklıkla verilerdeki varyansı layıkıyla resmedecek şekilde verilerin iç

yapısını açığa çıkarma olarak düşünülebilir. Çok değişkenli bir veri seti, yüksek boyutlu veri alanındaki bir dizi koordinat olarak hayal edildiğinde PCA, kullanıcıya en bilgilendirici bakış açısıyla bakıldığında bu nesnenin bir yansıması olan daha düşük boyutlu bir temsil sağlayabilmektedir. Bu, dönüştürülen verilerin boyutluluğu azalsın diye yalnızca ilk birkaç temel bileşeni uygulayarak başarılır.

PCA kesinlikle faktör analizi ile ilişkilidir. Faktör analizi tipik olarak temeldeki yapı ile ilgili daha fazla alana özgü varsayımları birleştirir ve anlaşılamaz derecede farklı bir matrisin özvektörlerini belirler.

PCA, kanonik korelasyon analizi (Canonical Correlation Analysis-CCA) ile daha ilgilidir. PCA, tek bir veri setinde varyasyonu en iyi şekilde temsil eden farklı bir dikgen koordinat sistemi atarken CCA, iki veri kümesi arasındaki çapraz kovaryansı en iyi şekilde temsil eden koordinat yöntemlerini belirlemektedir (Tong; Barnett ve Preisendorfer, 1987).

PCA matematiksel olarak verileri yeni bir koordinat sistemine dönüştüren dikgen bir doğrusal dönüşüm olarak karakterize edilmektedir. Öyle ki, verilerin belirginliği ile en anlamlı çeşitlilik, birinci temel bileşen, ikinci koordinattaki en yüksek ikinci varyans ve bunun gibi isimlendirilen birinci koordinata dayanmaktadır. (Jolliffe, 2011) n-satırlarının her birinin deneyin farklı bir tekrarını gösterdiği her yerde her bir sütunun örneklem ortalamasının sıfıra kaydırıldığı sütun yönünden sıfır deneysel ortalamaya sahip bir veri matrisi olan X varsayalım ve p sütunlarının her biri uygun bir özellik sağlamaktadır.

3.2.2.1. Özyüzler (EigenFaces)

Özyüzler, bir dizi yüz görüntüsü arasındaki varyasyona dayanan ve özellik tabanlı bir yöntemin aksine yüz görüntülerini karşılaştırmak için bu farklılıkları kullanan görünüm tabanlı bir yüz tanıma yöntemidir. Özyüzler yüz görüntüleri koleksiyonunun temel bileşenleridir. Özel olarak bunun özvektörler kümesi 1987 yılında Sirovich ve Kirby sunulmuş ve M.Turk ve A.Pentland tarafından geliştirilmiştir (Turk ve Pentland, 1991). Yüz görüntülerinin özvektörleri, alıştırma görüntülerinin yüksek boyutlu vektör uzayı üzerindeki olasılık dağılımının kovaryans matrisinden elde edilir. Yüz görüntülerinin özvektörleri, alıştırma görüntülerinin yüksek boyutlu vektör uzayı üzerindeki olasılık dağılımının kovaryans matrisinden elde edilir. Özyüzler kavramı, salınımlı fonksiyonların bir toplamı olarak Fourier ayrışımındaki periyodik

sinyal temsili ile benzerdir. Özyüzler hesaplanmadan önce tüm alıştırma görüntüleri, gözlerin ve ağızların hizalanması için normalleştirilmeli ve daha sonra aynı piksel çözünürlüğünde yeniden örneklenmeli ve sonrasında ana bileşenler analizi kullanılarak özyüzlerin aşağıdaki şekilde görüntülerden çıkarımı yapılmalıdır:

1. $h \times w$, M boyutunda bir resim dizisi verildiğinde her resim D(=*hw*) boyutlu bir vektöre dönüştürülür: Denklem (3.16)

$\{\Gamma_1, \Gamma_2, \dots \dots, \Gamma_M\}$

(3.16)

Tüm görüntüler ölçeklenmeli ve hizalanmalıdır.

2. Ortalama yüz vektörünün aşağıdaki gibi tanımlandığı koşullarda her bir yüz görüntüsü ortalamadan Φi=Γi−Ψ vektörü ile ayrılır: Denklem (3.17)

$\Psi = \frac{1}{M}\sum_{i=1}^{M} \Gamma_i$ (3.17)

3. Kovaryans matrisi $\mathbf{C} \in R^{D \times D}$ Denklem (3.18) ve (3.19)'daki gibi tanımlanmaktadır:

$C = \frac{1}{M}\sum_{i=1}^{M} \Phi_i \Phi_i^{\intercal} = AA^{\intercal}$

(3.18)

$A = \{\Phi 1, \Phi 2, \cdots, \Phi M\} \in RD \times M$ (3.19)

4. D≫M olduğunda C'nin özvektörlerini tanımlamak, tipik görüntü boyutları için zor bir görevdir. Bu sorunu çözmek için önce çok daha küçük MxM matrisi AᵀA'nın özvektörleri hesaplanır. Özvektör matrisleri Denklem (3.20)'deki gibi tanımlanır:

$V = \{v_1, v_2, \dots \dots, v_r\}$ (3.20)

Dahası, r'nin A kertesi olduğu koşulda ;

Λ=diag{λ1,λ2,⋯λr} , λ1≥λ2≥⋯λr>0

(3.21)

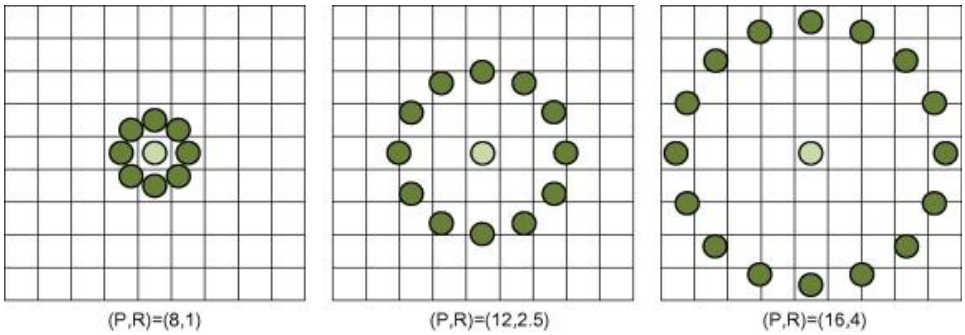
5. Özyüzlerin koleksiyonunun U={ui} olduğu koşulda C'nin özvektörleri ve özdeğerleri Λ ve U=AVΛ−1/2'dir.

3.2.3. Yerel ikili örüntü (LBP)

Yerel İkili Örüntü (Local Binary Patterns-LBP), her pikselin komşuluğunu eşikleyerek görüntünün piksellerini tanımlayan ve sonucu ikili sayı olarak hesaplayan kullanışlı bir doku operatörüdür. Ayırt edici gücü ve hesaplama sadeliği nedeniyle LBP doku operatörü sayısız uygulamada basitleştirilmiş bir yaklaşım haline gelmiştir. Doku analizinin geleneksel olarak farklı istatistiksel ve yapısal modellerine birleştirici bir yaklaşım olarak görülebilir. Belki de LBP operatörünün gerçek dünya amaçlarındaki temel özelliği, örneğin aydınlatma varyasyonlarından kaynaklanan monoton gri ölçekli varyasyonlara karşı gürbüzlüğüdür. Diğer bir değerli özelliği, zorlu gerçek zamanlı ortamlarda görüntüleri analiz etmeyi mümkün kılan hesaplama basitliğidir.

LBP operatörünü iyileştirmek için esas fikir, iki boyutlu yüzey dokularının denk düşen iki ölçü ile karakterize edilebilmesidir: Yerel mekansal örüntüler ve gri ölçek varyasyonu. Birincil LBP operatörü (Ojala ve ark., 1996) merkez değer arasındaki piksel başına 3 x 3 komşuluğu eşikleyerek ve sonucu ikili bir sayı olarak değerlendirerek görüntü pikselleri için etiketler üretir. Bu $2^8 = 256$ çeşitli etiketlerin histogramı daha sonra doku tanımlayıcısı olarak kullanılabilir. Basit bir yerel fark ölçüsü ile birlikte kullanılan bu operatör, denetimsiz doku bölütlemesinde mükemmel bir performans ortaya koymuştur (Ojala ve Pietikäinen, 1999). Bunun akabinde doku ve renk dokusu bölütlemesi için birkaç tamamlayıcı yaklaşım geliştirilmiştir.

LBP operatörü, çeşitli ebatlardaki komşulukları kullanmaya devam etmiştir. (Ojala ve ark., 2002). Dairesel komşuluk kullanarak ve tamsayı olmayan piksel koordinatlarındaki değerleri çiftdoğrusal olarak içererek komşuluktaki herhangi bir yarıçapı ve piksel sayısını düşer. Yerel komşuluğun gri ölçek varyasyonu, eşit varyans ölçüsü olarak uygulanabilir. Aşağıda, notasyon (P, R), R yarıçapındaki bir dairede P örneklem noktalarını ifade eden piksel komşulukları için uygulanacaktır. Şekil 3.8'de LBP hesaplamasının bir örneğini göstermektedir.



Şekil 3.8. LBP hesaplaması örneği

$$LBP_{P,R} = \sum_{p=1}^{p} 2^{(p-1)} X(G_p - G_c) \quad (3.22)$$

$$X(x) = \begin{cases} 1, x \geq 0 \\ 2, else \end{cases} \quad (3.23)$$

Gc, orta pikselin gri değeriyse, Gp, komşularının gri değeridir. P, komşuların sayısıdır ve R, çevrenin yarıçapıdır.

Birincil operatöre ekstra bir ekleme, öznitelik vektörünün uzunluğunu azaltmak ve basit bir rotasyon-değişimsiz tanımlayıcı gerçekleştirmek için uygulanabilen sözde tek biçimli görüntülerin belirlenmesidir. Bu devamlılık, göze çarpan ikili örüntülerin doku görüntülerinde diğerlerinden daha genel olarak meydana gelmesiyle motive olmuştur. Bit örüntüsü dairesel olarak çaprazlandığında ikili örüntü 0'dan 1'e veya tam tersi en çok iki bitsel geçişi içeriyorsa yerel bir ikili örüntü birbiçimli olarak adlandırılır. Örneğin, 00000000 (0 geçiş), 01110000 (2 geçiş) ve 11001111 (2 geçiş) örüntüleri birbiçimli iken, 11001001 (4 geçiş) ve 01010010 (6 geçiş) örüntüleri değildir. LBP etiketlerinin hesaplanmasında, her bir birbiçimli örüntü için ayrı bir etiket olacak şekilde birbiçimli örüntülerden faydalanılır ve tüm birbiçimli olmayan örüntüler tek bir etiket ile etiketlenir. Örneğin; (8, R) komşuluğunu kullanırken, 58'i tek biçimli olan ve 59 çeşitli etiketi açığa çıkaran toplam 256 örüntü bulunmaktadır.

(Ojala ve ark., 2002), doku görüntülerine sahip deneylerinde (8,1) komşuluğu kullanırken tek biçimli örüntülerin tüm örüntülerin %90'ından biraz azı için tahmin edildiğini ve (16,2) komşuluğunda bunun yaklaşık %70 olduğunu belirtmiştir. Ayrı olarak ikilik (LBP kodu) bir mikro-tekston olarak gözlemlenebilir. Bu ikilikler tarafından düzenlenen yerel ilkeller çeşitli kavisli kenarları, noktaları, düz alanları vb. kapsamaktadır.

Bir sonraki notasyon LBP operatörüne uygulanır: LBP P, R u2. Ek, operatörün (P, R) komşuluğunda uygulanmasını açıklamaktadır. Üst simge u2, sadece birbiçimli örüntülerden faydalanmak ve tüm kalan örüntüleri tek bir etiketle etiketlemek için erişir. Daha sonra LBP etiketli görüntü fl(x, y) elde edilmiştir ve LBP histogramı aşağıdaki tanımlanabilir:

$$H_i = \sum_{x,y} I\{f_{l(x,y)} = i\}, i = 0, \ldots, n-1 \quad (3.24)$$

Burada n, (LBP) operatörü tarafından sağlanan çeşitli etiketlerin sayısıdır ve I{A}, A doğru ise bir, A yanlış ise sıfırdır.

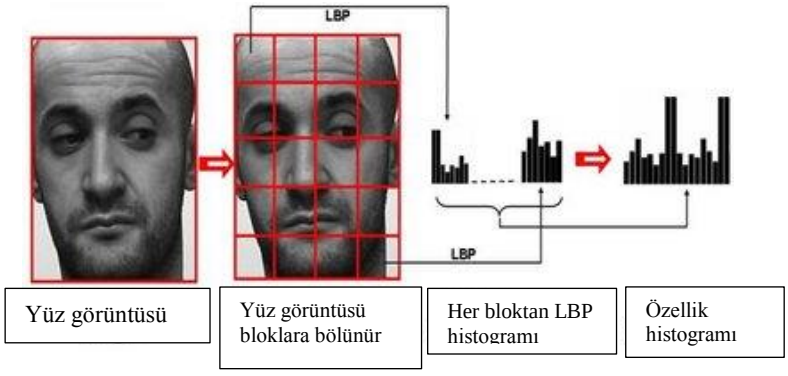
Histogramları incelenecek olan resim yamaları farklı boyutlarda olduğunda tutarlı bir tanım elde etmek için histogramlar normalleştirilmelidir:

$$N_i = \frac{H_i}{\sum_{j=0}^{n-1} H_j} \quad (3.25)$$

3.2.3.1. LBP kullanarak yüz tanımlama

Doku analizi için olan LBP yönteminde bir görüntüdeki LBP ilkelerinin olguları bir histogramda ele alınır. Sınıflandırma daha sonra basit histogram ilişkileri hesaplanarak gerçekleştirilir. Buna rağmen, yüz görüntüsü temsili için ayrıntılı bir öneriyi kabul etmek uzamsal bilgi kaybına neden olur ve bu nedenle bunların konumlarını korurken doku bilgisini düzenlemek gerekir. Bu amacı gerçekleştirmenin bir yolu, LBP doku tanımlayıcılarını birkaç yerel yüz tanımı oluşturmak ve bunları global bir tanımlamayla katmak için uygulamaktır. Bu tür yerel tanımlar, bütüncül temsillerin koşullarına makul şekilde gösterilen ilgiyi son zamanlarda arttırmıştır. Bu yerel özellik tabanlı sistemler, poz veya aydınlatmadaki varyasyonlar üzerinde bütünsel yöntemlerden daha gürbüzdür.

(Ahonen ve ark., 2006) tarafından sunulan LBP tabanlı yüz tanımlama için birincil metodoloji aşağıdaki gibidir: Yüz görüntüsü yerel alanlara ayrılmakta ve LBP doku tanımlayıcıları her bölgeden ayrı ayrı elde edilmektedir. Tanımlayıcılar daha sonra Şekil 3.9'da da gösterildiği gibi yüzün genel bir tanımını oluşturmak için birleştirilmektedir.



Şekil 3.9. LBP kullanarak yüz tanımlama

Bu histogram mutlaka lokalitenin üç farklı düzeyinde yüz tanımına sahiptir: Histogram için LBP etiketleri piksel seviyesindeki örüntüler ile ilgili bilgi içerir, etiketler lokalize bir düzeyde bilgi sağlamak için küçük bir alan boyunca toplanır ve lokalize histogramlar yüzün global bir tanımını üretmek amacıyla birleştirilir.

Histogram tabanlı teknikleri uygularken bölgelerin dikdörtgen olması gerekmediği belirtilmelidir. Eşdeğer boyuta veya şekle sahip olmaları gerekmez ve doğrusu tüm görüntüyü kapsamaları gerekmez. Ayrıca az çok üst üste binen bölgelere sahip olmakta mantıklıdır.

İki boyutlu yüz tanımlama yöntemi, zaman-mekansal alana devam etmiştir (Zhao ve Pietikainen, 2007). Bu yaklaşımla en iyi yüz ifadesi tanıma performansı elde edilmiştir.

LBP temelli yüz tanımının beyanından bu yana metodoloji evvelce yüz analizi araştırması ve uygulamalarında köklü bir yer edinmiştir.(Near Infrared-NIR) Görüntülemeyi LBP özellikleri ve Adaboost öğrenmesi arasında birleştiren (Liao ve ark., 2007) tarafından sunulan aydınlatma-değişimsiz yüz tanıma sistemi iyi bilinen bir örnektir. Zhang ve ark. (2005), mükemmel sonuçlar elde ederek çeşitli ölçeklerde ve yönlerde 40 Gabor filtresi ile yüz görüntüsünü filtreleyerek elde edilen görüntülerden LBP özelliğinin inişini ortaya koymuştur. (Zhao ve ark., 2009) hareketli dudakların hata yapmaya açık bölütlemesi olmaksızın ön kenar performansı sağlayan görsel konuşma tanımaya yönelik LBP-TOP yaklaşımını kullanırken (Hadid ve Pietikäinen, 2009) video dizinlerinden yüz ve cinsiyet tanıma için zaman-mekansal LBP'leri uygulamıştır.

3.2.4. Yüz veri seti

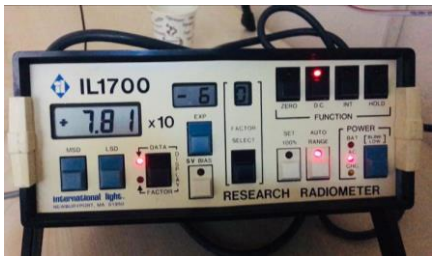
Tez çalışmasında 30 kişiden oluşan bir veritabanı seti oluşturulmuştur. Veri seti oluşturulurken ortamın ışık seviyesi, mesafe, yüz detayı ve kameraya bakış durumu gözönüne alınmıştır. Ortamın ışık seviyesi az ışık ($9{,}12.10^{-6} W/m^2$), orta ışık ($7{,}81.10^{-6}$ W/m^2), çok ışık ($3{,}12.10^{-5}$ W/m^2) olmak üzere 3 farklı ışık seviyesi olarak belirlenmiştir (Şekil 3.10). Kamera kullanılarak görüntüler elde edilirken ortamın ışık seviyesi, Şekil 3.11'de görülen mikroişlemci kontrollü UV ışık ölçme cihazı olan IL1700 Research Radiometer ile ölçülmüştür.



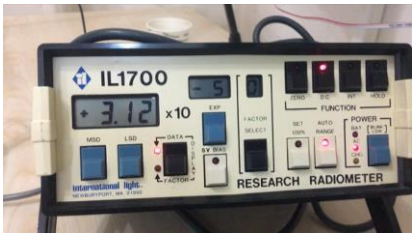
Şekil 3.10. Farklı ışık seviyelerinde aynı mesafeden, aynı açıdan, gözlüksüz alınmış görüntüler



a)



b)



c)



d)

Şekil 3.11 a) IL1700 Radiometer UV ışık ölçme cihazı **b)** orta ışık ölçümü **c)** Çok ışık ölçümü **d)** Az ışık ölçümü

Mesafeler seçilirken algoritmaların kapı geçişinde yüz tanıma amacı ile kullanılacağı dikkate alınarak 4 farklı (50cm, 100cm, 150cm, 200cm) mesafe belirlenmiştir (Şekil 3.12).



Şekil 3.12. 4 Farklı mesafeden, aynı ışık seviyesinde, aynı açıdan, gözlüksüz alınmış görüntüler

Yüz detayının etkisini değerlendirebilmek için her kişiden gözlüklü ve gözlüksüz olmak üzere 2 farklı görüntü alınmıştır (Şekil 3.13).



Şekil 3.13. Aynı mesafeden, aynı ışık seviyesinde, aynı açıdan alınmış gözlüklü ve gözlüksüz görüntüler

Sistemi kullanacak kişilerin kameraya farklı açılardan bakabileceği de gözönüne alınarak Şekil 3.14'te görüldüğü gibi 5 farklı durum (Ön, Yukarı, Aşağı, Sol, Sağ) için görüntü alınmasına dikkat edilmiştir.



Şekil 3.14. Aynı mesafeden, aynı ışık seviyesinde, gözlüksüz, 5 farklı açı için alınmış görüntüler

Tüm bu özellikler gözönüne alındığında, her kişi için 120 farklı görüntü elde edilmiştir. 30 kişilik gruptan oluşan veri seti toplamda 3600 görüntü içermektedir.

Yüz tanıma testleri bu görüntülere ORB, PCA ve LBP algoritmaları uygulanarak gerçekleştirilmiştir. Şekil 3.15'te veri setini oluşturan 30 kişiye ait bazı görüntü örnekleri verilmektedir.





Şekil 3.15. Veri setini oluşturan kişilerin farklı görüntü örnekleri

3.2.5 Linde–Buzo–Gray algoritması (LBG)

(Linde–Buzo–Gray-LBG) algoritması, Yoseph Linde, Andrés Buzo ve Robert M. Gray tarafından 1980'de (Linde ve ark., 1980) tanıtılan K-Ortalama algoritması gibidir; bu, S = {xi is Rd | i = 1,2, ..., n} giriş olarak ve C = {cj ∈ Rd | j = 1,2, ..., K},

benzerlik ölçüsüne göre çıkış olarak K << n olarak belirtilen bir kullanıcıyla. Vektör Miktarı VQ uygulaması için, d = 16, K = 256 veya 512 yaygın olarak kullanılır.

LBG algoritması aşağıdaki gibi çalışır:

1. Giriş eğitimi vektörleri

$$S = \{x_i \in R_d \mid i = 1, 2, \cdots, n\}. \quad (3.26)$$

2. Bir kod defteri başlatın

$$C = \{c_j \in R_d \mid j = 1, 2, \cdots, K\}. \quad (3.27)$$

3. Set

$D_0 = 0$ ve $k = 0$. (3.28)

4. N eğitim vektörlerini K kümelerine göre sınıflandırın.

$$x_i \in S_q \; if \; \|x_i - c_q\|_p \leq \|x_i - c_j\|_p \; for \; j \neq q. \quad (3.29)$$

5. Küme merkezlerini güncelle

$$c_j, j = 1,2, \ldots., K \; by \; c_j = \frac{1}{|S_j|} \sum_{x_i \in S_j} x_i. \quad (3.30)$$

6. Set $k \leftarrow k + 1$ ve bozulmayı hesaplamak

$$D_k = \sum_{j=1}^{K} \sum_{x_i \in S_j} ||x_i - c_j||_p \quad (3.31)$$

7. Eğer $\frac{D_{k-1} - D_k}{D_k} > ǫ$ (küçük bir numara), adımları tekrarla 4 ~ 6.

8. Kod defterini çıkar

$$C = \{c_j \in R_d \mid j = 1, 2, \cdots, K\}, \quad (3.32)$$

LBG algoritmasının yakınsaklığı, ilk kod kitabı C'ye, bozulmaya, Dk'ye ve eşiğe bağlıdır. Uygulamada, yakınsamayı garanti etmek için maksimum sayıda yineleme sağlanmalıdır.

3.2.6 K-en yakın komşu (KNN)

Örüntü tanımada, K-En Yakın Komşu (k-Nearest Neighbors-KNN), sınıflandırma ve regresyon için kullanılan parametrik olmayan bir yöntemdir (Altman, 1992) . Her iki durumda da, giriş özelliği uzayda k en yakın eğitim örneklerinden oluşmaktadır. Çıktı, KNN'nin sınıflandırma veya regresyon için kullanılmasına bağlıdır.

KNN sınıflamasında, çıktı bir sınıf üyeliğidir. Bir nesne, komşularının çok sayıda oyu ile sınıflandırılır. Nesne, en yakın komşuları arasında en yaygın sınıfa atanır (k, tipik olarak küçük olan bir tamsayıdır). Eğer k = 1 ise, nesne basitçe en yakın

komşunun sınıfına atanır. KNN regresyonunda çıktı, nesnenin özellik değeridir. Bu değer, en yakın komşusunun k değerlerinin ortalamasıdır.

KNN, örnek tabanlı öğrenme veya tembel öğrenme türüdür; burada işlev yalnızca yerel olarak yaklaşılır ve tüm hesaplamalar sınıflandırmaya kadar ertelenir. KNN algoritması, tüm makine öğrenimi algoritmalarının en basit olanlarından biridir. Hem sınıflandırma hem de regresyon için, komşuların katkılarına ağırlık vermek için yararlı bir teknik kullanılabilir, böylece daha yakın komşular daha uzak olanlardan daha ortalamaya daha fazla katkıda bulunur. Örneğin, ortak bir ağırlıklandırma şeması, her komşuya 1/d'lik bir ağırlığının verilmesini içerir; Burada d komşuya olan uzaklıktır . Komşular, sınıfın KNN sınıflandırması için veya nesne özellik değerinin KNN regresyonu için bilindiği bir dizi nesneden alınır. Bu, algoritma için ayarlanmış eğitim olarak düşünülebilir, ancak açık bir eğitim basamağı gerekmemektedir.

3.2.7 Haar Cascade sınıflandırıcı

Haar benzeri özellikler, nesne tanımada kullanılan dijital görüntü özellikleridir. İsmini Haar dalgacıkları ile olan sezgisel benzerliklerinden alır ve ilk gerçek zamanlı yüz dedektöründe kullanılmıştır (Viola ve Jones, 2001).

Tarihsel olarak, yalnızca görüntü yoğunluklarıyla (yani, her bir resmin pikselinde RGB piksel değerleri) çalışmak, özellik hesaplaması görevini hesaplamalı olarak pahalı hale getirilmiştir. Papageorgiou ve diğerleri tarafından bir yayın (Papageorgiou ve ark., 1998). yerine olağan görüntü yoğunlaşmaları Haar dalgacıkları dayalı alternatif bir özellik seti ile çalışması tartışılmıştır. Viola ve Jones (Viola ve Jones, 2001) Haar dalgacıkları kullanma fikrini uyarlamış ve sözde Haar benzeri özellikleri geliştirmiştir. Haar benzeri bir özellik, algılama penceresinde belirli bir konumdaki bitişik dikdörtgen bölgeleri dikkate alır, her bölgedeki piksel yoğunluğunu özetler ve bu toplamlar arasındaki farkı hesaplar. Bu fark daha sonra bir görüntünün alt bölümlerini kategorize etmek için kullanılır. Örneğin, insan yüzleri olan bir görüntü veri tabanı var diyelim. Tüm yüzler arasında gözlerin bölgesinin yanakların bölgesinden daha koyu olduğu yaygın bir gözlemdir. Bu nedenle yüz tespiti için yaygın bir Haar özelliği, göz ve yanak bölgesinin üzerinde yatan iki bitişik dikdörtgenden oluşan bir kümedir. Bu dikdörtgenlerin konumu, hedef nesneye sınırlayıcı bir kutu gibi davranan bir algılama penceresine göre tanımlanır (bu durumda yüz). Viola — Jones nesne algılama çerçevesinin algılama aşamasında, hedef boyuttaki bir pencere girdi

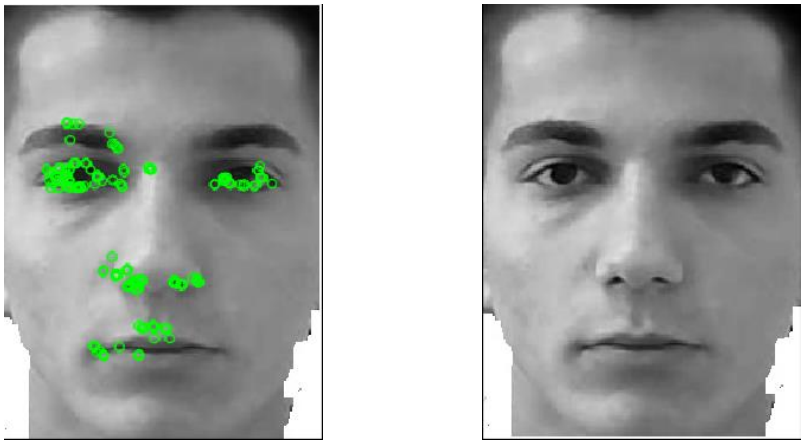
görüntüsünün üzerine taşınır ve görüntünün her alt bölümü için Haar benzeri özellik hesaplanır. Bu fark daha sonra nesnelerden olmayan nesneleri ayıran öğrenilen bir eşik ile karşılaştırılır. Böyle bir Haar benzeri özellik sadece zayıf bir öğrenci veya sınıflandırıcı olduğundan (algılama kalitesi rastgele tahminlerden biraz daha iyidir) bir nesneyi yeterli doğrulukla tanımlamak için çok sayıda Haar benzeri özellik gereklidir. Viola — Jones nesne algılama çerçevesinde, Haar benzeri özellikler güçlü bir öğrenci veya sınıflandırıcı oluşturmak için kaskad adı verilen sınıflandırıcı düzenlenir. Haar benzeri bir özelliğin diğer birçok özelliğe göre en önemli avantajı hesaplama hızıdır. Entegre görüntülerin kullanımı nedeniyle, her boyuttaki Haar benzeri bir özellik sabit zamanda hesaplanabilir (2 dikdörtgen özellik için yaklaşık 60 mikroişlemci talimatı).

3.2.8. Önerilen sistem

Bu tezde önerilen üç algoritmayı uygulamak için Python programlama dili ve Opencv bilgisayar vizyon kütüphanesi kullanmıştır.

3.2.8.1. ORB'ye dayalı yüz tanıma

Arka plandaki görüntülerden gelen yüzleri algılamak için Haar cascade sınıflandırıcı kullanılmıştır. Yüz görüntüsünden ilgi özelliklerini ayıklamak için ORB algoritması ve daha sonra LBG kümeleme kullanarak tanımlayıcıları bir görsel kelimeler modeli oluşturmak için küme haline getirilmiştir. KNN sınıflandırıcıyı eğitir. Şekil 3.16'da örnek bir yüz görüntüsünden anahtar noktalarının çıkarılmasını göstermektedir.

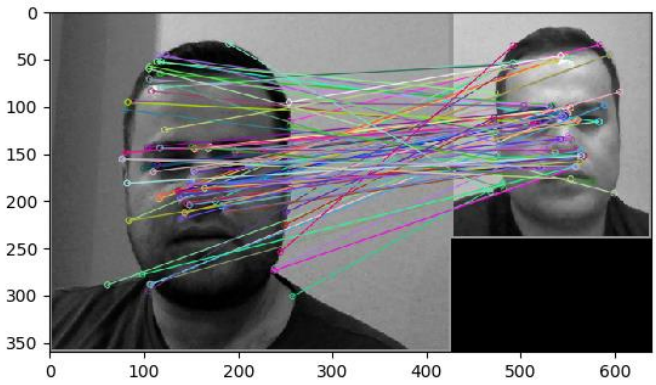


a) **b)**

Şekil 3.16 a) Orb kullanılarak tespit edilen görüntü **b)** Orijinal görüntü

Şekil 3.17 ORB algoritması kullanılarak yüzler arasındaki özellik eşleşmesi gösterilmektedir. Şekil 3.18'de önerilen ORB yüz tanıma sisteminin akış şemasını

göstermektedir. ORB algoritması kullanılarak eğitim görüntüleri ve özüt özellikleri alınmıştır ve daha sonra kod çizelgesini oluşturmak için LBG yapılmıştır ve ardından KNN sınıflandırıcısını eğitmiştir.



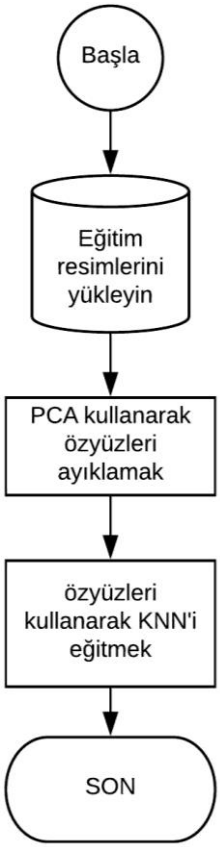
Şekil 3.17. Orb'nin eşleşmesi özellikleri



Şekil 3.18. ORB bazlı yüz tanıma sistemi akış şeması

3.2.8.2. PCA'ya dayalı yüz tanıma

Eğitim görüntülerinin boyutsal olarak küçültülmesi için PCA kullanılmıştır ve KNN Özyüzler üzerinde özellikler olarak eğitilmiştir. Şekil 3.19'da PCA tabanlı yüz tanıma sistemi akış şeması gösterilmiştir.



Şekil 3.19. PCA tabanlı yüz tanıma sistemi akış şeması

3.2.8.3. LBP'ye dayalı yüz tanıma

LBP'ye dayalı yüz tanıma sistemini uygulamak için python ve Skimage kütüphanesi kullanılmıştır. LBP görüntüleri eğitim görüntülerinden ayıklamak için kullanılmıştır. Her bir görüntü için LBP kodunun histogramı hesaplanmıştır ve daha sonra KNN sınıflandırıcısı ile eğitilmiştir. Şekil 3.20'de işlemin akış şeması gösterilmektedir.



Şekil 3.20. LBP tabanlı yüz tanıma sistemi akış şeması

3.2.8.4 Yüz tanıma donanımı

3B baskılı plastik kasada Lattepanda Computer-on-module ve 720p Resolution Webcam kullanılmıştır. Gerçek zamanlı yüz rengini düzeltmek ve gömülü sisteme uygulanan 3 algoritmayı test etmek için kullanılmıştır. Şekil 3.21'de kamera modülü ve IPS ekranı ile plastiğin içine monte edilen son sistem gösterilmektedir. Şekil 3.22'de cihazı içeriden göstermektedir.



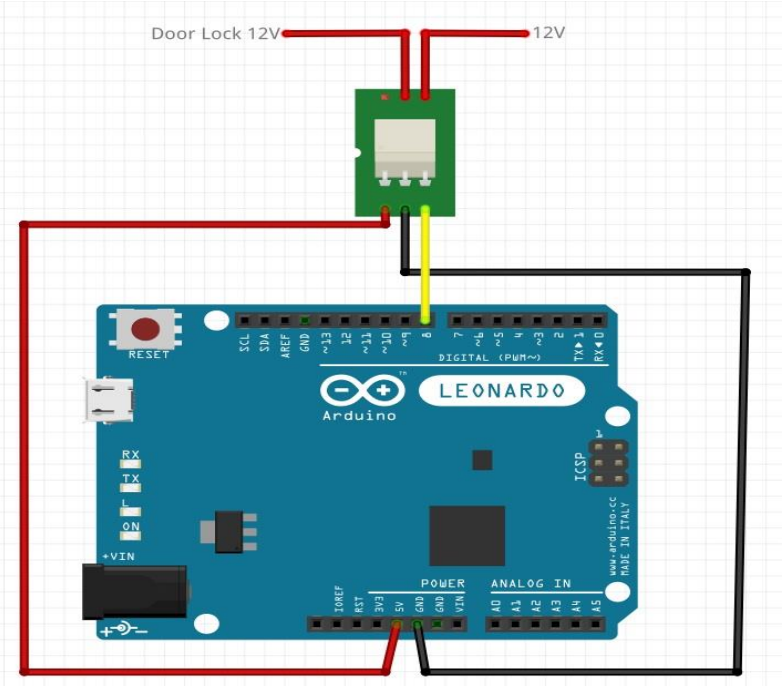
Şekil 3.21. 3B baskılı plastik kutuya monte edilen yüz tanıma donanımı

Şekil 3.22'de gösterildiği gibi röle devresi tarafından kapı kilidini kontrol etmek için Lattepanda gömülü Arduino kullanılmıştır.

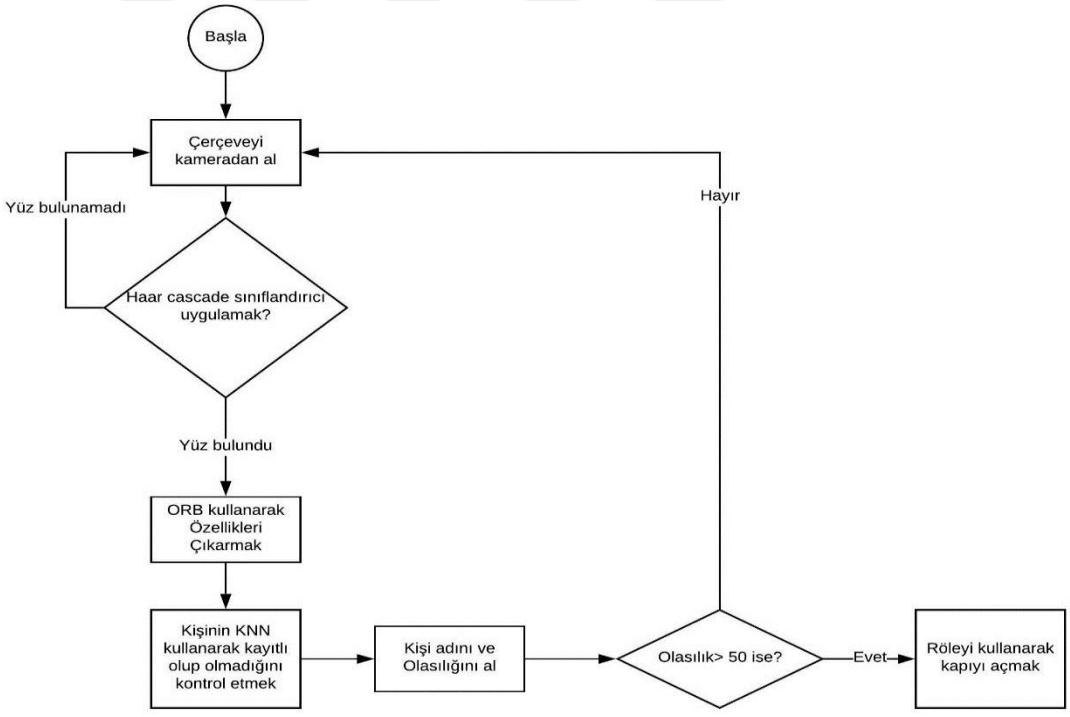


Şekil 3.22. Yüz tanıma sisteminin donanımı

Şekil 3.23'de Lattepanda gömülü Arduino Leonardo ile kapı kilidinin kontrolünü ve yüz tanındığında kapıyı açmak için kilit için bir güç vermek üzere bir düğme olarak kullanılan röleyi göstermektedir. Kamera görüntüyü alır. Haar cascade sınıflandırıcı uygulanarak yüz görüntüsü bulunur. ORB kullanılarak yüz görüntü özellikleri çıkarılır. KNN kullanılarak kişinin kayıtlı olup olmadığı kontrol edilir. Kişi adı ve olasılığı alınır. Olasılık 50'den büyük ise röle tetiklenir ve kapı açılır. Bu işlemler Şekil 3.24'te sistemin blok şemasında verilmektedir.



Şekil 3.23. Kapı kilidi kontrolünün şeması



Şekil 3.24. Son sistem blok şeması

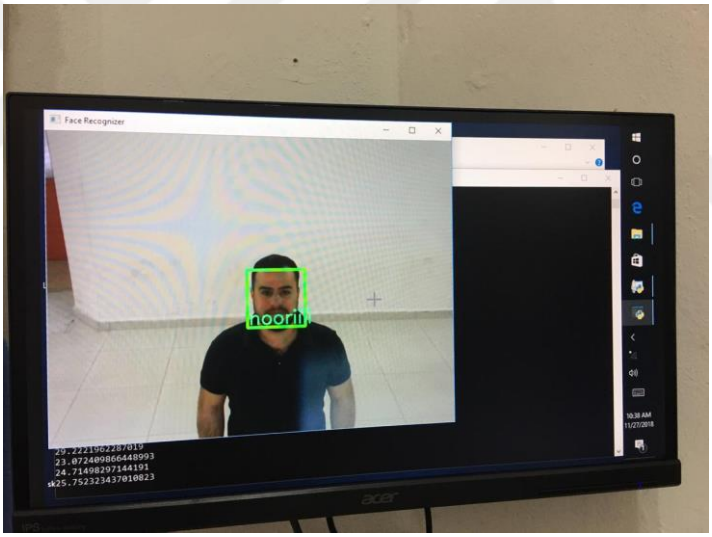
Şekil 3.25'de tasarlanan sistemin bir kapı geçiş kontrolüne uygulanışı görülmektedir. Sistem iki kısımdan oluşmaktadır. Bunlardan birincisi kameranın ve kontrol kartının bulunduğu kapı ünitesi, ikincisi ise sistemin kontrolünü yapacak olan kontrol operatörünün sistemini kontrol etmesini sağlayan kontrol ünitesidir.

Burada kapıya yaklaşan kişi kameraya bakacak, bunun sonucunda sistem onu tanıyacak ve elektrikli kilidi kullanarak kapıyı açacaktır.



Şekil 3.25. Koruyucu çanta ile kapı kilidine bağlı cihaz

Şekil 3.26'da kapının açılmasını deneyen kişiyi gösteren laboratuvar içindeki ekranı göstermektedir.



Şekil 3.26. Laboratuvardaki izleme ekranı

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Tez çalışmasında 3 farklı algoritmanın yüz tanıma performansı ve gömülü sistemler üzerinde daha az hesaplama maliyeti ve yüksek performans ile çalışabilme yeteneği incelenmiştir. Bu amaçla, 30 kişilik bir gruptan 3600 resim alınmak üzere bir veri seti oluşturulmuştur. Veri seti oluşturulurken ortamın ışık seviyesi, mesafe, yüz detayı ve kameraya bakış durumu gözönüne alınmıştır. 3600 resimden oluşan veri setinde her kişi için 120 resim bulunmaktadır. Bu resimlerden ORB, PCA ve LBP'den oluşan 3 farklı algoritma kullanılarak çıkarılan özellikler bir KNN kullanılarak sınıflandırılmıştır. Sınıflandırıcının eğitilmesi ve testi için resimler kendi içinde rastgele olarak 84 resim eğitim, 36 resim test için kulanılmak üzere ayrılmıştır. Bunun sonucunda toplamda 3600 resimden oluşan veri setinin 2520 resmi eğitim, 1080 resmi ise test için kullanılmıştır.

Bu dağılıma uygun olarak 3 farklı algoritmanın herbiri için eğitim ve test işlemleri gerçekleştirilmiştir. Testlerin sonucunda elde edilen veriler kaydedilerek analiz edilmiştir.

Eğitim ve test setleri için arka plana sahip görüntülerden yüzleri tespit etmek için Haar Cascade sınıflandırıcı kullanılmıştır. Çizelge 4.1'de ORB+KNN sınıflandırmasının gözlüklü görüntüler için test sonuçları gösterilmektedir. Çizelge 4.2'de ise ORB+KNN sınıflandırmasının gözlüksüz görüntüler için test sonuçları gösterilmektedir.

Çizelge 4.1. ORB+KNN sınıflandırmasının test sonuçları (gözlüklü görüntüler için)

Kişi	Az Işık																				Orta Işık																				Çok Işık																			
	50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1																																																											
2																				X																																								
3																																									X																	
4											X															X																																
5																																									X																	X
6																															X				X																					
7																																																									
8																X																																									
9																																																									
10																																																									X
11														X																																									X	X	X	
12																																																									X
13																				X																																				X		X		X
14																																									X															X			
15																																																X										X
16																																									X																	
17																																									X															X			
18			X				X																		X																			X		X												
19																																																									
20																							X																							X									X			
21														X			X																										X														X		
22																																									X																X
23								X																	X												X		X																	
24																X				X																																						
25															X		X																										X														X		
26																																									X	X														X		X
27																																																									
28																																										X																
29																																																									
30																																																									

Çizelge 4.2. ORB+KNN sınıflandırmasının test sonuçları (gözlüksüz görüntüler için)

ID No.	Az Işık																				Orta Işık																	Çok Işık																
	50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1																X				X					X										X									X										
2		X						X											X			X																																
3																																																					
4																																																					
5											X																																										
6														X				X																																			X
7														X					X											X																						
8																																																				X
9																																																				
10				X										X						X																																X	
11																X				X									X																								
12																		X	X																																		
13																																																				
14																				X																																
15																X				X					X																											
16																																																				
17										X																																									
18						X																																													
19	X																				X																															
20																																																			
21																																																			
22																																																			
23																																																			X
24														X				X																																		
25																X																																				
26																																																			
27																																																			
28																																																			
29																																																			
30								X																	X																										

Elde edilen sonuçlar Çizelge 4.1 ve 4.2'de doğru ve yanlış görüntü dağılımı olarak verilmektedir. Yanlış tespit edilen görüntüler "X", doğru bulunanlar ise boş kutu ile gösterilmektedir. Sonuçlar, yüz detayının sınıflandırma üzerindeki etkisini belirleyebilmek için gözlüklü ve gözlüksüz olarak iki ayrı çizelge şeklinde verilmektedir.

Çizelge 4.3'de LBP+KNN sınıflandırmasının gözlüklü görüntüler için test sonuçları gösterilmektedir. Çizelge 4.4'de ise LBP+KNN sınıflandırmasının gözlüksüz görüntüler için test sonuçları gösterilmektedir.

Çizelge 4.3. LBP+KNN sınıflandırmasının test sonuçları (gözlüklü görüntüler için)

ID No.	Az Işık																				Orta Işık																	Çok Işık																
	50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1				X																			X																		X	X														
2																							X																																	
3																																																								
4																																																								
5																																																								
6		X																					X																				X													
7																																																							
8																																																								
9		X																																																					
10																																																							
11																																																							
12																																																							
13			X	X														X					X																																X
14			X		X																			X																				X											
15											X			X				X				X																		X		X									X				X
16										X				X				X	X								X																					X					
17					X																				X																		X											
18				X	X																				X	X																				X									
19																																																							
20		X		X																		X			X																		X											
21		X			X																		X																		X		X									X		
22									X					X					X					X																		X				X								
23										X									X									X																									
24					X									X																												X												
25		X																	X	X																																		
26																																																							
27																																																							
28																																																							
29				X															X																																			
30																																																							

Çizelge 4.4. LBP+KNN sınıflandırmasının test sonuçları (gözlüksüz görüntüler için)

ID No.	Az Işık																				Orta Işık																	Çok Işık																
	50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1																							X																		X														
2		X		X																																				X														
3					X																																																	
4		X		X																																				X		X												
5																																																						
6		X																							X																				X									
7																																																						
8																							X																															
9		X											X								X	X																		X														
10											X												X	X					X											X		X											
11																							X																				X											
12	X	X							X		X												X		X					X				X							X			X	X				X				X
13														X																											X	X											
14				X				X															X				X																									
15																	X			X																																		
16				X																																				X													
17					X																																																
18																																																						
19																																																						
20																																										X											
21		X	X		X				X					X					X																						X		X		X		X				X	
22																							X	X																		X				X							
23																																																					
24																							X	X																		X											
25																																										X											
26																																																					
27																																																					
28																																																					
29																							X																		X											
30																																																				

Çizelge 4.5'de PCA+KNN sınıflandırmasının gözlüklü görüntüler için test sonuçları gösterilmektedir. Çizelge 4.6'de ise PCA+KNN sınıflandırmasının gözlüksüz görüntüler için test sonuçları gösterilmektedir.

Çizelge 4.5 PCA+KNN sınıflandırmasının test sonuçları (gözlüklü görüntüler için)

ID No.	Az Işık																				Orta Işık																	Çok Işık																
	50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1																																																					
2																																																					
3																																																					
4		X													X									X					X	X																	X					
5																																																		X			
6																						X					X																								X		
7																																																				
8									X								X																																			
9																																																					
10																																																					
11																																																					
12																																																					
13					X														X								X					X										X					X					X
14																																																				
15																																																					
16										X			X		X					X								X								X	X			X					X						
17																			X																				X							X						
18																		X																						X							X					
19																																																					
20				X					X															X								X							X	X											
21														X																																			X				
22									X				X	X			X											X																			X				X	
23																																																				
24																																																				
25		X							X		X						X		X			X					X					X															X				X	
26																																																			
27																																																				
28																																																				
29				X									X						X								X																					X			
30																																																				

Çizelge 4.6. PCA+KNN sınıflandırmasının test sonuçları (gözlüksüz görüntüler için)

ID No.	Az Işık																				Orta Işık																	Çok Işık																
	50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm					50cm					100cm					150cm					200cm				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	X					X					X					X				X				X		X				X				X				X				X		X					X				
2									X							X							X																X				X						X					X
3																																																				
4				X						X				X	X					X								X	X				X								X					X				X	
5																																																				
6					X																															X				X						X					X
7																						X																				X											
8																																																				
9																					X																X																
10				X			X		X			X		X					X			X		X								X																				
11																																																				
12		X											X										X		X			X		X			X		X				X				X				X			X	
13																																																		X			
14								X					X										X					X																				X				X	
15																																																				
16				X							X			X											X								X																				
17																																																					
18																																																					
19																																																					
20																																																					
21													X																																							
22															X																																					
23										X				X																																					
24													X																																							
25													X																																							
26																																																				
27																																																				
28																																																				
29																																																				
30																																																				

Farklı özellikler içeren veri seti üzerine 3 farklı algoritma uygulandığında elde edilen test sonuçları algoritmaların davranışları ile ilgili bilgiler vermektedir. Bunları daha iyi anlayabilmek için bu veriler aşağıdaki Çizelge 4.7 ve 4.8'de sayısal olarak ifade edilmektedir. Bu çizelgeleri incelediğimiz zaman algoritmaların performansının

veri seti içindeki 4 farklı özellikten nasıl etkilendiği daha belirgin olarak ortaya çıkmaktadır.

Çizelge 4.7 ve Çizelge 4.8'de yüz detayı olarak ifade edilen gözlüklü ve gözlüksüz görüntülerden oluşan veri setleri üzerinde algoritmaların test sonuçları verilmektedir.

Çizelge 4.7. Algoritmaların yüz detayına göre hatalı görüntü sayıları

Yüz Detayı	ORB	LBP	PCA
Gözlüklü	66	71	80
Gözlüksüz	82	81	105
Toplam	148	152	185

Çizelge 4.8. Algoritmaların yüz detayına göre başarı yüzdeleri (%)

Yüz Detayı	ORB	LBP	PCA
Gözlüklü	87,77	86,85	85,18
Gözlüksüz	84,81	85	80,55
Toplam Başarı (%)	86,29	85,92	82,87

Çizelgelerde verilen değerler ORB, LBP ve PCA algoritmalarının, veri setinin içerdiği yüz detayı, ışık seviyesi, mesafe ve bakış pozisyonu gibi durumlardan nasıl etkilendiğini belirgin bir şekilde göstermektedir.

Çizelge 4.9, Çizelge 4.10 ve Çizelge 4.11'de 3 farklı algoritmanın ortamın ışık seviyesindeki değişime bağlı olarak ortaya çıkan test sonuçları verilmektedir.

Çizelge 4.9. Algoritmaların ışık seviyesine göre hatalı görüntü sayıları

Yüz Detayı	Işık Seviyesi	ORB	LBP	PCA
Gözlüklü	Az Parlak	14	29	23
	Orta Parlak	20	22	25
	Çok Parlak	32	20	32
Gözlüksüz	Az Parlak	27	25	32
	Orta Parlak	17	23	34
	Çok Parlak	38	33	39

Çizelge 4.10 Algoritmaların ışık seviyesine göre başarı yüzdeleri (%)

Yüz Detayı	Işık Seviyesi	ORB	LBP	PCA
Gözlüklü	Az Parlak	92,22	83,88	87,22
	Orta Parlak	88,88	87,77	86,11
	Çok Parlak	82,22	88,88	82,22
Gözlüksüz	Az Parlak	85	86,11	82,22
	Orta Parlak	90,55	87,22	81,11
	Çok Parlak	78,88	81,66	78,33

Çizelge 4.11 Algoritmaların ışık seviyesine göre toplam başarı yüzdeleri (%)

Işık Seviyesi	ORB	LBP	PCA
Az Parlak	88,61	85	84,72
Orta Parlak	89,72	87,5	83,61
Çok Parlak	80,55	85,27	80,27

Çizelge 4.12, Çizelge 4.13 ve Çizelge 4.14'de 3 farklı algoritmanın mesafeye bağlı olarak ortaya koydukları test sonuçları görülmektedir.

Çizelge 4.12. Algoritmaların mesafeye göre hatalı görüntü sayıları

Yüz Detayı	Mesafe	ORB	LBP	PCA
Gözlüklü	50cm	5	45	17
	100cm	10	6	21
	150cm	9	10	19
	200cm	42	10	23
Gözlüksüz	50cm	12	52	27
	100cm	17	9	24
	150cm	9	14	29
	200cm	44	6	25

Çizelge 4.13. Algoritmaların mesafeye göre başarı yüzdeleri (%)

Yüz Detayı	Mesafe	ORB	LBP	PCA
Gözlüklü	50cm	96,29	66,66	87,4
	100cm	92,59	95,55	84,44
	150cm	93,33	92,59	85,92
	200cm	68,88	92,59	82,96
Gözlüksüz	50cm	91,11	61,48	80
	100cm	87,40	93,33	82,22
	150cm	93,33	89,62	78,51
	200cm	67,4	95,55	81,48

Çizelge 4.14. Algoritmaların mesafeye göre toplam başarı yüzdeleri (%)

Mesafe	ORB	LBP	PCA
50cm	93,7	64,07	83,7
100cm	90	94,44	83,33
150cm	93,33	91,11	82,22
200cm	68,14	94,07	82,22

Çizelge 4.15, Çizelge 4.16 ve Çizelge 4.17'de 3 farklı algoritmanın 5 farklı bakış pozisyonuna göre ortaya koydukları test sonuçları görülmektedir.

Çizelge 4.15. Algoritmaların bakış pozisyonuna göre hatalı görüntü sayıları

Yüz Detayı	Pozisyon	ORB	LBP	PCA
	Pozisyon	10	5	26
	Ön	8	11	3
Gözlüklü	Yukarı	15	10	14
	Aşağı	11	18	13
	Sol	22	27	23
	Sağ	17	6	14
	Ön	10	17	15
Gözlüksüz	Yukarı	20	7	4
	Aşağı	30	26	40
	Sol	8	26	32

Çizelge 4.16. Algoritmaların bakış pozisyonuna göre başarı yüzdeleri (%)

Yüz Detayı	Pozisyon	ORB	LBP	PCA
	Ön	90,74	95,37	75,92
	Yukarı	92,59	89,81	97,22
Gözlüklü	Aşağı	86,11	90,74	87,03
	Sol	89,81	83,33	87,96
	Sağ	79,62	75	78,7
	Ön	84,25	94,44	87,03
	Yukarı	90,74	84,25	86,11
Gözlüksüz	Aşağı	81,48	93,51	96,29
	Sol	72,22	75,92	62,96
	Sağ	92,59	75,92	70,37

Çizelge 4.17. Algoritmaların bakış pozisyonuna göre toplam başarı yüzdeleri (%)

Pozisyon	ORB	LBP	PCA
Ön	87,5	94,90	81,48
Yukarı	91,66	87,03	91,66
Aşağı	83,79	92,12	91,66
Sol	81,01	79,62	75,46
Sağ	86,11	75,46	74,53

5. SONUÇLAR VE ÖNERİLER

5.1 Sonuçlar

Bu tez çalışmasında yüz tanıma alanında kullanılan Yönelimli FAST ve döndürülmüş BRIEF ORB, yerel ikili örüntü LBP ve temel bileşenler analizi PCA olmak üzere üç ayrı öznitelik çıkarımı algoritması incelenmiştir. ORB algoritması, FAST (hızlandırılmış segment testi özellikleri) köşe algılayıcısını kullanarak ilgi noktalarına erişen piksellere bağlı olan ve BRIEF (İkili Gürbüz Bağımsız Temel Özellikler) kullanarak öznitelik üreten bir yerel öznitelik tipidir. Bu öznitelikler rotasyon-değişimsizdir. Bunun yanı sıra LBP, görüntüyü ikili temsillere (siyah ve beyaz) dönüştüren ve doku histogramını kullanarak yüz özniteliklerini temsil eden bir doku özniteliği tipidir. PCA algoritması yüz tanımada kullanıldığında özyüzler olarak adlandırılmaktadır. İlintisiz görüntü değerlerini azaltarak boyutsallığın azalmasını kullanmakta ve öznitelik olarak kullanılan özyüzleri oluşturmaktadır.

Bu çalışmada kullanılan yüz tanıma sistemleri, Python programlama dili, OpenCV (bilgisayar görüntü kütüphanesi), skimage (görüntü işleme kütüphanesi) ve Scikit-learn (makine öğrenme kütüphanesi) kullanılarak gerçekleştirilmiştir. Bu sistemleri değerlendirmek için kullanılan veri kümesi 30 denekten toplanmıştır. Her bir denekten toplam 120 adet görüntü alınmıştır. Görüntüler 5 farklı yüz pozu, üç farklı ışık seviyesi, iki farklı yüz detayı ve dört farklı kamera uzaklığı dikkate alınarak elde edilmiştir. Toplam veri seti 3600 adet görüntüden oluşmaktadır. Çekilen görüntülerin işlenmesinde ve görüntülerden arka planın çıkarılmasında Haar Cascade sınıflandırıcısı kullanılmıştır. Çapraz geçerlilik, eğitim setini ve test setini bölmek için kullanılmıştır. Eğitim seti, her denek için 120 görüntü içeren veri kümesinden 84 rastgele görüntü (%70) içermekte ve test seti, 36 rastgele görüntü (%30) içermektedir. ORB tabanlı yüz tanıma sistemi, her yüz görüntüsünden özellik tanımlayıcılarını çıkarmak için ORB'yi kullanır ve ardından K-En Yakın Komşu KNN sınıflandırıcısını eğitmek için sabit uzunluklu bir özellik vektörü oluşturmak üzere Linde-Buzo-Gray LBG vektör niceleme algoritması uygulanmıştır. Tüm görüntülerin aynı kesişme noktalarına sahip olmamasından ve bunun makine öğrenme modelleri için farklı bir özellik vektörleri uzunluğu anlamına gelmesinden dolayı LBG algoritması kullanılmıştır. ORB tabanlı sistemde sınıflandırıcı olarak KNN kullanılmıştır. Yapılan bu çalışmada eğitim seti üzerinde eğitim gerçekleştirilmiş ve test seti üzerinde rastgele doğrulanmış KNN'li sistemin ortalama doğruluğu diğer yöntemlerin doğruluğundan daha yüksek tespit edilmiştir (% 86,29). LBP tabanlı yüz tanıma sistemi, her yüz görüntüsünde dokuya

erişmek için yerel ikili örüntü algoritmasını kullanmakta ve sonrasında her dokuyu KNN sınıflandırıcısını eğitmek için kullanılan histograma dönüştürmektedir. Bu yöntemle, test setinde %85.9 doğruluk elde edilmiştir. PCA tabanlı yüz tanıma sistemi, eğitim ve test veri kümesinden özyüzlere erişmek için kullanılmaktadır. Bu özyüzler, ortalama %82.8 doğruluk elde edilen KNN sınıflayıcıyı eğitmek için kullanılmıştır. Yüz tanıma sisteminin pratik kullanımı laboratuvar girişinde kullanılan bir kapı geçişinde gerçekleştirilmiştir. Öğrenci Lattepanda Modül Bilgisayarı, Logitech c310 web kamerası kullanılarak yapılan bu gerçek zamanlı yüz tanıma sistemi kontrollü ve güvenli bir şekilde laboratuvara öğrencilerin girişinde aktif olarak kullanılarak test edilmiştir. Donanım, 3B baskılı plastik bir kutuya yerleştirilmiş ve tüm sistemi kontrol etmek için laboratuarın içinde klavye ve fare ile 21 inç bir ekrana bağlanmıştır. Bu sistemde bir elektrikli kapı kilidi kullanılmış ve kişi doğrulandığında kapıyı açmak için Lattepanda kullanılarak kontrol edilen bir röle aracılığıyla kapı açılmıştır.

5.2 Öneriler

Bu tez çalışmasında, ORB, PCA ve LBP'dan oluşan 3 farklı algoritmanın yüz tanıma performansı incelenmiştir. Bu üç algoritma, ışık, mesafe, yüz detayı ve poz gibi farklı koşullara sahip 30 denekten toplanan veri kümesi üzerinde test edilmiştir. Bu yaklaşımlar gerçek zamanlı olarak kullanılmış ve öğrenciler için laboratuvarda uygulanan yüz tanıma tabanlı bir kapı kilidinde Lattepandadan oluşan bir gömülü sistem kullanılarak gerçekleştirilmiştir.

Yapılan testlerin sonucunda, ORB algoritmasının yüksek ışık seviyesinde ve uzak mesafede daha başarısız olduğu görülmektedir. Bunların yanısıra, yüz detayının artması algoritmanın başarı oranını artırmaktadır.

PCA yöntemi ise mesafe ve ışık değişiminden etkilenmemektedir. Fakat yüksek hesaplama gücü gereksinimleri sebebiyle sistemin uygulamasının dışında bırakılmıştır.

LBP yaklaşımı, hesaplama gücünün doğruluğu açısından iyi bir sonuç vermiştir ve gerçek zamanlı olarak herhangi bir gecikme olmaksızın düzgün şekilde çalışmaktadır. Yalnız yakın mesafedeki başarı oranı çok düşmektedir ve ışık seviyesine göre orta parlak ışık seviyesinde nispeten daha başarılı olduğu saptanmıştır.

Kameraya bakıştaki poz farlılıkları açısından bakdığımızda ise genelde yüzün bir yarısını kapsayan yanal görüntülerde algoritmaların başarı oranının düştüğü görülmektedir.

Sistemin kapalı bir ortamda az veya orta ışık seviyelerinde ve yakın mesafe aralığında çalışacağı düşünüldüğünde, gömülü sistemin üzerinde ORB tabanlı yüz tanıma sisteminin daha başarılı olacağı öngörülmüştür. Bu nedenle, kapı geçiş kontrolünü gerçekleştirecek olan pratik sistem ORB tabanlı yüz tanıma algoritması kullanılarak uygulanmıştır.

KAYNAKLAR

Ahonen, T., Hadid, A. ve Pietikainen, M., 2006, Face description with local binary patterns: Application to face recognition, *IEEE Transactions on Pattern Analysis & Machine Intelligence* (12), 2037-2041.

Altman, N. S., 1992, An introduction to kernel and nearest-neighbor nonparametric regression, *The American Statistician*, 46 (3), 175-185.

Ballantyne, M., Boyer, R. S. ve Hines, L., 1996, Woody bledsoe: His life and legacy, *AI magazine*, 17 (1), 7.

Barnett, T. ve Preisendorfer, R., 1987, Origins and levels of monthly and seasonal forecast skill for United States surface air temperatures determined by canonical correlation analysis, *Monthly Weather Review*, 115 (9), 1825-1850.

Baron, R. J., 1981, Mechanisms of human facial recognition, *International Journal of Man-Machine Studies*, 15 (2), 137-178.

Beymer, D. ve Poggio, T., 1995, Face recognition from one model view, *Proc. Fifth Int'l Conf. Computer Vision*.

Bichsel, M., 1991, Strategies of robust object recognition for the automatic identification of human faces, *ETH Zurich*.

Blanz, V. ve Vetter, T., 2003, Face recognition based on fitting a 3D morphable model, *IEEE Transactions on Pattern Analysis and machine intelligence*, 25 (9), 1063-1074.

Brunelli, R. ve 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. ve Yianilos, P. N., 1996, Feature-based face recognition using mixture-distance, *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*, 209-216.

Déniz, O., Castrillon, M. ve Hernández, M., 2003, Face recognition using independent component analysis and support vector machines, *Pattern recognition letters*, 24 (13), 2153-2157.

Galton, F., 1878, Composite portraits, *Nature*, 18, 97-100.

Gao, Y. ve Leung, M. K., 2002, Face recognition using line edge map, *IEEE Transactions on Pattern Analysis & Machine Intelligence* (6), 764-779.

Georghiades, A. S., Belhumeur, P. N. ve Kriegman, D. J., 2001, From few to many: Illumination cone models for face recognition under variable lighting and pose, *IEEE Transactions on Pattern Analysis and machine intelligence*, 23 (6), 643-660.

Goldstein, A. J., Harmon, L. D. ve Lesk, A. B., 1971, Identification of human faces, *Proceedings of the IEEE*, 59 (5), 748-760.

Grudin, M. A., 1997, A compact multi-level model for the recognition of facial images, *Liverpool John Moores University*.

Guo, G.-D., Zhang, H.-J. ve Li, S. Z., 2001, Pairwise face recognition, *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 282-287.

Guo, G., Li, S. Z. ve Chan, K., 2000, Face recognition by support vector machines, *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 196-201.

Hadid, A. ve Pietikäinen, M., 2009, Combining appearance and motion for face and gender recognition from videos, *Pattern recognition*, 42 (11), 2818-2827.

Heisele, B., Ho, P. ve Poggio, T., 2001, Face recognition with support vector machines: Global versus component-based approach, *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 688-694.

Hong, L. ve Jain, A., 1998, Integrating faces and fingerprints for personal identification, *Asian Conference on Computer Vision*, 16-23.

Hotelling, H., 1933, Analysis of a complex of statistical variables into principal components, *Journal of educational psychology*, 24 (6), 417.

Huang, J., Shao, X. ve Wechsler, H., 1998, Face pose discrimination using support vector machines (SVM), *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, 154-156.

Huang, J., Blanz, V. ve Heisele, B., 2002, Face recognition using component-based SVM classification and morphable models, In: Pattern Recognition with Support Vector Machines, Eds: Springer, p. 334-341.

Jayant, N. K. ve Borra, S., 2016, Attendance management system using hybrid face recognition techniques, *Advances in signal processing (CASP), Conference on*, 412-417.

Jolliffe, I., 2011, Principal component analysis, In: International encyclopedia of statistical science, Eds: Springer, p. 1094-1096.

Jonsson, K., Matas, J., Kittler, J. ve Li, Y., 2000, Learning support vectors for face verification and recognition, *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 208-213.

Jonsson, K., Kittler, J., Li, Y. ve Matas, J., 2002, Support vector machines for face authentication, *Image and Vision Computing*, 20 (5-6), 369-375.

Kanade, T., 1974, Picture processing system by computer complex and recognition of human faces.

Kaya, Y. ve Kobayashi, K., 1972, A basic study on human face recognition, In: Frontiers of pattern recognition, Eds: Elsevier, p. 265-289.

Kim, K. I., Kim, J. H. ve Jung, K., 2002, Face recognition using support vector machines with local correlation kernels, *International Journal of Pattern Recognition and Artificial Intelligence*, 16 (01), 97-111.

Kirby, M. ve 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.

Kohonen, T., 1989, Various Aspects of Memory, In: Self-Organization and Associative Memory, Eds: Springer, p. 1-29.

Kung, S.-Y. ve Taur, J.-S., 1995, Decision-based neural networks with signal/image classification applications, *IEEE Transactions on Neural Networks*, 6 (1), 170-181.

Lades, M., Vorbruggen, J. C., Buhmann, J., Lange, J., Von Der Malsburg, C., Wurtz, R. P. ve Konen, W., 1993, Distortion invariant object recognition in the dynamic link architecture, *IEEE Transactions on computers* (3), 300-311.

Lawrence, S., Giles, C. L., Tsoi, A. C. ve Back, A. D., 1997, Face recognition: A convolutional neural-network approach, *IEEE Transactions on Neural Networks*, 8 (1), 98-113.

Lepetit, V. ve Fua, P., 2006, Keypoint recognition using randomized trees, *IEEE Transactions on Pattern Analysis and machine intelligence*, 28 (9), 1465-1479.

Leung, M. K. ve Yang, Y.-H., 1990, Dynamic two-strip algorithm in curve fitting, *Pattern recognition*, 23 (1-2), 69-79.

Li, Y., Gong, S. ve Liddell, H., 2000, Support vector regression and classification based multi-view face detection and recognition, *fg*, 300.

Liao, S., Zhu, X., Lei, Z., Zhang, L. ve Li, S. Z., 2007, Learning multi-scale block local binary patterns for face recognition, *International Conference on Biometrics*, 828-837.

Lin, C.-J., 2001, On the convergence of the decomposition method for support vector machines, *IEEE Transactions on Neural Networks*, 12 (6), 1288-1298.

Lin, S.-H., Kung, S.-Y. ve Lin, L.-J., 1997, Face recognition/detection by probabilistic decision-based neural network, *IEEE Transactions on Neural Networks*, 8 (1), 114-132.

Linde, Y., Buzo, A. ve Gray, R., 1980, An algorithm for vector quantizer design, *IEEE Transactions on communications*, 28 (1), 84-95.

Manjunath, B., Chellappa, R. ve von der Malsburg, C., 1992, A feature based approach to face recognition, *Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 373-378.

Ojala, T., Pietikäinen, M. ve Harwood, D., 1996, A comparative study of texture measures with classification based on featured distributions, *Pattern recognition*, 29 (1), 51-59.

Ojala, T. ve Pietikäinen, M., 1999, Unsupervised texture segmentation using feature distributions, *Pattern recognition*, 32 (3), 477-486.

Ojala, T., Pietikainen, M. ve Maenpaa, T., 2002, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and machine intelligence*, 24 (7), 971-987.

Ozuysal, M., Calonder, M., Lepetit, V. ve Fua, P., 2010, Fast keypoint recognition using random ferns, *IEEE Transactions on Pattern Analysis and machine intelligence*, 32 (3), 448-461.

Pang, S., Kim, D. ve Bang, S. Y., 2003, Membership authentication in the dynamic group by face classification using SVM ensemble, *Pattern recognition letters*, 24 (1-3), 215-225.

Papageorgiou, C. P., Oren, M. ve Poggio, T., 1998, A general framework for object detection, *Computer vision, 1998. sixth international conference on*, 555-562.

Pearson, K., 1901, LIII. On lines and planes of closest fit to systems of points in space, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2 (11), 559-572.

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

Phillips, P. J., 1999, Support vector machines applied to face recognition, *Advances in Neural Information Processing Systems*, 803-809.

Rosten, E. ve Drummond, T., 2006, Machine learning for high-speed corner detection, *European conference on computer vision*, 430-443.

Rublee, E., Rabaud, V., Konolige, K. ve Bradski, G., 2011, ORB: An efficient alternative to SIFT or SURF, *Computer Vision (ICCV), 2011 IEEE international conference on*, 2564-2571.

Samaria, F. ve Fallside, F., 1993, Face identification and feature extraction using hidden markov models, Citeseer, p.

Samaria, F. S. ve Harter, A. C., 1994, Parameterisation of a stochastic model for human face identification, *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, 138-142.

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

Stonham, T., 1986, Practical face recognition and verification with WISARD, In: Aspects of face processing, Eds: Springer, p. 426-441.

Sung, K.-K. ve Poggio, T., 1995, Learning human face detection in cluttered scenes, *International Conference on Computer Analysis of Images and Patterns*, 432-439.

Takacs, B., 1998, Comparing face images using the modified Hausdorff distance, *Pattern recognition*, 31 (12), 1873-1881.

Tamura, S., Kawai, H. ve Mitsumoto, H., 1996, Male/female identification from 8× 6 very low resolution face images by neural network, *Pattern recognition*, 29 (2), 331-335.

Tong, D. H. S. M. K., A spectral algorithm for learning hidden Markov models.

Turk, M. A. ve Pentland, A. P., 1991, Face recognition using eigenfaces, *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, 586-591.

Vapnik, V., 2013, The nature of statistical learning theory, Springer science & business media, p.

Verlinde, P. ve Mayoraz, E., 1998, Decision fusion using a multi-linear classi er.

Vetter, T. ve 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.

Vetter, T. ve Blanz, V., 1998, Estimating coloured 3D face models from single images: An example based approach, *European Conference on Computer Vision*, 499-513.

Viola, P. ve Jones, M., 2001, Rapid object detection using a boosted cascade of simple features, *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, I-I.

Weng, J. J., Ahuja, N. ve Huang, T. S., 1993, Learning recognition and segmentation of 3-d objects from 2-d images, *Computer Vision, 1993. Proceedings., Fourth International Conference on*, 121-128.

Wiskott, L. ve Von Der Malsburg, C., 1996, Face recognition by dynamic link matching, INI, p.

Zhao, G. ve Pietikainen, M., 2007, Dynamic texture recognition using local binary patterns with an application to facial expressions, *IEEE Transactions on Pattern Analysis and machine intelligence*, 29 (6), 915-928.

Zhao, G., Barnard, M. ve Pietikainen, M., 2009, Lipreading with local spatiotemporal descriptors, *IEEE Transactions on Multimedia*, 11 (7), 1254-1265.

Zhao, L. ve Yang, Y.-H., 1999, Theoretical analysis of illumination in PCA-based vision systems, *Pattern recognition*, 32 (4), 547-564.

Zhao, W. ve Chellappa, R., 1999, Robust image based 3 D face recognition, *research directed by Dept. of Electrical and Computer Engineering.University …*.

Zhao, W. ve Chellappa, R., 2000, SFS based view synthesis for robust face recognition, *fg*, 285.

Zhao, W., Chellappa, R., Phillips, P. J. ve Rosenfeld, A., 2003, Face recognition: A literature survey, *ACM computing surveys (CSUR)*, 35 (4), 399-458.

EKLER

EK-1 ORB Face recognition dataset testing:

```
import cv2
import numpy as np
import lbg
import os

import pickle
from sklearn.model_selection import train_test_split
orb=cv2.ORB_create(nfeatures=500,fastThreshold=7,scaleFactor=1.2,edgeThres
hold=25,patchSize=25)
path='./done/'
x=[]
y=[]
z=[]

for directory in os.listdir(path):
    for file in os.listdir(path+directory):
        print(directory)
        im=cv2.imread(path+directory+"/"+file,0)
        kp,des=orb.detectAndCompute(im,None)
        if des is not None:

            x.append(des)
            y.append(directory)
            z.append(file)


ll=[]
for i in range(len(x)):
    print(i)
    cb,_,_=lbg.generate_codebook(x[i],1)
    ll.append(cb)

shapes=np.array(ll).shape
feat=np.reshape(ll,(shapes[0],shapes[1]*shapes[2]))

X_train, X_test, y_train, y_test,z_train,z_test = train_test_split(feat, y,z,
test_size=0.3, random_state=42)
with open('./data_orb.pkl','wb') as fb:

    pickle.dump((X_train,y_train,X_test,y_test,z_train,z_test),fb)

with open('./data_orb.pkl','rb') as fb:

    X_train,y_train,X_test,y_test,z_train,z_test=pickle.load(fb)
from sklearn.preprocessing import StandardScaler
```

```
        std=StandardScaler().fit(X_train)
        X_train=std.transform(X_train)
        X_test=std.transform(X_test)

        from sklearn.neighbors import KNeighborsClassifier
        clf = KNeighborsClassifier()

        clf.fit(X_train,y_train)
        T=[]
        F=[]
        from sklearn.metrics import accuracy_score
        z=clf.predict(X_test)
        print(accuracy_score(y_test,z))
        for i in range(len(X_test)):
           print(i)
           pred=clf.predict(np.array(X_test[i]).reshape(1,-1))
           if y_test[i]==pred:
              T.append((z_test[i],y_test[i]))
           else:
              F.append((z_test[i],y_test[i]))
        import pandas as pd
pd.DataFrame(F).to_excel('orb_results.xlsx', header=False, index=False)
```

EK-2 LBP face recognition dataset test:

```
import cv2
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import os
import numpy as np
from skimage.feature        import local_binary_pattern
import pickle

def get_lbp(gray):
   row       = gray.shape[0]
   col       = gray.shape[1]
   neighbors = 24
   radius    = 3
   weight    = [[1,1,1,1,1,1,1],
              [1,1,1,1,1,1,1],
              [1,1,1,1,1,1,1],
              [0,1,1,1,1,1,0],
              [0,1,1,1,1,1,0],
              [0,1,1,1,1,1,0],
              [0,1,1,1,1,1,0]]

   #extract the LBP feature of the whole image
   lbp = local_binary_pattern(gray,
                        neighbors,
                        radius,
                        method="uniform")
   local_hist=[]
   for r in range(7):
      for c in range(7):

         #the range of the block
         r_start = r * int(row / 7)
         c_start = c * int(col / 7)

         if((r + 1) * int(row / 7) <= row):
            r_end = (r + 1) * int(row / 7)
         else:
            r_end = row
         if((c + 1) * int(col / 7) <= col):
            c_end = (c + 1) * int(col / 7)
         else:
            c_end = col
         if not weight[r][c] == 0:
            #get the regional histogram
            (hist_temp, _) = np.histogram(lbp[r_start:r_end, c_start:c_end].ravel(),
                                 bins=np.arange(0, neighbors + 3),
                                 range=(0, neighbors + 2))
```

```
                #normalize the histogram
                hist_temp = hist_temp.astype("float")
                hist_temp /= (hist_temp.sum())

            local_hist = local_hist + list(hist_temp * weight[r][c])
        return local_hist

    path="./done/"
    x=[]
    y=[]
    z=[]
    for directory in os.listdir(path):
        for file in os.listdir(path+directory):
            print(directory)

            img=cv2.imread(path+directory+"/"+file,0)
            lbpfeat=get_lbp(img)
            x.append(lbpfeat)
            y.append(directory)
            z.append(file)


    X_train, X_test, y_train, y_test,z_train,z_test = train_test_split(x, y,z, test_size=0.3, random_state=42)
    with open('./data_lbp.pkl','wb') as fb:

        pickle.dump((X_train,y_train,X_test,y_test,z_train,z_test),fb)

    with open('./data_lbp.pkl','rb') as fb:

        X_train,y_train,X_test,y_test,z_train,z_test=pickle.load(fb)
    from sklearn.neighbors import KNeighborsClassifier
    clf = KNeighborsClassifier()
    clf.fit(X_train,y_train)
    pred=clf.predict(X_test)
    print(accuracy_score(y_test,pred))
    T=[]
    F=[]
    for i in range(len(X_test)):
        print(i)
        pred=clf.predict(np.array(X_test[i]).reshape(1,-1))
        if y_test[i]==pred:
            T.append((z_test[i],y_test[i]))
        else:
            F.append((z_test[i],y_test[i]))
    import pandas as pd
pd.DataFrame(F).to_excel('lbp_results.xlsx', header=False, index=False)
```

EK-3 PCA face recognition dataset test:

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import cv2
import pickle
from sklearn.decomposition import PCA
import numpy as np
import os
n_components = 100
path='./done/'
x=[]
y=[]
z=[]

for directory in os.listdir(path):
   for file in os.listdir(path+directory):
      print(directory+"/"+file)
      img=cv2.imread(path+directory+"/"+file,0)
      x.append(np.array(img).ravel())
      y.append(directory)
      z.append(file)
print(np.array(x).shape)

X_train, X_test, y_train, y_test,z_train,z_test = train_test_split(x, y,z, test_size=0.3, random_state=42)


pca = PCA(n_components=n_components, svd_solver='randomized',whiten=True).fit(X_train)
eigenfaces = pca.components_.reshape((n_components, 300, 300))
X_train_pca = pca.transform(X_train)
X_test_pca = pca.transform(X_test)

with open('./data_pca.pkl','wb') as fb:


pickle.dump((X_train_pca,y_train,X_test_pca,y_test,z_train,z_test,eigenfaces),fb)

with open('./data_pca.pkl','rb') as fb:

X_train_pca,y_train,X_test_pca,y_test,z_train,z_test,eigenfaces=pickle.load(fb)

from sklearn.neighbors import KNeighborsClassifier
clf = KNeighborsClassifier()
clf.fit(X_train_pca,y_train)
pred=clf.predict(X_test_pca)
print(accuracy_score(y_test,pred))
T=[]
```

```
        F=[]
        for i in range(len(X_test_pca)):
            print(i)
            pred=clf.predict(np.array(X_test_pca[i]).reshape(1,-1))
            if y_test[i]==pred:
                T.append((z_test[i],y_test[i]))
            else:
                F.append((z_test[i],y_test[i]))
        import pandas as pd
pd.DataFrame(F).to_excel('pca_results.xlsx', header=False, index=False)
```

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı	**:**	Abdullah Mohammed NOORI
Uyruğu	**:**	IRAQ
Doğum Yeri ve Tarihi	**:**	KERKUK 19/10/1988
Telefon	**:**	05315142634
Faks	**:**	
e-mail	**:**	abdullah.noori.1988@gmail.com

EĞİTİM

Derece		**Adı, İlçe, İl**	**Bitirme Yılı**
Lise	:	**Kerkük Erkek lisesi , Kerkük**	**2007**
Üniversite	:	**AL-Rafidain University , Baghdad**	**2014**
Yüksek Lisans	:		
Doktora	:		

İŞ DENEYİMLERİ

Yıl	**Kurum**	**Görevi**

YAYINLAR

Selek, M. ve NOURİ, A. 2018, Face Recognition System Based on ORB (Oriented Fast and Rotated Brief) Algorithm, II International Symposium Humanities and Social Sciences, COMRAT/ Moldova. 88-95.