

**ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

DERİN ÖĞRENME İLE YÜZ TANIMA

Fatma Zehra ÜNAL

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**ANKARA
2017**

Her hakkı saklıdır


TEZ ONAYI


Fatıma Zehra ÜNAL tarafından hazırlanan “**Derin Öğrenme ile Yüz Tanıma**” adlı tez çalışması 20/07/2017 tarihinde aşağıdaki jüri tarafından oy birliği ile Ankara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.


Danışman : Yrd. Doç. Dr. Mehmet Serdar GÜZEL
Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı

Jüri Üyeleri :


Başkan: Doç. Dr. Süleyman TOSUN
Hacettepe Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı


Üye: Yrd. Doç. Dr. Gazi Erkan BOSTANCI
Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı


Üye: Yrd. Doç. Dr. Mehmet Serdar GÜZEL
Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı

Yukarıdaki sonucu onaylarım.

Prof. Dr. Atila YETİŞEMİYEN
Enstitü Müdürü

ETİK

Ankara Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez içindeki bütün bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atıf yaparak belirttiğimi beyan ederim.

20/07/2017


Fatima Zehra ÜNAL

ÖZET

Yüksek Lisans Tezi

DERİN ÖĞRENME İLE YÜZ TANIMA

Fatma Zehra ÜNAL

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Yrd. Doç. Dr. Mehmet Serdar GÜZEL

Güvenlik uygulamaları ve ticari uygulamalar olmak üzere çok sayıda alanı ve disiplini kapsayan yüz tanıma, görüntü analizi alanının amacına en uygun önemli bir araştırma problemidir. İnsanların yüzleri tanıma kabiliyetine eşit, otomatik bir sistem geliştirmek ciddi bir problemdir. Yüz tanıma uygulamalarının farklı disiplinlere konu olması, bu alanda yapılan ve yapılacak çalışmalara olan ilgiyi arttırmaktadır. Uzun yıllar yapay zeka alanında geliştirilen uygulamalar ile sonuç alınamamış sorunların çözümünde Derin Öğrenme teknolojisi ile büyük ilerleme kaydedilmiştir. Yüksek boyutlu verilerin karmaşık yapılarını keşfetme konusunda çok iyi olduğu ortaya çıkan Derin öğrenme, bilim, iş dünyası ve devletin birçok alanında uygulanmaya başlamıştır. Derin Öğrenme teknikleri kullanarak bilgisayarlarla görme alanında birçok sınıflandırma problemi performansı arttırılmıştır. Derin öğrenme, özellikle Konvolüsyonel Sinir Ağı (KSA), son zamanlarda yüz tanıma uygulamalarında umut verici sonuçlar elde etmiştir.

Bu tez çalışması kapsamında, geleneksel makine öğrenme tekniği ve derin öğrenme tekniği kullanılarak bir yüz tanıma algoritması eğitilmiş ve iki ayrı yöntemin başarı oranlarını karşılaştırılmıştır. Algoritma, yüz tespiti, ön işlemler, öznitelik çıkarımı ve sınıflandırma olmak üzere 4 ana kısımdan oluşmaktadır. Öznitelik çıkarımı kısmında Histogram of Gradients (HOG) ve Derin Öğrenme olmak üzere iki ayrı teknik kullanılmıştır. Elde edilen özellikler çok sınıflı Destek Vektör Makineleri (DVM) metodu kullanılarak sınıflandırılmıştır.

Temmuz 2017, 79 sayfa

Anahtar Kelimeler: Yüz Tanıma, Derin Öğrenme, Konvolüsyonel Sinir Ağı, Yüz Tanıma Algoritması, Yapay Sinir Ağları, MATLAB

ABSTRACT

Master Thesis

FACE RECOGNITION WITH DEEP LEARNING

Fatma Zehra ÜNAL

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Asst. Prof. Dr. Mehmet Serdar GÜZEL

Face recognition, which encompasses a large number of fields and disciplines, including safety applications and commercial applications, is an important research problem most suited to the purpose of image analysis. Developing an automated system that is equal to the ability of people to recognize faces is a serious problem. The fact that facial recognition applications are subject to different disciplines increases the relevance of the work done in this area. Deep Learning technology has made great progress in solving problems that have not been achieved with applications developed for a long time in artificial intelligence field. It has begun to be applied in many areas of deep learning, science, business, and government, which have proven to be very good at discovering complex structures of high-dimensional data. Deep Learning techniques have improved the performance of many classification problems in the field of vision with computers, especially using Convolutional Neural Networks. Deep learning, especially the Convolutional Neural Network (CNN), has recently achieved promising results in face recognition applications.

In this thesis, a face recognition algorithm is trained by using traditional machine learning technique and deep learning technique and the success rates of the two methods are compared. Algorithm consists of 4 main parts; face detection, preprocessing, feature extraction and classification. Two separate techniques, Histogram of Gradients (HOG) and Deep Learning, have been used in the attribute extraction section. The properties obtained are classified using the multi-class Support Vector Machines (SVM) method.

July 2017, 79 pages

Key Words: Face Recognition, Deep Learning, Convolutional Neural Network, Face Recognition Algorithm, Artificial Neural Networks, MATLAB

TEŞEKKÜR

Tez çalışmam süresince çalışmalarımı yönlendiren, araştırmalarımın her aşamasında bilgi, öneri ve yardımlarını esirgemeyerek akademik ortamda olduğu kadar beşeri ilişkilerde de engin fikirleriyle yetişme ve gelişmeye katkıda bulunan, zor zamanlarda tüm içtenliğiyle destek olup yardımcı olan danışman hocam sayın Yrd. Doç. Dr. Mehmet Serdar GÜZEL (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) ve çalışmalarım süresince maddi manevi desteklerini esirgemeyen değerli bölüm başkanımız sayın Prof. Dr. İman ASKERBEYLİ'ye (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) en derin sevgi ve saygılarımla teşekkür ederim.

Hayatım boyunca karşılaştığım her türlü zorlukta elimi tutan, sakin ve dik duruşuyla yanımda olup güven veren, bizler için hiçbir fedakarlıktan kaçınmamış babama, anneme ve kardeşlerime en derin sevgi ve saygılarımla teşekkür ederim.

Birlikte geçirdiğimiz zaman içinde iyi ve kötü günü gördüğümüz, bu süreçte her zaman bana destek olan eşime, maddi ve manevi her anlamda yanımda olan değerli ailesine en derin sevgi ve saygılarımla teşekkür ederim.

Tez çalışması boyunca aynı süreçten geçtiğimiz, her an desteğini hissettiğim başta değerli dostum Büşra EMEK SOYLU olmak üzere çalışma arkadaşlarıma en derin duygularla teşekkür ederim.

Fatıma Zehra ÜNAL

Ankara, Temmuz 2017

İÇİNDEKİLER

TEZ ONAY SAYFASI	
ETİK.....	i
ÖZET.....	ii
ABSTRACT	iii
TEŞEKKÜR	iv
KISALTMALAR DİZİNİ	vi
ŞEKİLLER DİZİNİ	vii
ÇİZELGELER DİZİNİ	ix
1. GİRİŞ	1
2. KURAMSAL TEMELLER ve KAYNAK ÖZETLERİ	4
2.1 Özellik Tabanlı Yöntemler	4
2.2 Bütüncül Yöntemler (Holistic Methods)	8
2.3 Derin Öğrenme ile Geliştirilmiş Çalışmalar	11
2.3.1 Derin öğrenme yöntemi kullanılarak yapılmış yüz tanıma uygulamaları.....	14
3. GELİŞTİRİLEN YÖNTEM.....	20
3.1 Yüz Tanıma Çalışmalarında Gerçekleştirilen Temel Konsept.....	20
3.2 Yüz Tanıma Aşamaları ve Kullanılan Yöntemler.....	21
3.3 Viola&Jones Algoritması.....	21
3.3.1 Haar özellikleri.....	22
3.3.2 Integral görüntü	23
3.3.3 Adaboost eğitimi.....	24
3.3.4 Cascading sınıflandırıcı	24
3.4 Histogram of Gradients Algoritması	25
3.5 Derin öğrenme ve Çalışma Prensipleri	27
3.5.1 Derin öğrenme mimarisi.....	35
3.6 Destek Vektör Makineleri	46
3.6.1 İki sınıflı doğrusal destek vektör makineleri.....	48
3.6.2 İki sınıflı doğrusal olmayan destek vektör makineleri	50
3.6.3 Çok sınıflı destek vektör makineleri.....	52
3.7 Tez Kapsamında Geliştirilen Uygulamanın Açıklaması	54
3.7.1 Klasik yöntem ile yüz tanıma algoritması.....	55
3.7.2 Derin öğrenme ile yüz tanıma algoritması.....	58
4. BULGULAR	64
5. SONUÇ ve TARTIŞMA	72
KAYNAKLAR	73
ÖZGEÇMİŞ.....	79

KISALTMALAR DİZİNİ

BBA	Bağımsız Bileşen Analizi
DVM	Destek Vektör Makineleri
EGÇE	Elastik Grup Çizgesi Eşleme
HOG	Histogram of Gradients
KSA	Konvolüsyonel Sinir Ağı
LFW	Labelled Faces in the Wild
LRN	Local Response Normalization
ReLU	Rectified Linear Unit
SIFT	Scale Invariant Feature Transform
TCA	Temel Bileşenler Analizi
YSA	Yapay Sinir Ağları



ŞEKİLLER DİZİNİ

Şekil 2.1 Geometrik özelliklerin gösterimi (Jafri ve Arabnia 2009).....	5
Şekil 2.2 Elastik Grup Çizgesi Eşleme yöntemi ile referans noktalarının tespiti (Wiskott vd. 1997)	6
Şekil 2.3 Sahne ayırıştırma sistemi diagramı (Farabet vd. 2013)	12
Şekil 2.4 Gezen pencere konvolüsyon ağ mimarisi (Tompson vd. 2014)	13
Şekil 2.5 Kullanılan sinir ağı (Cheung 2012)	15
Şekil 2.6 Kullanılan sinir ağı (Sun ve Tang 2014).....	16
Şekil 2.7 Kullanılan sinir ağı (Taigman vd. 2014).....	17
Şekil 2.8 Kullanılan sinir ağı (Sun vd. 2014).....	18
Şekil 2.9 Kullanılan sinir ağı (Syafeeza 2016)	19
Şekil 3.1 Yüz tanıma algoritması	20
Şekil 3.2 Yüz tanıma akış şeması.....	21
Şekil 3.3 Haar özellikleri (Viola ve Jones 2004)	22
Şekil 3.4 İntegral görüntü gösterimi (Viola ve Jones 2004)	23
Şekil 3.5 İntegral görüntü hesaplanması (Viola ve Jones 2004).....	24
Şekil 3.6 Cascading classifier çalışma prensibi (Viola ve Jones 2004)	25
Şekil 3.8 HOG algoritması çalışma şeması (Anonymous 2017a).....	27
Şekil 3.8 Kavramlar arası ilişki.....	28
Şekil 3.9 Biyolojik sinir hücresi yapısı	29
Şekil 3.10 Yapay sinir hücresi modeli (Anonymous 2017b).....	30
Şekil 3.11 Aktivasyon fonksiyonları (Anonymous 2017c).....	31
Şekil 3.12 Derin öğrenme mimarisi (Anonymous 2017e)	34
Şekil 3.13 Özellik öğrenimi (Anonymous 2017d)	35
Şekil 3.16 Konvolüsyonel sinir ağı mimarisi örneği.....	35
Şekil 3.15 Konvolüsyon işlemi (Wu 2016).....	37
Şekil 3.16 Adım işlemi.....	38
Şekil 3.17 Konvolüsyon örneği (Anonymous 2017g)	39
Şekil 3.18 Filtre Kullanım örneği (Anonymous 2017g)	40
Şekil 3.19 Aktivasyon fonksiyonu (Wuo 2016).....	41
Şekil 3.20 ReLU işlemi (Anonymous 2017d).....	42
Şekil 3.21 Havuzlama katmanı işlevi (Anonymous 2017f)	42
Şekil 3.22 Havuzlama katmanı (Anonymous 2017f).....	43
Şekil 3.23 Tam bağlı katmanı (O'Shea ve Nash 2015).....	44
Şekil 3.24 Destek vektör makinelerinde sınıflandırmanın kavramsal açıklaması (Vapnik ve Cortes 1995)	47
Şekil 3.25 Destek vektör makineleri çalışma prensibi	48
Şekil 3. 26 İki boyutlu uzayda ayrılabilir DVM örneği	49
Şekil 3. 27 İki boyutlu uzayda ayrılabilir DVM örneği	49
Şekil 3.28 İki sınıflı problem örneği (Başer ve Apaydın 2015).....	50
Şekil 3.29 İki Sınıflı Problem İçin Doğrusal Ayrılama Durumu (Başer ve Apaydın 2015)	51
Şekil 3.30 İki boyutlu veri setinin üç boyutlu özellik uzayına taşınması (Cortes ve Vapnik 1995)	52
Şekil 3.31 İkili karar ağacı (Guo vd. 2000).....	53
Şekil 3.32 Orijinal resim	55

Şekil 3.33 Gri renge çevrilmiş resim.....	56
Şekil 3.34 Yüzün tespit edilmesi.....	56
Şekil 3.35 Kırpılmış resim	57
Şekil 3.36 HOG özellikleri.....	57
Şekil 3.37 Yüz tanıma işlemi	58
Şekil 4.1 Klasik yöntem ile yüz tanıma performans oranı	67
Şekil 4.2 Derin öğrenme ile yüz tanıma performans oranı	69
Şekil 4.3 Yöntemlerin başarı oranları karşılaştırılması.....	69
Şekil 4.4 lfw veri setinde iki yöntem karşılaştırılması.....	70
Şekil 4.5 faces96 veri setinde iki yöntem karşılaştırılması.....	70
Şekil 4.6 att_faces veri setinde iki yöntem karşılaştırılması.....	71



ÇİZELGELER DİZİNİ

Çizelge 3.1 AlexNet mimarisi.....	61
Çizelge 4.1 At&t veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi.....	65
Çizelge 4.2 Faces96 veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi.....	65
Çizelge 4.3 Lfw veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi	66
Çizelge 4.4 At&t veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi.....	67
Çizelge 4.5 Faces96 veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi.....	68
Çizelge 4.6 Lfw veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi	68



1. GİRİŞ

Çok sayıda alanı ve disiplini kapsayan yüz tanıma, görüntü analizi alanının amacına en uygun önemli bir araştırma problemidir. İnsanların yüzleri tanıma kabiliyetine eşit, otomatik bir sistem geliştirmek ciddi bir problemdir. İnsanlar bilinen yüzleri tanımlamakta oldukça iyi olmasına rağmen, büyük miktarda bilinmeyenliğin üstesinden gelmek gerektiğinde çok yetenekli değildir. Bilgisayarlar neredeyse sınırsız bellek ve hesaplama hızı sayesinde insanın kısıtlı kaldığı durumları aşabilir. Yüz tanıma; örüntü tanıma, sinir ağları, görüntü işleme ve psikoloji ile de ilgilidir. Farklı disiplinlere konu olması, bu alanda yapılan ve yapılacak çalışmalara olan ilgiyi arttırmaktadır (Carrera ve Marques 2010). Gerçek dünya uygulamalarının sayısı (örneğin, gözetim, güvenli erişim, kişisel bilgisayar arayüzü), ucuz ve güçlü donanımın bulunması, yüz tanıma sistemlerinin geliştirilmesine yol açmaktadır (Heisele vd. 2001).

Yüz tanıma, bilgisayarlarla görme alanında (computer vision) yoğun olarak incelenen konulardan biri olan yüz tanıma uygulamaları, parmak izi, iris ve retina tanıma gibi biyometri uygulamaları ile karşılaştırıldığında, kişinin direkt kendine ihtiyaç duyulmadan tanıma potansiyeline sahiptir. Bu nedenle güvenlik izleme sistemleri, sınır kontrolü, adli tıp, dijital eğlence gibi uygulamalarda kullanılabilir. Nitekim, yüz tanıma alanında gözetim kameralarından suçluların başarılı bir şekilde tespitinden popüler veritabanından insan düzeyinde tanıma performansına kadar birçok çalışma gerçekleştirilmiş ve büyük ilerleme sağlanmıştır (Ding ve Tao 2016).

Yüz tanıma uygulamaları banka kartı kimlik saptama, giriş kontrolü, güvenlik izleme ve gözetim sistemleri gibi birçok pratik uygulamalarda kullanılmasının yanı sıra etkili iletişim ve insanlar arasındaki etkileşim için gerekli olan temel bir insani davranıştır. Yüzleri sınıflandırmak için ilk resmi bir yöntem Chellappa vd. (1995) yapmış olduğu çalışmada belirtilmiştir. Belirtilen çalışma, yüz profillerini eğri olarak toplamayı ve normlarını bulmayı ve ardından normdan sapma durumlarına göre diğer profilleri kendi sınıflarına göre sınıflandırmayı önermiştir. Bu sınıflandırma çok modellenmiş sınıflandırma olarak geçmiştir. Yüz tanıma algoritmalarının gerçek zamanlı uygulamalarda kullanılabilirliğinin ispat edilmesi bu alanda ilerlemelere yol açmıştır.

Yüz tanıma tekniklerinin kullanım alanları güvenlik uygulamaları ve ticari uygulamalar olmak üzere iki ana başlık altında kategorize edilebilir. Yüz tanıma algoritmaları öncelikle sabıka kaydı için çekilen resim albümleri ve güvenlik kamera sistemleri (video görüntü dizilerine göre gerçek zamanlı eşleme) gibi güvenlik uygulamalarında kullanılmaktadır. Ticari uygulamalarda ise kredi kartlarının üzerinde bulunan resimden statik eşleme, ATM kartları, pasaportlar, ehliyet ve kimlik kartı ile resimden veya video görüntüsü üzerinden gerçek zamanlı işlemleri gibi uygulamalarda kullanılmaktadır (Tolba vd. 2006).

Yüz tanıma uygulamaları güvenlik alanında başarılı sonuçlar verdiği için son derece önemli hale gelmiştir. Bu tür sistemler için Yapay Zeka, tanıma ve doğrulama görevlerinde önemli bir rol oynamaktadır. Yapay olarak bir şeyleri tanıma işlemi, kodları ve özellikleri harekete geçirerek denetlenen (supervised) ve denetlenmeyen (unsupervised) öğrenme mekanizmaları yolu ile yapılabilmektedir. Ancak bu öğrenme, kişi görüntüleri kontrollü koşullarda iken başarılı bir şekilde uygulanabilmektedir. Yüz ifadesi değişiklikleri, baş yönlendirmeleri ve aydınlatma koşullarının değişmesi gibi kontrol edilemeyen durum olduğunda tanıma işlemi zorlaşmaktadır. Bu gibi durumlarda özellik çıkarımı ve sınıflandırma, bilgisayarlarla görme uygulamaları için kritik görev haline gelmektedir (Sharif. vd 2012).

Uzun yıllar yapay zeka alanında geliştirilen uygulamalar ile sonuç alınamamış sorunların çözümünde Derin Öğrenme teknolojisi ile büyük ilerleme kaydedilmiştir. Yüksek boyutlu verilerin karmaşık yapılarını keşfetme konusunda çok iyi olduğu ortaya çıkan Derin öğrenme, bilim, iş dünyası ve devletin birçok alanında uygulanmaya başlamıştır (LeCunn vd. 2015).

Işıklar, engeller, farklı yüz ifadeleri ve yüz konumlarından kaynaklanan görüntülerde olası yüksek değişiklik farkı, sınıflamayla ilgili bu sorunlara karşı değişmeyen bir yüz tanıma algoritmanın bulunmasını zorlaştırmaktadır. Geçmiş yıllarda, özyüz (eigenface) gibi yöntemler yüz tanıma sorunları için geliştirilen algoritmalarıdır. Geçen zaman içinde, Derin Öğrenme teknikleri, özellikle Konvolüsyonel Sinir Ağları kullanarak bilgisayarlarla görme alanında birçok sınıflandırma problemi performansı arttırılmıştır

(Vilardi 2016). Derin öğrenme, özellikle Konvolüsyonel Sinir Ağı (KSA), son zamanlarda yüz tanıma uygulamalarında umut verici sonuçlar elde etmiştir.

Bu tez kapsamında, geleneksel makine öğrenme tekniği ve derin öğrenme tekniği kullanılarak bir yüz tanıma algoritması eğitilmiş ve iki ayrı yöntemin başarı oranlarını karşılaştırılmıştır. Bu amaçla yüz tanıma algoritması kurulmuştur. Algoritma, yüz tespiti, ön işlemler, öznitelik çıkarımı ve sınıflandırma olmak üzere 4 ana kısımdan oluşmaktadır. Öznitelik çıkarımı kısmında Histogram of Gradients (HOG) ve derin öğrenme olmak üzere iki ayrı teknik kullanılmıştır. Elde edilen özellikler çok sınıflı Destek Vektör Makineleri (DVM) metodu kullanılarak sınıflandırılmış ve başarı oranları karşılaştırılmıştır.

Çalışmanın ikinci bölümünde yüz tanıma uygulamalarında kullanılan yöntemler, derin öğrenme alanında yapılan uygulamalar ve derin öğrenme kullanılarak geliştirilen yüz tanıma uygulamaları incelenmiştir. Çalışmanın üçüncü bölümünde geliştirilen algoritma ve kullanılan yöntemler detaylı bir şekilde açıklanmıştır. Çalışmanın dördüncü bölümünde çalışma sonucunda elde edilen bulgular belirtilmiştir. Son bölümde tez kapsamında yapılan çalışma özetlenmiş ve değerlendirmeler yapılmıştır.

2. KURAMSAL TEMELLER ve KAYNAK ÖZETLERİ

Bu bölümde, yüz tanıma uygulamalarında kullanılan yöntemler, derin öğrenme alanında yapılan çalışmalar ve derin öğrenme kullanılarak geliştirilen yüz tanıma uygulamaları incelenmiştir.

Yüz tanıma teknikleri yüz verisi elde etme yöntemine bağlı olarak geniş şekilde 3 kategoriye ayrılabilir. Bunlar; görüntü (resim) üzerinde, video serisi üzerinde ve 3 boyutlu bilgi ve kızılötesi görüntü üzerinde çalışan yöntemlerdir (Jafri ve Arabnia 2009). Tez çalışması resim üzerinden yüz tanıma uygulaması olduğu için, resim üzerinde yapılan yüz tanıma teknikleri incelenmiştir.

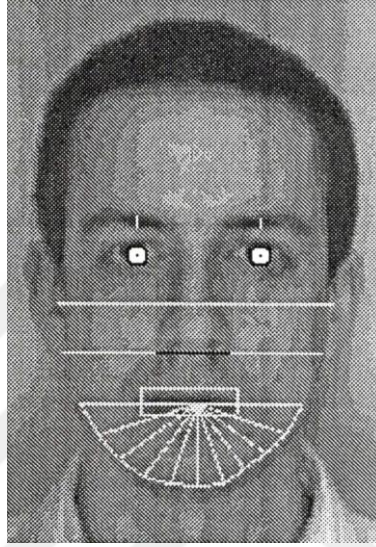
Çalışmada yüz tanıma teknikleri, özellik tabanlı (feature based), bütüncül (holistic) yöntemler ve derin öğrenme olmak üzere üç ana başlık altında toplanmıştır.

2.1 Özellik Tabanlı Yöntemler

Özellik tabanlı yöntemler, gözler, burun, ağız gibi yüz bileşenlerini kullanarak yüzleri tanımaya çalışmaktadırlar. Özellik tabanlı yaklaşımların genel olarak çalışma prensipleri şu şekildedir. Öncelikle, gözler, ağız, burun vb. ayırt edici yüz özelliklerini aynı zamanda yüz referans noktalarını ve bu noktalar arasındaki geometrik ilişkiyi tespit etmek ve çıkarmak için girdi görüntüsünü (input image) işlerler. Böylece, girdi yüz görüntüsü geometrik özelliklere sahip bir vektöre dönüştürülür. Daha sonra yüzleri eşleştirmek için standart istatistiksel model tanıma teknikleri kullanılır (Jafri ve Arabnia 2009). Otomatik yüz tanıma uygulamalarında yapılan ilk çalışmalar çoğunlukla bu teknikler üzerine kurulmuştur.

Brunelli ve Poggio (1993) özellik tabanlı yöntem kullanarak yapmış oldukları çalışmada 6 sı kadın, 21 i erkek olmak üzere 47 farklı kişinin toplam 188 fotoğrafı kullanılmıştır. Burun genişliği ve uzunluğu, ağız konumu ve çene şekli gibi 35 geometrik özellik kullanılarak Bayes sınıflandırıcısı ile tanıma işlemi gerçekleştirilmiş ve %90 başarı elde

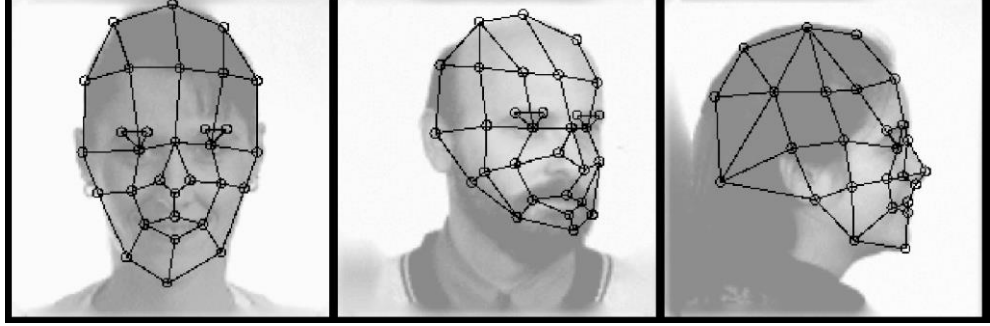
edilmiştir. Geliştirdikleri bu yöntem ile kalitesiz çözünürlüğe sahip görüntülerde bile yüz özelliklerinin genel geometrik şekli kullanılarak yüz tanıma işleminin gerçekleştirilebileceğini ispatlamışlardır. Genel geometrik özellikler ana yüz özelliklerinin (kaşlar, gözler ve burun gibi) konumu ve boyutunu temsil eden sayısal verilerin bir vektörü olarak tanımlanabilir.



Şekil 2.1 Geometrik özelliklerin gösterimi (Jafri ve Arabnia 2009)

Yüz tanıma uygulamaları için önerilen yöntemlerden bir tanesi de çizge eşleme yaklaşımıdır. (Graph matching approach) Çizge eşleme yaklaşımları, depolanan grafikleri eşleştirmek için harcanan zaman çok fazla olduğu için gerçek zamanlı ortamlar için ideal değildir (Sharif vd. 2012). Bu kapsamda Wiskott ve arkadaşları tarafından (1997) önerilen yöntem Elastik Grup Çizgesi Eşleme yöntemidir (Elastic Bunch Graph Matching). Bu yöntem ile 250 kişilik veri seti kullanılarak yapılan yüz tanıma uygulamalarında %99 başarı elde edilmiştir. Uygulamanın temeli olan yüz grafik üretimi şu şekilde gerçekleştirilir: yüzdeki referans noktaları kümesi seçilir (Her yüz için tek olan ve değişmez özellikler gibi). Her bir referans noktası bütün grafikteki bir düğümü ifade eder ve bu noktanın etrafındaki pencereye uygulanan gabor filtrelerinin cevabına göre referans noktası etiketlenir. Referans noktalarını birbirine bağlayan yay, karşılıklı referans noktaları arasındaki mesafe ile etiketlenmiştir. Her bir yüz için elde edilen grafikler yığın yapısına benzeyen, yüz grup grafiği (face bunch graph) adı verilen

yapı altında birleştirilir. Yüz grup grafiği sayesinde, yeni yüz görüntüleri için grafikler otomatik olarak Elastik Grup Çizgesi Eşleme yöntemi ile oluşturulur.



Şekil 2.2 Elastik Grup Çizgesi Eşleme yöntemi ile referans noktalarının tespiti (Wiskott vd. 1997)

Yeni bir yüz görüntüsünün tanınması, yeni görüntü grafiğinin veri setinde bulunan bütün yüz görüntüleri ile karşılaştırılıp, en yüksek benzerlik olan değer alınması ile yapılır (Wiskott vd. 1997). Bu mimari kullanılarak yapılan uygulamalarda yüksek başarı elde edilse bile, bu yöntem uygulanmadan önceki 70 yüzün grafik çıkarımı manuel olarak yapılması gerektiği için dezavantajları vardır (Jafri ve Arabnia 2009).

Pervaiz (2010), Elastik Grup Çizgesi Eşleme algoritmasını kullanılarak yapılan uygulamada, değişen aydınlatma ve ifade altında sağlam tanıma oranı sağlanmıştır. Nihayet, yüz grafiği eşlemesi, referans görüntüsüyle yüzleri tanımak için kullanılmıştır. Böylece gerçek zamanlı ortamda tanıma gerçekleştirilmiştir.

Campadelli ve Lanzaretti (2005) tarafından, Elastik Grup Çizgesi Eşleme yönteminden esinlenerek gerçekleştirilen çalışma ile yukarıda belirtilen dezavantaj çözülmüştür. Yaptıkları çalışmada grafik yerleşiminin manuel yapılması ihtiyacını parametrik modeller kullanarak aşmışlardır. Böylece referans noktalarının yerinin belirlenmesi kısmı otomatik olarak gerçekleştirilerek tamamen otomatik yüz tanıma sistemi geliştirilmiştir. Bu yaklaşımın diğer çeşitleri grafik işleme stratejisi ve HOG dur.

Gabor filtre yanıtları, doku segmentasyonu, yüz tespiti ve tanıma, iris tanıma gibi pek çok bilgisayarlarla görme alanındaki uygulamalarda genel amaçlı özellikler olarak sıkça

ve başarılı bir şekilde uygulanır. Tipik bir özellik yapısında Gabor filtreleri, birçok farklı frekanslara ve yönlendirmelere ayarlanmış filtrelerden oluşan çok çözünürlüklü yapıları kullanırlar. Çok çözünürlüklü yapı Gabor özelliklerini dalgacıklarla ilişkilendirir. Gabor filtreleri herhangi bir doğrusal filtreye karşılık geldiğinden en basit teknik Filtreleme işleminin gerçekleştirilmesi mekansal alanda konvolüsyon yoluyla gerçekleştirilir (Ilonen vd. 2005).

Albiol vd. (2008) tarafından EGÇE tabanlı yeni bir yüz tanıma algoritması geliştirilmiştir. Özellik çıkarımı aşamasında HOG tanımlayıcıları ve Gabor özellikleri performansları karşılaştırılmış; aydınlatma ve rotasyon değişikliklerine karşı daha sağlam olduğu, elde edilen yüz grafiklerindeki daha yüksek doğruluk özellikleriyle açıklanmıştır.

Déniz vd. (2011) tarafında yapılan çalışmada, yüz tanıma için HOG özelliklerinin güçlü bir şekilde kullanılması için basit ama güçlü bir yaklaşım araştırılmıştır. Belirtilen çalışmada EGGE yöntemi kullanılarak 25 yüz referans noktası lokalize edilmiştir. Referans noktalarının her birinin çevresinden çıkarılan HOG özellikleri, en yakın komşu ve Öklid uzaklığı kullanılarak sınıflandırma için kullanılmıştır. Yüz tanıma, poz ve aydınlatma değişikliklerine bağlı olarak yüz özellik tespitinde oluşan hataları telafi etmek için, düzenli bir ızgaradan HOG tanımlayıcıları çıkarılması önerilmiştir. Farklı ölçeklerde HOG tanımlayıcılarının bir araya getirilmesi ile yüz tanıma için önemli bir yapının yakalanması sağlanmıştır. Gürültüyü gidermek ve sınıflandırma sürecinde oluşabilecek ezberleme olayını (overfitting) azaltmak için boyutsal azaltmanın gerçekleştirilmesinin gerekliliği tespit edilmiştir. Son olarak önerilen yöntem 4 ayrı veri setinde test edilmiş ve sonuçların başarılı olduğu gözlenmiştir.

Çekirdek makine-bazlı Ayrımcılık analiz yöntemi (Kernel Direct Discriminant), Juwei vd. (2003) tarafından önerilmiştir. Geliştirilmiş ayırım yapabilme gücüne sahip düşük boyutlu özellik gösterimini sunabilen teknikler, yüz tanıma sistemlerinde büyük önem taşır. Bilindiği gibi, yüz imgelerinin dağılımı, bakış açıları, aydınlatma gibi yüz ifadesinde algılanabilir bir varyasyon altında son derece karmaşık ve doğrusal değildir. Bu nedenle doğrusal teknikler karmaşık yüz varyasyonlarına sahip yüz tanıma

uygulamalarında güvenilir ve sağlam çözümler sağlayamamaktadır. Çekirdek makine-bazlı Ayrımcılık analiz yöntemi doğrusal olmayan yüz modellerinin dağılımı ile ilgilenmektedir. Yüz tanıma uygulamalarında karşılaşılan küçük örnek boyutu (small sample size, sss) etkili bir şekilde çözer. Geliştirilen algoritma sınıflandırma hata oranı performansına göre UMIST veri setinde test edilmiş. Çok küçük özellik kümesi ile başarılı bir performans elde edildiği görülmüştür.

Özellik çıkarma işleminin yüz tanıma uygulamalarında kişi eşleştirme işleminden önce yapılması, girdi görüntülerinin konum değişikliklerine karşı dirençli olmasını sağlamıştır. Bu özellik tabanlı yöntemlerin temel avantajıdır. Ayrıca yüz görüntülerinin ifade edilmesindeki kompaktlık ve yüksek hızlı eşleme diğer avantajları arasındadır. Otomatik özellik tespitinin olmayışı ve bu yüzden bu tarz tekniklerin hangi özelliğin önemli olduğu hususunda rastgele karar vermek zorunda olmaları başlıca dezavantajıdır. Özellikleri ayırtma konusundaki eksiklik sonraki işlemlerin başarısını etkilemektedir (Jafri ve Arabnia 2009).

2.2 Bütüncül Yöntemler (Holistic Methods)

Bütüncül yöntemler, yüzleri tanımlamak için yüzün yerel özelliklerinden çok resmin tamamına dayalı genel gösterimler üzerinde çalışırlar. Yapay zeka yaklaşımları da bu başlık altında ele alınmaktadır.

Özvektör ya da Temel Bileşen Analizi (Principal Component Analysis) diye de adlandırılan özyüz yöntemi (eigenfaces) yüz tanıma uygulamalarında kullanılan genel yöntemlerden biridir. İlk olarak Sirovich ve Kirby 1987 ve 1990 yılında yaptıkları çalışmalarda yüzlerin geometrisini etkili bir şekilde karakterize etmek için bu yöntemi kullanmışlardır. Yazarlara göre, yüz sadece özyüzler kullanılarak elde edilebilecek az miktarda bilgi ile kolayca oluşturulabilir (Sharif vd. 2012). Yaptıkları çalışmalarda, özyüz koordinat alanı boyunca belirli bir yüzün verimli bir şekilde temsil edilebileceğini ve herhangi bir yüzün sadece küçük bir özyüz topluluğu ve her bir özyüze karşılık gelen izdüşümlerin kullanılarak yeniden oluşturulabileceğini önermişlerdir. Özyüz tabanlı yöntemlerin en büyük problemi, yüzlerin bir dizi özyüzün doğrusal kombinasyonları ile

yeniden oluşturulabilen katı, sabit nesnelere olduğunu varsaymaktır ancak bu doğru değildir. Ayrıca aydınlatma koşulları değiştiğinde bu yöntemlerin performansı hızla düşmektedir. Bu zorlukların üstesinden gelmek için orijinal yöntemlerin birçok varyasyonu önerilmiştir (Albiol vd. 2008).

Bağımsız Bileşen Analizi yöntemi (Independent Component Analysis), yüz tanıma uygulamaları için kullanılan yöntemlerden bir tanesidir. Temel Bileşenler Analizi (Principal Component Analysis) yönteminin geliştirilmiş hali olan bu yöntemin amacı, birbirleriyle ilgisiz, görüntüden bağımsız bir görüntü ayırması ve temsil etmektir (Comon, 1994). BBA, Temel Bileşen Analizi (PCA) 'nın geliştirilmesi olarak düşünülen bir özellik çıkarım tekniğidir ve esas olarak kör sinyal ayırma problemi (blind signal separation) üzerinde kullanılmıştır. TCA girdilerin ilişkisiz değişkenlere dayalı bir gösterimini elde etmeye çalışırken, BBA istatistiksel olarak bağımsız değişkenlere dayalı bir temsil sunmaktadır (Deniz vd. 2001).

Türk ve Petland (1991), Sirovich ve Kirby'nin çalışmasını esas alarak, özyüz izdüşümlerini, yüzleri tanımak için sınıflandırma özellikleri olarak kullanmışlardır. Geliştirilen yöntem, yüz tanıma problemine, üç boyutlu geometrinin iyileştirilmesini gerektirmek yerine, yüzlerin normal dik olduğu iki boyutlu tanıma problemi olarak yaklaşır. Yöntem, 16 yaşın altındaki toplam 2500 resimden oluşan veri seti kullanılarak test edilmiştir. Kafa oryantasyonu, aydınlatma koşulları ve çeşitli çözünürlük koşulları altında denendiğinde % 96, % 85 ve % 64'lük tanıma oranları elde edilmiştir (Jafri ve Arabnia 2009).

Şablon eşleştirme yöntemi (template matching), bir kişinin yüzünü tanımlamak, temsil etmek için farklı bakış açılarına sahip birden fazla yüz şablonlarının kullanılması mantığına dayanmaktadır. Bu yöntem, özellik tabanlı yöntemler ile karşılaştırıldığında daha başarılı sonuçlar verdiği görülmüştür (Sharif vd. 2012) . Bruneli ve Poggio (1993), 47 kişinin toplam 188 resminden oluşan verisini kullanarak yaptıkları çalışmada şablon eşleştirme yöntemi ile %100 başarı oranı alırken, özellik tabanlı method kullanarak % 90 oranında başarı elde etmişlerdir. Konuşma tanıma uygulamaları için ideal olan Gizli Markov Modeli (Hidden Markov Model) yüz tanıma uygulamalarında

da kullanılmıştır. Samara ve Harter (1994) tarafından, tek boyutlu gizli markov modeli kullanılarak ORL veriseti üzerinde yapılan yüz tanıma çalışmasında % 87 başarı elde edilmiştir. Aynı veri seti kullanılarak Nefian vd. (1999) tarafından yapılan yüz tanıma uygulamasında % 98 başarı oranı elde edilmiştir.

Model sınıflandırma problemlerinin çözümünde en etkili algoritmalarından biri olan Destek Vektör Makineleri yüz tanıma uygulamalarında da çok verimli sonuçlar vermektedir. Vapnik ve Cortes (1995) yaptıkları çalışmada DVM'nin ayrılabilir olmayan eğitim verilerinde yüksek genelleme yeteneğini gösterilmişlerdir. Philips (1998), DVM tabanlı aynı kişinin yüzleri arasındaki benzerlikler ve farklı kişilerin yüzleri arasındaki farklılıklar olmak üzere iki sınıflı yüz tanıma uygulamasını FERET veri seti üzerinde geliştirmiştir. Aynı veri seti üzerinde temel bileşenler analizi yöntemi kullanılarak yüz tanıma uygulaması geliştirilmiş ve sonuçlar karşılaştırılmıştır. DVM yöntemi başarı oranı % 78 iken TBA yöntemi başarı oranı % 54 olmuştur.

Guo vd. (2000), yaptıkları çalışmada ikili ağaç tanıma stratejisine sahip DVM leri yüz tanıma problemini çözmek için kullanmıştır. DVM tabanlı uygulama ile özyüz tabanlı geliştirilen uygulama başarı sonuçları karşılaştırılmıştır. 40 kişinin toplam 400 resminden oluşan Cambridge ORL veri seti üzerinde gerçekleştirilen çalışmada DVM tabanlı yöntem en küçük hata oranı % 3 iken özyüz tabanlı yöntem en küçük hata oranı % 5,25 olmuştur.

Deniz vd. (2001) tarafından, iki güçlü teknik olan BBA ve DVM yöntemleri kullanılarak yüz tanıma uygulaması geliştirilmiştir. İki farklı veri seti üzerinde gerçekleştirilen çalışmada BBA özellik çıkarıcı olarak DVM ise sınıflandırıcı olarak kullanılmış ve çok yüksek sonuçlar alınmıştır. Ayrıca BBA/DVM ve TBA/DVM yöntemleri ayrı ayrı denenmiş ve iki kombinasyon sonuçları arasında büyük bir fark olmadığı gözlenmiştir. Ancak BBA/DVM eğitim süresinin TBA/DVM eğitim süresinden daha uzun olduğu için, en verimli ve pratik kombinasyonun TBA/DVM olduğu görülmüştür.

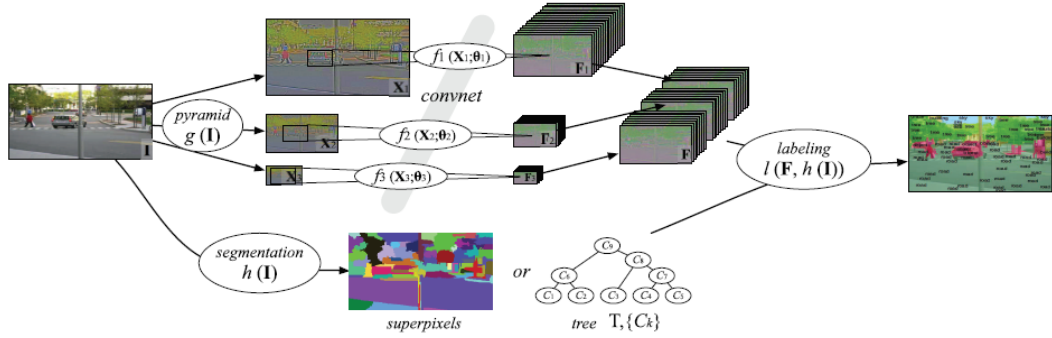
Jonsson vd. (2000) yüz tanıma ve doğrulama uygulaması için DVM leri kullanmışlardır. Yapılan çalışmada DVM lerin ışıklandırma koşulları değişikliklerine karşı sağlam olduğunu belirtilmiştir. Önerilen sistem 295 kişiden oluşan veri seti üzerinde denenmiş; yüz tanımda % hata oranı ve yüz doğrulamada %1 hata oranı görülmüştür.

2.3 Derin Öğrenme ile Geliştirilmiş Çalışmalar

Krizhevsky vd. (2012) derin öğrenme yöntemini ile ImageNet veriseti kullanılarak nesne tanıma uygulaması geliştirmişlerdir. Yapılan bu çalışma ile, konvansiyonel ağların nesne tanıma için hata oranını neredeyse yarıya indiren ve bilgisayarla görme alanında çalışan topluluk tarafından derin öğrenmenin hızla benimsenmesini sağlayan bir gelişme olmuştur. Derin konvolüsyonel sinir ağı kullanarak katıldıkları LSVRC-2010 (Pascal Visual Object Challenge'in) yarışmasında; ImageNet veri setinde bulunan 1.2 milyonluk yüksek çözünürlüklü görüntünün 1000 farklı kategoride sınıflandırılması sağlanmış ve daha önce kullanılan teknolojiden daha iyi sonuçlar elde edilmiştir. 60 milyon parametre ve 650.000 nöron içeren sinir ağı, beş konvolüsyon katmanından oluşur; bunlardan bazılarını takip eden maksimum havuz katmanları, 3 tam bağlı katman ve 1000 softmax kullanılarak sonlandırılmıştır. Daha hızlı eğitim yapmak için doymuş olmayan (non saturating) nöronlar ve etkili konvolüsyon işlemi için çok verimli bir GPU uygulaması kullanılmıştır. Tam bağlanmış tabakalardaki aşırı uyumsuzluğu (overfitting) azaltmak için, son zamanlarda geliştirilen, "Dropout" adlı normalleştirme yöntemini kullanılmış ve bu yöntemin çok etkili olduğu kanıtlanmıştır.

Farabet vd. (2013) derin öğrenme yöntemlerine dayanan sahne ayrıştırma, sahne etiketleme olarak da bilinen sistemi geliştirmişlerdir. Sahne etiketleme, bir görüntünün içindeki her pikselin ait olduğu nesnenin kategorisiyle etiketlenmesinden oluşur. Görüntü anlama, ayrıştırma, çok çeşitli pratik uygulamalar için birincil öneme sahip bir görevdir. Bir görüntüyü anlamaya yönelik önemli bir adım, görüntüdeki her pikseli ait olduğu nesnenin kategorisine etiketlemeyi içeren, sahne ayrıştırma olarak da bilinen tam sahne etiketlemeyi gerçekleştirmektir. Mükemmel bir sahne ayrıştırma işleminden sonra, her bölge ve her nesne tanımlanır ve etiketlenir. Sahne ayrıştırmasının bir zorluğu, algılama, bölümlenme ve tanıma gibi geleneksel sorunları tek bir işlemle

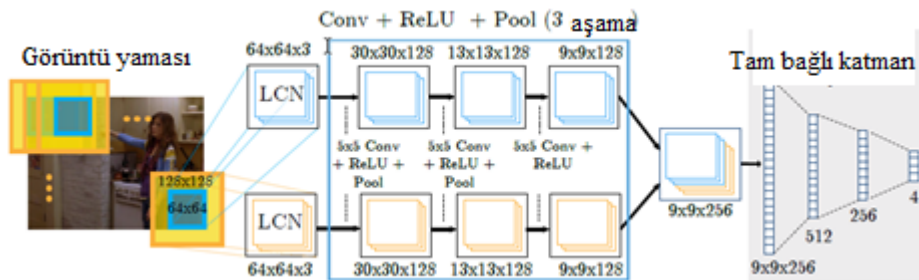
birleştirmesidir. Önerilen sistemde özellikle öznelik çıkarımı sürecinde, dokuyu, şekli ve içeriksel bilgileri yakalayan güçlü bir gösterim üreterek mühendislik özelliklerine olan ihtiyacı hafifletmektedir. Sahneyi en iyi açıklayan optimum bileşen setini otomatik olarak alınmaktadır. Yapılan çalışmada, her bir pikselde ortalananmış birden çok boyutlu bölgeleri kodlayan özellik vektörlerini çıkarmak için ham piksellerle eğitilmiş çok katmanlı konvolüsyon sinir ağı kullanılmış ve sistem SIFT akış veri seti (33 sınıf), Barcelona veri seti (175 sınıf) ve Stanford arka plan veri setleri (8 sınıf) üzerinde denenmiştir. Bir saniyeden kısa sürede, 320×240 piksellik görüntülerin özellik çıkarımı ve etiketlenmesi yapılmıştır. Bu sonuç önerilen sistemin rakip sistemlerden daha seri olduğunu kanıtlamıştır.



Şekil 2.3 Sahne ayrıştırma sistemi diagramı (Farabet vd. 2013)

Şekil 2.3’de sahne etiketleme sistemi şekil üzerinde anlatılmıştır. Ham girdi görüntüsü bir Laplace piramidi ile dönüştürülür. Her bir ölçek, özellik haritası üreten üç katmanlı konvolüsyon ağına girdi olarak verilir ve tüm ölçeklerin özellik haritaları birleştirilir, daha kaba ölçekli haritalar en iyi ölçek haritasının boyutuyla eşleştirmek için örneklenir. Her özellik vektörü, bu nedenle, her pikselin çevresindeki geniş bağlamsal pencereyi temsil eder. Buna paralel olarak, görüntünün doğal konturlarından istifade etmek için tek bir segmentasyon (yani, süper pikseller) veya segmentasyon ailesi (örneğin bir segmentasyon ağacı) hesaplanır ve son olarak etiketleme işlemi yapılır.

Tompson vd. (2014) tarafından moleküler görüntülerde insan pozı tahmini sistemi için Derin Konvolüsyon Sinir ağı ile Markov Random Field modelinden oluşan mimariyi önermişler ve bu zorlu soruna karşı başarılı sonuçlar almışlardır. Mimari, vücut bağlantı yerleri arasındaki geometrik ilişkiler gibi yapısal alan kısıtlamalarını kullanmaktadır. Konvolüsyon sinir ağı, bütün mevcut diğer yöntemlerden daha iyi performans gösterdiği gerekçesiyle vücut bölüm lokalizasyonu tespit sürecinde kullanılmıştır. Kullanılan derin öğrenme mimarisinde giriş, bir veya daha fazla kişiyi içeren bir RGB görüntüsüdür ve çıktı, insan iskeletindeki anahtar bağlantı noktaları için bir piksel olasılığı üreten bir ısı haritasıdır. Bu iki model paradigmanın müşterek eğitiminin performansı artırdığını ve mevcut teknolojik tekniklerden önemli ölçüde üstün olduğu yapılan çalışma ile kanıtlanmıştır. Bu tarz problemlerin çözümünde geleneksel ve derin öğrenme tabanlı yaklaşımlar kullanılır. İnsan vücudu doğal olarak mafsallara bölündüğünden, alttan üste parçalı modeller bu sorunun ortak bir tercihidir. Geleneksel olarak bu yaklaşımlar, SIFT veya HOG gibi elle hazırlanmış düşük seviye özelliklerin bir araya getirilmesine dayanır ve bunlar daha sonra standart bir sınıflandırıcıya girerler. Bu mühendislik özelliklerinin, algılamaya çalıştıkları parçaya duyarlı olduğundan ve girdi alanındaki sayısız deformasyona (aydınlatma değişiklikleri gibi) değişmez olmasına özen gösterilir. Öte yandan, derin öğrenme yaklaşımları, eğitim setindeki varyasyonlara genellikle daha toleranslı olan ve son zamanlarda bölüm tabanlı modellerden daha iyi olan düşük ve üst düzey özellikleri öğrenirler.



Şekil 2.4 Gezen pencere konvolüsyon ağı mimarisi (Tompson vd. 2014)

Mikolov vd. (2011), büyük veri setlerinde büyük ölçekli sinir ağı temelli dil, konuşma modellerini etkin bir şekilde nasıl eğitileceği hakkında çalışma yapmıştır. Çalışma kapsamında geliştirilen modelin muhtemelen şimdiye kadar eğitilmiş en büyük sinir ağı

tabanlı dil, konuşma modelleri olduğu ifade edilmiştir. Yaklaşık 400M simge kullanılmıştır. En büyük modelin gizli katmanının boyutu 640 nöron, kelimenin boyutu 84K kelimedir. Eğitim verileri, alaka düzeyi ile sıralandığında, eğitim sırasında hızlı yakınsama ve daha iyi genel performans gözlemlenmiştir. Sinir ağı modelinin bir parçası olarak eğitilebilen maksimum entropi modelinin karma tabanlı uygulanmasını sunulmuştur. Bu, hesaplama karmaşıklığının önemli ölçüde azalmasına neden olmaktadır. İngilizce Yayın Haberleri konuşma tanıma görevinde, 400M simge üzerinde eğitilmiş ve kelime hata oranının görece % 10 civarında olması sağlanmıştır.

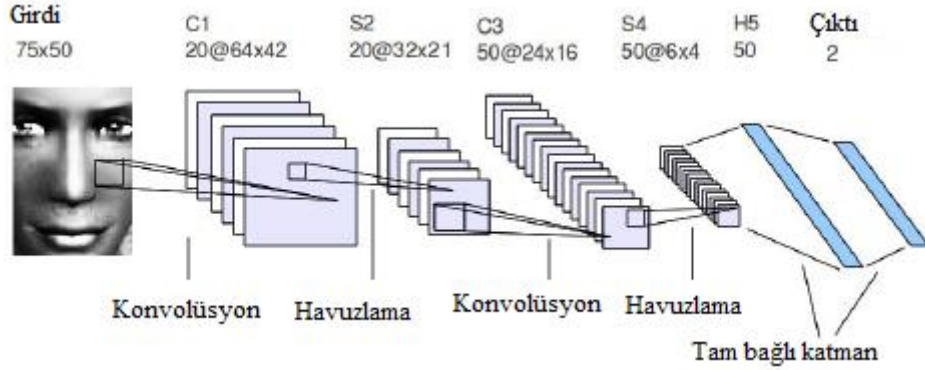
Collobert vd. (2011) tarafından, hem hız hem de doğrulukla konuşma parçası etiketleme, bölme, adlandırılmış varlık tanıma ve semantik rol etiketi gibi bir dizi doğal dil işleme görevini yerine getirebilen çok katmanlı bir sinir ağı mimarisi ve öğrenme algoritması geliştirilmiştir. Her bir görev için dikkatle optimize edilmiş insan yapımı girdi özelliklerini kullanmak yerine, sistem büyük oranda etiketlenmemiş eğitim verilerini temel alarak iç temsilleri öğrenmektedir. Bu çalışma, iyi performans ve minimum hesaplama gereksinimleri olan serbestçe temin edilebilen bir etiketleme sistemi oluşturmak için bir temel olarak kullanılmaktadır.

Makinelerin, herhangi bir konuda veya herhangi bir alanda doğal dilde sorulan soruları otomatik olarak nasıl cevaplayacağı yapay zeka uygulamaları için ilgi çeken konulardan biri olmuştur. Bordes ve arkadaşları (2014) yaptıkları çalışmada derin öğrenme teknolojisinden faydalanarak az sayıdaki mühendislik özelliklerini kullanarak bir bilgi tabanından geniş bir yelpazedeki konular üzerine sorular cevaplamaı öğrenen gömülü bir sistem önermişlerdir. 5,810 soru cevap çiftinin bulunduğu WebQuestion veri seti üzerinde eğitilen soru cevaplama sistemi, bu tarz uygulamalar için umut verici bir performans göstermiştir.

2.3.1 Derin öğrenme yöntemi kullanılarak yapılmış yüz tanıma uygulamaları

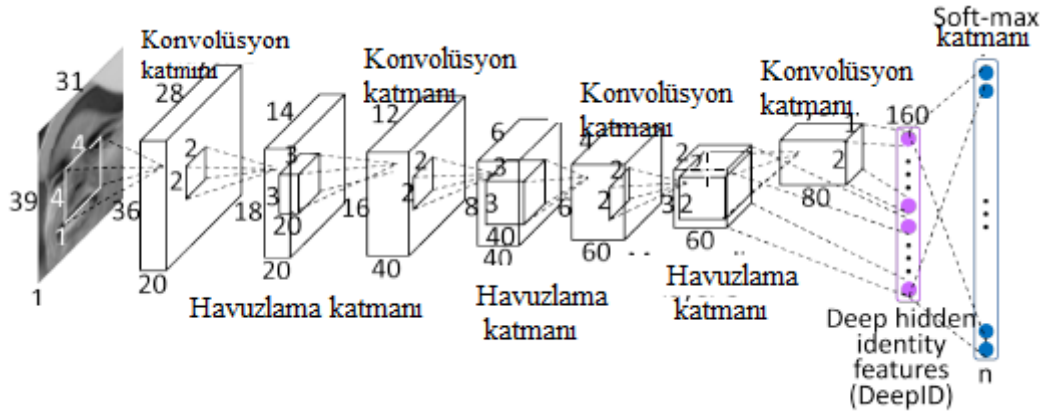
Cheung (2012) tarafından konvolüsyonel sinir ağı kullanılarak yüz sınıflandırma uygulaması geliştirilmiştir. Geliştirilen sistem Avatar CAPTCHA üzerinde denenmiştir

ve % 99 başarı oranı elde edilmiştir. Şekil 2.5’de geliştirilen sinir ağı mimarisi bulunmaktadır.



Şekil 2.5 Kullanılan sinir ağı (Cheung 2012)

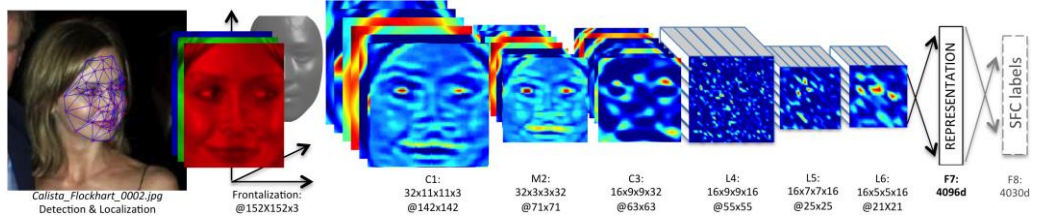
Sun ve Tang (2014) tarafından derin konvolüsyon sinir ağı kullanılarak üst düzey yüz özelliklerini tahmin eden ve bu özellikler yardımıyla bir resmi 10.000 sınıftan bir sınıfa dahil edebilen için bir uygulama geliştirilmiştir. Bu özelliklerin yüz doğrulama ve eğitim setinde görünmeyen yeni yüzlerin bilinmesi için etkili olduğu gösterilmiştir. Ağ, eğitim setinde bulunan tüm yüzleri kimliklerine göre sınıflandırmak üzere eğitilir. Her ağ katmanı, bir yüz yamasını giriş olarak alır. Alt katmanlarda yerel düşük düzey özellikleri ayıklanır, üst katmanlarda giderek daha fazla küresel ve üst düzey özellikler elde edilir. Özellik numaraları, özellik çıkarma süreci boyunca azalmaya devam etmektedir. Özellik çıkarımı sürecinde son katmanda; 160 boyutlu DeepID katmanı elde edilir, bu katman zengin kimlik bilgileri içermektedir ve daha büyük sayıdaki (örneğin, 10 000) kimlik sınıfını doğrudan tahmin edebilmeyi sağlamaktadır.



Şekil 2.6 Kullanılan sinir ağı (Sun ve Tang 2014)

Yapılan çalışmada kullanılan konvolüsyonel sinir ağı şekil 2.6'da gösterilmiştir. Ağ, özelliklerin hiyerarşik olarak çıkarılmasını sağlayan 4 tane konvolüsyonel katman ve havuzlama katmanı kullanılmıştır. Tam bağlı DeepID katmanı ve kimlik sınıflarını belirten softmax çıktı katmanını içermektedir. Geliştirilen sistem LFW veri setinde eğitilmiş ve % 97.45 oranında başarı elde edilmiştir.

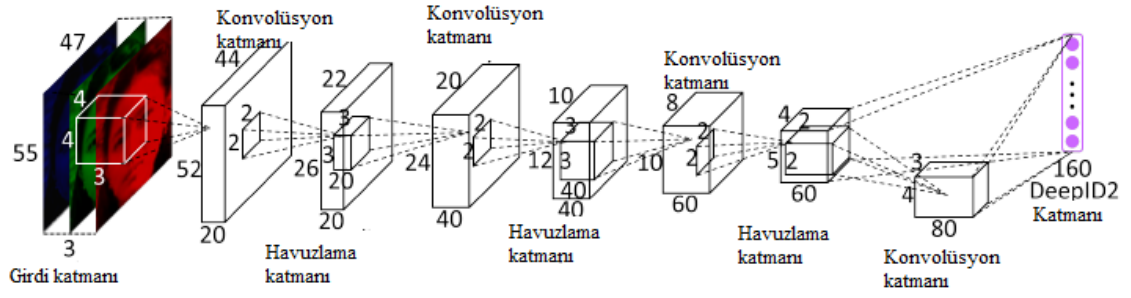
Taigman vd. (2014) kişi doğrulama uygulamalarında insan seviyesindeki performansı yakalayabilmek amacıyla DeepFace ismini verdikleri derin öğrenme teknolojisi kullanarak bir yüz tanıma sistemi geliştirmişlerdir. Derin ağ, standart konvolüsyonel katmanlardan ziyade ağırlık paylaşımı olmaksızın birkaç yerel olarak bağlı katmanı kullanarak 120 milyondan fazla parametre içermektedir. Aşağıdaki şekilde (Şekil 2.7) katmanları gösterilen derin ağ, 4000'den fazla kişiye ait dört milyon yüz görüntünün bulunduğu, bugüne kadarki en büyük yüz verilerinden biri olan LFW üzerinde çok katmanlı bir yüz tanıma görevinde, yani bir yüz görüntüsünün kimliğini sınıflandırmak amacıyla eğitilmiştir. Ağ tarafından öğrenilen özellikler sayesinde, basit bir sınıflandırıcı dahi kullanılsa başarılı sınıflandırma performansı elde edilmiştir. Geliştirilen yöntem ile birlikte % 97,35'lik performans oranına erişilmiştir.



Şekil 2.7 Kullanılan sinir ağı (Taigman vd. 2014)

Schroff vd. (2015) yüz tanıma ve kümeleme uygulamaları için derin konvolüsyonel sinir ağı tabanlı FaceNet adını verdikleri sistemi geliştirmişlerdir. Yüz görüntülerinden, mesafelerin doğrudan yüz benzerliği ölçüsüne karşılık geldiği Öklid uzayına doğrudan bir eşlemeyi öğrenen FaceNet adlı bir sistem önerilmiştir. Öklid uzayı üretildikten sonra, yüz tanıma, doğrulama ve kümeleme gibi görevler, özellik vektörleri olarak FaceNet sistemi ile standart teknikler kullanılarak kolayca uygulanabileceği belirtilmiştir. Geliştirilen sistem LFW ve Youtube Faces veri setlerinde uygulanmış ve sırasıyla % 99.63 ve % 95.12 oranında başarı elde edilmiştir.

Yüz tanıma konusunda geliştirilen çalışmalarda en büyük zorluk, kişiler arası farklılıkları büyütürken, kişisel değişimleri azaltmak için etkili özellikleri geliştirmektir. Sun ve arkadaşları(2014) bu soruna çözüm olabilmesi için yüz doğrulama ve tanıma sistemleri için derin öğrenme tabanlı DeepID2 ismini verdikleri öznelik çıkarıcı bir sistem geliştirmişlerdir. Yüz doğrulama görevi, aynı kimlikten çıkarılan DeepID2 özelliklerini bir araya getirerek kişisel tanımlamaları azaltırken, yüz tanıma görevi DeepID2 özelliklerini birbirinden ayırarak kişisel farklarını artırmaktadır. DeepID2 öznelik çıkarımı için derin öğrenme mimarisi şekil 2.8’de gösterilmiştir.

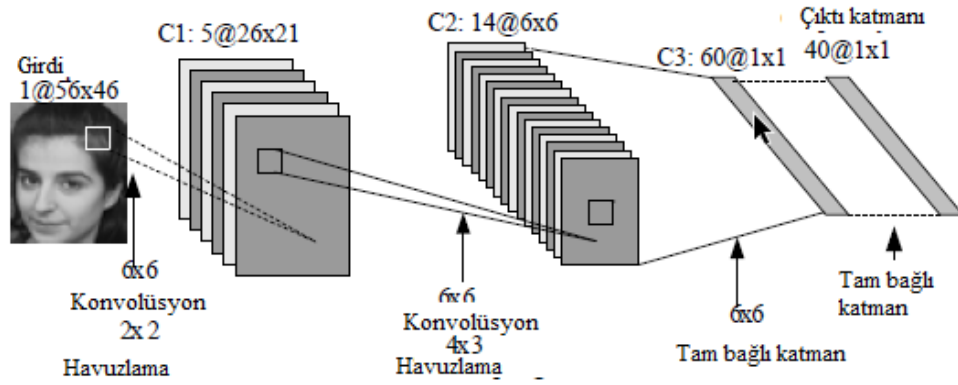


Şekil 2.8 Kullanılan sinir ağı (Sun vd. 2014)

Geliştirilen sistem, LFW veri setinde yüz tanıma uygulaması için uygulanmış ve % 99.15 başarı elde edilmiştir.

Hu vd. (2015) derin öğrenme temelli ağların performansının neden yüksek olduğunun ve iyi bir ağ mimarisinin nasıl tasarlanması gerekir sorularına cevap bulabilmek için araştırma yapmışlardır. Bu kapsamda, farklı sayıda filtre ve katman içeren KSA küçük, orta ve büyük olarak isimlendirilen 3 farklı mimari tasarlanmış ve ayrı ayrı LFW veri seti üzerinde denenmiştir. Çalışmanın sonucunda farklı ağ mimarilerinin farklı bölgelerdeki bilgileri ve ölçekleri güçlü bir yüz temsili oluşturmak için yakaladığından, ağ füzyonunun yüz tanıma performansını önemli ölçüde artırabileceğini gösterilmiş ve farklı KSA katmanlarından özelliklerin bir araya getirilmesinin yüz tanıma performansını artırabileceğini gözlemlenmiştir.

Syafeeza (2016) MATLAB tabanlı Konvolüsyonel Sinir ağı kullanarak yüz tanıma sistemi geliştirmiştir. Görüntü ön işleme aşamaları MATLAB üzerinde uygulanmış, Konvlüsyonel ağ algoritması ise LINUX tabanlı C dilinde geliştirilmiştir. Şekil 2.9'da geliştirilen yüz tanıma sisteminde kullanılan ağ mimarisi gösterilmiştir.



Şekil 2.9 Kullanılan sinir ağı (Syafeeza 2016)

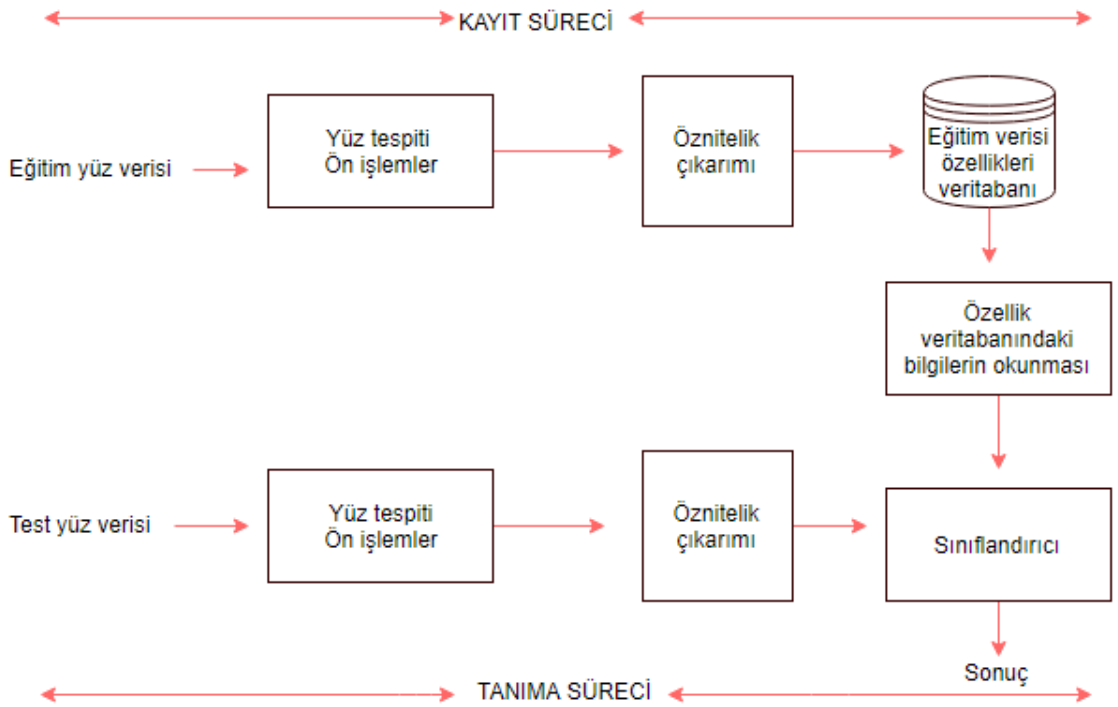
Kullanılan 3 konvolüsyon katmanı ve 1 tam bağlı katmandan oluşmaktadır. Konvolüsyon katmanlarında sırasıyla; 5, 14 ve 60 özellik haritası bulunmaktadır. Ağ ORL veri seti kullanılarak eğitilmiştir. Eğitmeden kasıt; her katmandaki birbirine bağlı optimal ağırlık değerlerinin elde edilmesidir. 1.katmanda 3.katmana kadar olan optimum ağırlık öznelik çıkarıcı gibi hareket eder ve sabitlenir. Sisteme yeni bir resim girdiğinde, 3. Katmandan çıktı katmanına kadar olan kısım için rastgele ağırlık değeri ataması başlatılır. Kullanılan veri setinde 40 farklı kişinin sınıflandırılması yapılacağı için 40 tane özellik haritası vardır. Geliştirilen yüz tanıma sistemi AT&T ve JAFFE veri setinde denenmiş ve % 100 tanıma oranı elde edilmiştir.

3. GELİŞTİRİLEN YÖNTEM

Bu bölümde; yüz tanıma çalışmalarının genel mimarisi, tez kapsamında geliştirilen algoritma ve kullanılan yöntemler detaylı bir biçimde anlatılmıştır.

3.1 Yüz Tanıma Çalışmalarında Gerçekleştirilen Temel Konsept

Bir yüz tanıma sistemi, kayıt aşaması ve tanıma aşaması olmak üzere iki bölümden oluşur. Kayıt sürecinde; sistem binlerce yüz görüntüsü kullanılarak eğitilir ve eğitilmiş bir model oluşturulur. Üretilen yüz özellikleri veritabanında saklanır. Tanıma sürecinde; test yüz görüntüsü kayıt sürecinde üretilen modele girdi olarak verilir. Test görüntüsünden çıkarılan yüz özellikleri, veri tabanında saklanan yüz özellikleri ile karşılaştırılarak tanıma işlemi gerçekleştirilir. Şekil 3.1’de yüz tanıma algoritması işleyiş adımları gösterilmiştir.



Şekil 3.1 Yüz tanıma algoritması

3.2 Yüz Tanıma Aşamaları ve Kullanılan Yöntemler

Tez kapsamında geliştirilen çalışmanın amacı; geleneksel makine öğrenme tekniği ve derin öğrenme tekniği kullanılarak bir yüz tanıma algoritması eğitmek ve iki ayrı yöntemin başarı oranlarını karşılaştırmaktır. Bunun için yüz tanıma algoritması kurulmuştur. Algoritma, yüz tespiti, ön işlemler, öznitelik çıkarımı ve sınıflandırma olmak üzere 4 ana kısımdan oluşmaktadır.

Bu kapsamda gerçekleştirilen çalışmada şekil 3.2’de gösterilen akış şeması ve belirtilen yöntemler kullanılmıştır.



Şekil 3.2 Yüz tanıma akış şeması

Akış şemasında belirtilen yöntemlerin detaylı açıklamaları başlıklar halinde anlatılacaktır.

3.3 Viola&Jones Algoritması

Yaygın kullanılan gerçek zamanlı obje tespit metodudur.

Avantajları:

- Çok yüksek obje tespit etme oranına sahiptir.
- Görüntünün kendisini ölçeklendirmek yerine, özellikleri ölçeklendirir.

Dezavantajları:

- Dedektör, yalnızca yüzlerin ön görüntülerinde etkilidir.

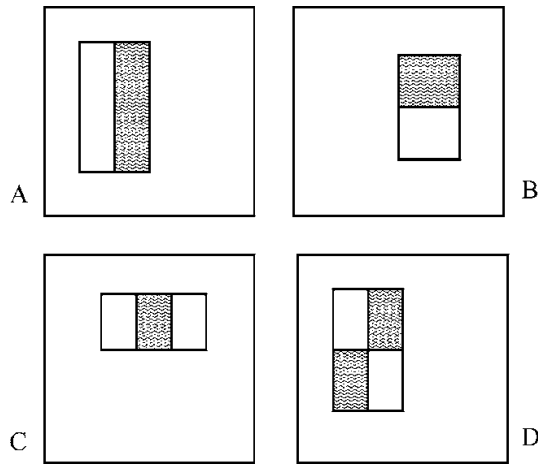
- Dikey ve yatay eksen çevresinde 45 derece yüz rotasyonu ile pek başa çıkamaz.
- Işık koşullarına duyarlıdır.
- Aynı yüz üzerinde birden çok algılama yapabilir.

Algoritma 4 aşamadan oluşmaktadır.

- Haar özellikleri
- Integral görüntü
- Adaboost eğitimi
- Cascading sınıflandırıcı (Viola ve Jones 2004)

3.3.1 Haar özellikleri

Çalışılan şekildeki yüzü bulmak için kullanılan zayıf sınıflandırıcılardır. İki dikdörtgenli, üç dikdörtgenli, dört dikdörtgenli olmak üzere üç tip özellik tanımlamaktadırlar. Aşağıdaki şekilde (Şekil 3.3) görüleceği üzere bu özellikler detectorün penceresi içinde herhangi bir yerde tanımlanabilirler. Tek başlarına ifade edebilecekleri bilgi ve başarımlar çok az olduğundan zayıf sınıflandırıcı (weakclassifier) olarak adlandırılırlar.



Şekil 3.3 Haar özellikleri (Viola ve Jones 2004)

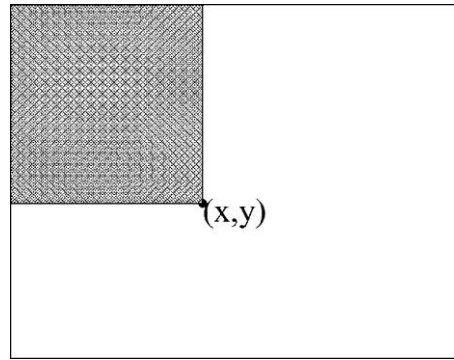
$$\sum_{1 \leq i \leq N} \sum_{1 \leq j \leq N} I(i, j) 1_{P(i, j) \text{ Beyaz}} - \sum_{1 \leq i \leq N} \sum_{1 \leq j \leq N} I(i, j) 1_{P(i, j) \text{ Siyah}} \quad (3.1)$$

Yukarıda belirtilen formül 3.1'den anlaşılacağı gibi dikdörtgenlerin beyaz kısımlarındaki tüm piksel değerleri toplamından siyah kısımlarındaki tüm piksel değerleri toplamı çıkarılır. Elde edilen değer belli bir eşik değerinden yüksekse orada yüz vardır, değilse yüz yoktur demektir (Wang 2014).

3.3.2 Integral görüntü

Yüz bulma işleminin gerçek zamanlı olarak çalışabilmesi için çok az işlem yaparak hızlıca piksellerin toplamını bulabilmesi gerekmektedir. Bu durumda integral görüntü kavramı devreye girmektedir. Haarözelliklerin hızlı çalışmasını sağlar. Aşağıdaki şekilde (Şekil 3.4) belirtildiği üzere, her pikselin sol ve üstünde kalan piksellerin toplanması ile elde edilmektedir.

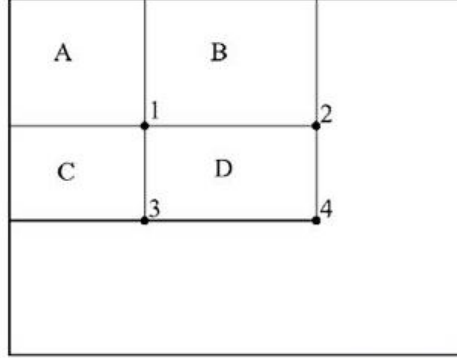
$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (3.2)$$



Şekil 3.4 İntegral görüntü gösterimi (Viola ve Jones 2004)

Şekil 3.5' de integral görüntünün hesaplanması basitleştirilmiş bir şekilde anlatılmaktadır. Oluşturulan integral görüntüdeki her bir piksel, kendisinin sol ve yukarısındaki piksellerin toplamına eşit olmaktadır. Örneğin 1 noktasındaki integral görüntünün

değeri, A dikdörtgeni içindeki piksel değerlerinin toplamına eşittir. 2 konumundaki değer $A+B$, 3 konumundaki değer $A+C$, 4 konumundaki değer $A+B+C+D$ 'dir. D içindeki piksel değerleri toplamı $4+1-(2+3)$ olarak hesaplanır (Viola ve Jones 2004).



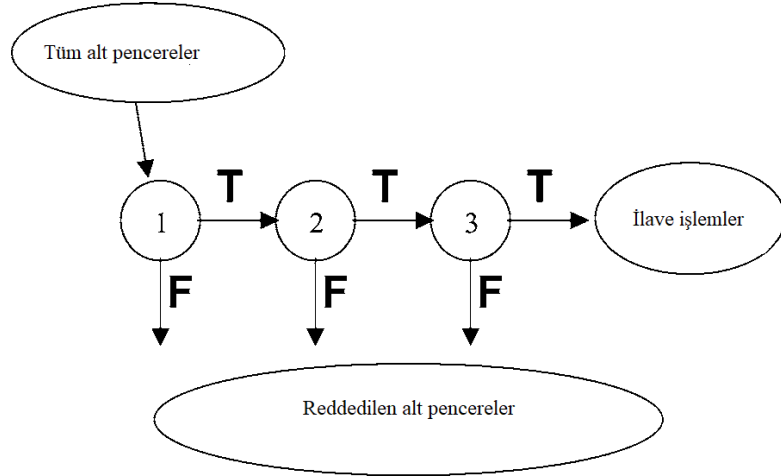
Şekil 3.5 İntegral görüntü hesaplanması (Viola ve Jones 2004)

3.3.3 Adaboost eğitimi

Adaboosttraining' in çalışma mantığı; anlamlı özelliklerin seçilip, gereksiz özelliklerin elenmesi esasına dayanmaktadır. Gereksiz özellikler azaltılarak, doğrusal kombinasyon ile birlikte güçlü bir sınıflandırıcı elde edilir (Wang 2014).

3.3.4 Cascading sınıflandırıcı

Cascadingclassifier çalışma mantığı; detektörün bir defada tüm sınıflandırıcıları uygulamak yerine kademeli bir şekilde uygulamasıdır. Bu şekilde yüz ihtimali olmayan yerler direk ihmal edilir. Bu tarz bir yaklaşım sayesinde, tüm sınıflandırıcılar uygulanmadan daha ilk aşamada birçok alt sınıf elimine olduğu için aşırı bir hız kazanılmış olur.



Şekil 3.6 Cascading classifier çalışma prensibi (Viola ve Jones 2004)

3.4 Histogram of Gradients Algoritması

Histogram of gradients yöntemi Ölçeksiz Değişken Özellik Dönüşümü (Scale Invariant Feature Transform, SIFT) yönteminden geliştirilmiştir. SIFT, bir nesnenin veya sahnenin farklı görünümünü eşleştirmek gibi görevler için algoritmalarda kullanılmak üzere, görüntülerden ayırt edici özellikleri çıkarmak için bir uç teknoloji olarak ortaya çıkmıştır. SIFT, görüntünün ölçek-alan gösteriminin yerel ekstremumunda kilit noktalarını çıkartarak ölçek değişikliklerine değişmezlik kazandırır, ardından her bir kilit noktası, değişkenlerin histogramları kullanılarak ifade edilir (Albiol vd. 2008).

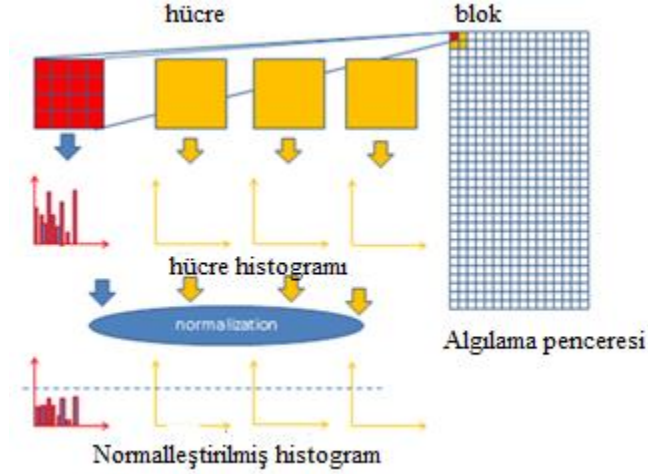
HOG tanımlayıcıları genel olarak nesne tanımda, yaya tespiti ve özellikle yüz tanıma gibi uygulamalarda etkili bir tanımlayıcı olarak kullanılmaktadır (Denz vd. 2011). Bu yaklaşımlarda, nesnelerin sabit bir ölçekte olduğu ve sabit konumlarda küçük bağlı bölgelere bölünmüş olduğu varsayılmaktadır. Ardından, her bölge için bir HOG tanımlayıcı elde edilir ve bu tanımlayıcıların kombinasyonu nesneyi temsil etmek için kullanılır (Albiol vd. 2008).

Özellik tanımlayıcılar, görüntüdeki yararlı bilgileri seçip gereksiz olanları eleyerek görüntüyü basitleştirir. Sık kullanılan ve başarılı sonuçlar veren özellik tanımlayıcılardan bir tanesi histogram of oriented gradients algoritmasıdır.

Histogram of oriented gradients (HOG) görüntü işleme alanında nesnelere tanımak için kullanılan bir öznelik çıkartıcısıdır (özellik tanımlayıcı). HOG tanımlayıcı tekniği bir resmin küçük parçalarında karşılaşılan eğim yönlendirmelerini sayar. Özellik tanımlayıcılar, görüntüdeki yararlı bilgileri seçip gereksiz olanları eleyerek görüntüyü basitleştirir.

HOG algoritması aşağıdaki adımlar ile gerçekleştirilir

- Resim hücre adı verilen küçük bloklara bölünür ve her hücre için eğim yönlerinin histogramı hesaplanır.
- Her hücreyi eğim yönlendirmelerine göre açısız bloklara ayrıştırır.
- Hücrenin her pikseli için ağırlıklı eğim hesaplanır ve bunların her biri açısız blokları etkiler.
- Komşu hücreler bloklar olarak gruplanır ve bu sayede histogram normalize edilir.
- Normalize edilmiş histogramlar grubu blok histogramı temsil eder. Bu blok histogramlar kümesi ise HOG tanımlayıcısını temsil eder.

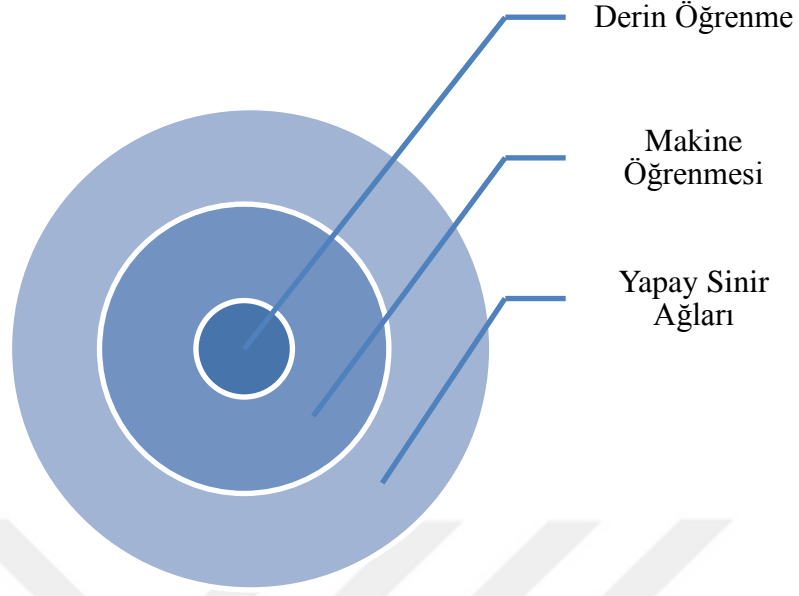


Şekil 3.7 HOG algoritması çalışma şeması (Anonymous 2017a)

3.5 Derin öğrenme ve Çalışma Prensipleri

Derin öğrenme, yapay zeka temelli bir dizi istatistiksel makine öğrenme tekniğidir. Ham veri kaynaklarından yüksek düzeyde gösterimler, simgeler üreten nispeten yakın zamanda geliştirilmiş makine öğrenme tekniklerinden biridir. Videoları, konuşmaları, sesleri, görüntüleri işleme, görsel nesne tanıma, sınıflandırma, regresyon ve kümeleme gibi birçok alanda başarılı sonuçlar üretmektedir. Özellik hiyerarşisinin otomatik olarak öğrenilmesini sağlayan çok katmanlı yapay sinir ağıdır. Çalışma prensibi, her bir ardışık katmanın daha karmaşık özellikleri öğrenebilmek için bir önceki katmandaki özellikleri kullanması esasına dayanmaktadır. Bu süreç, çıktı (örneğin bir tahmin ya da sınıflandırma) üretilinceye kadar bir dizi katman için devam etmektedir.

Bugün derin öğrenme ile eğitilen algoritmalar ile görüntü tanımadan, kandaki kanser belirtilerinin tespiti, MR taramalarında tümör tespitine kadar birçok konuda insandan daha başarılı sonuçlar elde edilmektedir. İnternet sayesinde devasa boyutlara ulaşan veri miktarını başarılı bir biçimde kullanabilmesi, grafik işleme birimi (GPU) ile sağlanan işlem gücündeki artış ve ucuzlamadan yararlanabilmesi ve işlem gücünün artmasının sonucunda daha derin modelleri eğitebilmesi nedeniyle son yıllarda yüz tanıma alanında da derin öğrenme güçlü duruma gelmiştir.



Şekil 3.8 Kavramlar arası ilişki

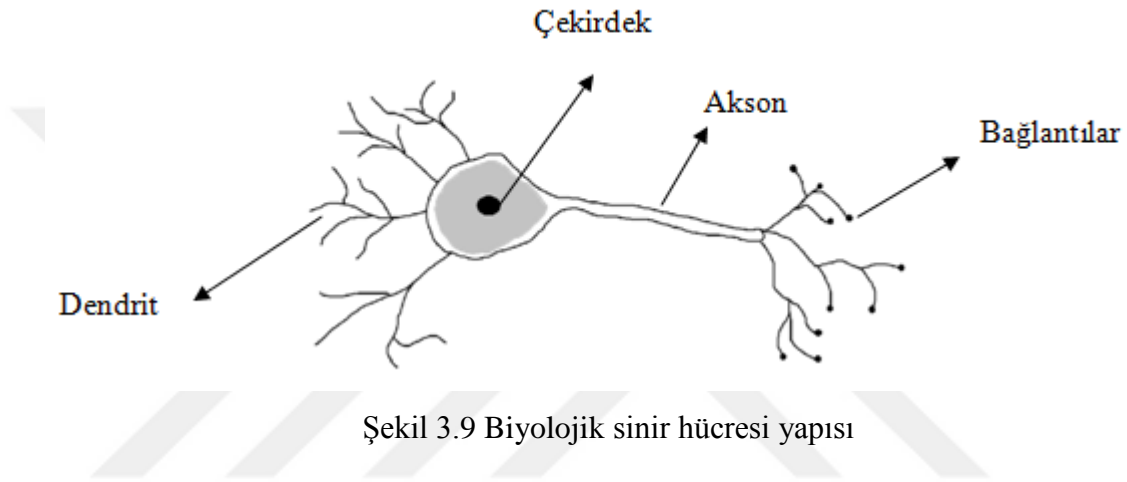
Şekil 3.8'den anlaşılacağı gibi yapay zeka, makine öğrenmesi ve derin öğrenme kavramları birbirleriyle birebir aynı kavramlar olmamakla birlikte, iç içe geçmiş ve birbirleri ile ilişkili kavramlardır.

Yapay sinir ağları; bilgisayarların örnekler üzerinden olaylar arasındaki ilişkileri öğrenerek, daha önce hiç karşılaşılmamış yeni örnekler ile karşılaşıldığında, öğrenilen bilgiler kullanılarak karar verme gücünden faydalanılmasını sağlayan Yapay zeka bilimidir. Yapay zeka, normalde insan zekası gerektiren görsel algılama, konuşma tanıma, karar verme ve diller arasında çeviri gibi görevleri yerine getirebilen bilgisayar sistemlerinin teorisi ve geliştirilmesi ile ilgilenen bilgisayar bilimlerinin bir dalıdır. YSA da başarılı eğitim, bir çıktı değerinin tahmin edilmesi, bir nesnenin sınıflandırılması, bir fonksiyonun öğrenilmesi, çok faktörlü bir veride bir örüntünün tanınması ve bilinen bir kalıbın tamamlanması gibi görevleri yerine getirilmesi ile sonuçlanabilir.

Literatürde YSA' ların birçok farklı uygulamaları bulunmaktadır. Otomatik hedef tanıma, uçan uçakları kontrol etme, motor yanma optimizasyonu, karmaşık mühendislik sistemlerinde arıza tespiti gibi askeri uygulamalar kullanılmıştır. Tıbbi görüntü analizi

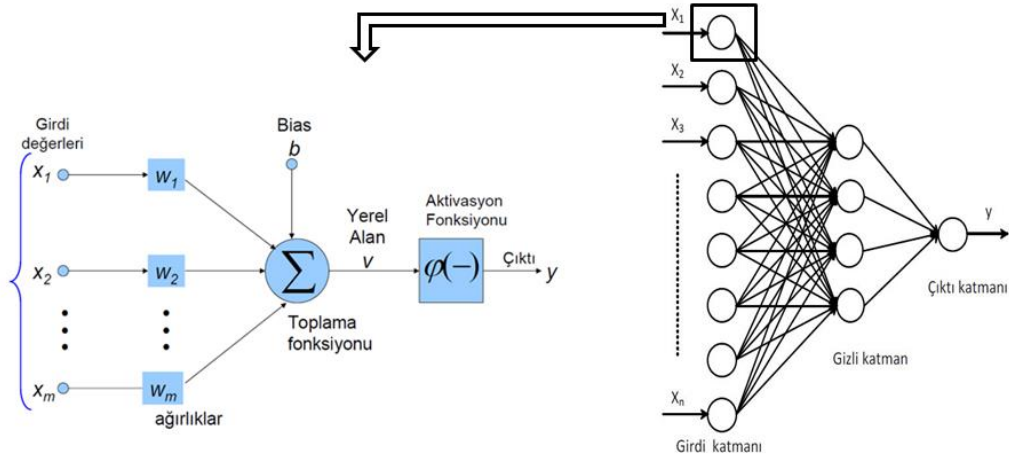
ile tıbbi görüntülerde tümörleri saptayan sistemlerin kurulması ve sitolojik testlerde normal hücelere karşı malign sınıflandıran sistemlerin kurulması gibi tıp alanındaki uygulamalarda kullanılmıştır. Zaman serisi tahminleri, düzensiz ve kaotik dizilerin tahminini de içeren sistemlerde kullanılmıştır (Dayhoff ve DeLeo 2001).

Yapay sinir ağları biyolojik nöron ağlarından esinlenerek geliştirilmiştir. Şekil 3.9)'da biyolojik sinir hücresi yapısı gösterilmiştir.



Şekil 3.9 Biyolojik sinir hücresi yapısı

Biyolojik Sinir hücreleri; işledikleri bilgileri aksonları yardımı ile diğer hücelere gönderirler. İki sinir ucunu birbirine bağlayan bağlantılar (synapse) aracılığı ile birbirleriyle iletişim kurarlar. Bir sinir hücresine duyu organları ile gelen bilgiler işlendikten sonra akson (axon) yardımı ile diğer sinir hücrelerine gönderilir. Bu sistemden esinlenerek geliştirilen YSA ların çalışma prensibi matematiksel olarak şekil 3.10'da gösterilmektedir. Dışarıdan gelen bilgiler bir toplama fonksiyonu ile ağırlık değerleri ile çapılarak toplanır bu sayede ağın net giriş bilgisi hesaplanır. Bu net girdi, genellikle doğrusal olmayan bir aktivasyon fonksiyonu (lineer, step, sigmoid vb.) ile analiz edilerek ağın çıkış bilgisi belirlenir ve bu bilgi ağ bağlantıları üzerinden diğer hücelere gönderilir (Öztemel 2003).



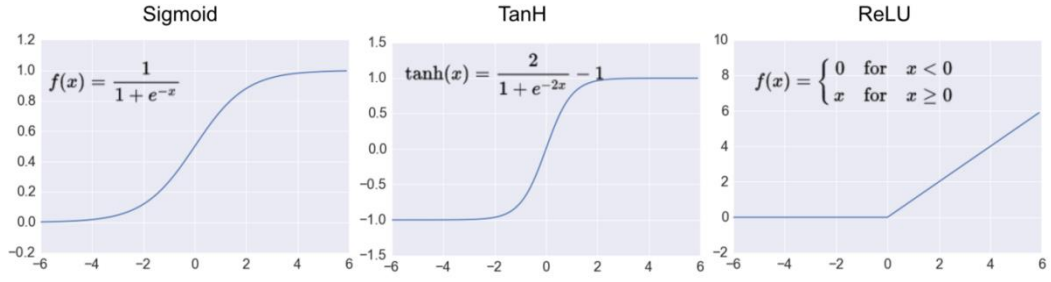
Şekil 3.10 Yapay sinir hücresi modeli (Anonymous 2017b)

Farklı ağırlıktaki farklı girdilerin toplamı (Denklem 3.3), aktivasyon fonksiyonuna verilmekte (Denklem 3.4) ve sonucunda nöronun çıktısı elde edilmektedir. Bias değeri sinir ağı modelinde mevcut olabilir veya olmayabilir.

Veriler eşzamanlı olarak girdi katmanından ağa verilir ve belirli bir ağırlık değeri ile çarpılarak gizli katmanlarda işlenir ve çıktı katmanına gönderilir. Ağa verilen giriş değerlere karşılık doğru çıkış değerlerinin üretilmesi doğru ağırlıkların bulunmasına bağlıdır bu da ağın eğitilmesi demektir. Ağırlık değerleri önce rastgele atanır, eğitim sırasında ağın örnekleri öğrenme kuralına göre değiştirilir. Ağırlık değerleri, eğitim sırasında bütün örnekler için doğru çıkış değerleri bulununcaya kadar devam eder. Test kümesinde bulunan veriler için, ağ doğru cevap verdiğinde ağ için eğitim işlemi tamamlanmış olmaktadır. (Öztemel 2003)

$$f(x) = \sum_{i=1}^N (w_i \cdot x_i) \quad (3.3)$$

$$y = \phi(f(x)) \quad (3.4)$$



Şekil 3.11 Aktivasyon fonksiyonları (Anonymous 2017c)

Yukarıda şekil 3.11’de belirtildiği gibi tanh, sigmoid ve ReLU aktivasyon fonksiyonları kullanılabilir.

Makine öğrenmesi; mevcut verileri ayrıştırarak çıkarım yapan, yaptığı çıkarım ile tahminde bulunan ya da karar verebilen algoritmaların geliştirilmesini konu alan bir bilim dalıdır. En basit biçimde Makine Öğrenmesi, verilerin ayrıştırılması için algoritma kullanma, onu öğrenme ve sonra dünyadaki bir şey hakkında bir belirleme veya tahmin yapma uygulamasıdır. Belirli bir görevi yerine getirmek için belirli talimatlar içeren yazılım rutinlerini elle kodlamaktan ziyade, makine, görevi nasıl yerine getireceğini öğrenme yeteneği kazandıran büyük miktarlarda veri ve algoritmalar kullanarak "eğitilir".

Geleneksel makine öğrenme teknikleri ile ilgili en büyük problemlerden biri öznitelik çıkarımı sürecidir. Veriyi ham biçimde işleyebilme yeteneklerinde sınırlı oluşu için sınıflandırma ya da karar verme aşamasında bilgilendirici ve kritik olan özelliklerin hangileri olması gerektiğinin bilgisayara programcı tarafından belirtilmesi gerekmektedir. Bu durum programcıya büyük bir yük getirmektedir çünkü algoritmanın etkinliği, hangi özelliklerinin çıkarılacağına belirlenmesine bağlıdır ve nesne tanıma gibi karmaşık uygulamalarda bu durum büyük bir sorun oluşturmaktadır. Kendiliğinden doğru özelliklere odaklanmayı öğrenme yeteneğine sahip olduğu ve bu yüzden programcıdan çok az rehberlik gerektiren derin öğrenme modelleri öznitelik sürecindeki sorunlara çözüm olmuştur. Doğru özellikleri kendiliğinden odaklanabilmesi derin öğrenmeyi, modern makine öğrenimi için son derece güçlü bir araç haline getirmiştir (Bengio vd.2013).

Makine öğrenme teknolojisi, web aramalarından sosyal paylaşım ağlarındaki içerik filtrelemesine, e-ticaret web sitelerindeki önerilere ve kameralar ve akıllı telefonlar gibi tüketici ürünlerinde, giderek artan bir biçimde modern toplumun birçok yönünü güçlendirmektedir. Makine öğrenme sistemleri, görüntüdeki nesnelere tanımlamak, konuşmayı metin haline getirmek, haber öğelerini, kullanıcıların ilgi alanına giren yayınları veya ürünleri eşleştirmek ve alakalı arama sonuçlarını seçmek için kullanılır. Bu uygulamalar, derin öğrenme adı verilen bir grup tekniğin kullanımından faydalanmaktadır (LeCun vd. 2015).

Temsili öğrenme (representation learning), bir makineye girdi olarak ham verilerin verilmesine ve tespit veya sınıflandırma için gerekli olan sembollerin, temsillerin otomatik olarak öğrenmesine olanak tanıyan bir dizi yöntemdir. Derin öğrenme yöntemleri, her bir gösterimi tek bir seviyede (ham girişten başlayarak) daha yüksek, biraz daha soyut seviyede bir gösterime dönüştüren basit ama doğrusal olmayan modüller oluşturarak elde edilen birden fazla gösterim düzeyine sahip temsili-öğrenme yöntemleridir. Özellikleri dönüştürerek öğrenme yetenekleri sayesinde çok karmaşık fonksiyonlar öğrenilebilirler.

Derin öğrenme ile geliştirilen uygulamanın sınıflandırma sürecinde; çok katmanlı yapısı yani yüksek katmanlarının olması ağa verilen girdinin ayırt edici özelliklerini güçlendirir ve alakasız olan özellikleri bastırır. Örneğin, bir görüntü, bir dizi piksel değeri biçiminde gelir ve ilk ağ katmanındaki öğrenilen özellikler, tipik olarak, görüntüdeki belirli yönlerde ve konumlarda kenarların varlığını veya yokluğunu temsil eder. İkinci katman, kenar konumlarındaki küçük değişikliklere bakılmaksızın, kenarların özel düzenlemelerini tespit ederek motifleri algılar. Üçüncü katman, motifleri tanıdık nesnelere parçalarına karşılık gelen daha büyük kombinasyonlara toplayabilir ve sonraki katmanlar nesnelere bu parçaların kombinasyonları olarak algılayabilir. Özetle ağ, ham veriden kenarları öğrenir, öğrendiği kenarları kullanarak şekilleri öğrenir, şekilleri kullanarak daha üst düzey özellikleri öğrenir. Bu hiyerarşik yapı derin öğrenmeyi güçlü yapmaktadır. Derin öğrenmeyi öne çıkaran temel unsuru, bu özellik katmanlarının programlayıcı tarafından tasarlanmamasıdır (LeCun vd. 2015).

Görsel algılama, işitsel algılama ve dil anlama gibi akıllı görevler, varyasyonlara karşı değişmeyen, ilgili özelliklerin başarılı bir temsilini gerektirmektedir. Bu gerekli özelliklerin otomatik olarak nasıl öğrenilebileceği Makine öğrenimi alanında önemli bir sorudur. Derin öğrenme, özelinde Konvolüsyon sinir ağları değişmeyen özellikleri öğrenebilen, biyolojik sinir hücresinden etkilenecek tasarlanmış, eğitilebilir çok katmanlı bir mimardır. Ağ genel olarak, filtrelerden oluşan konvolüsyon katmanı, özellik havuz katmanı ve tam bağlı katmandan oluşmaktadır. Çok katmanlı mimari sayesinde, KSA uçtan uca eğitimle özellikleri hiyerarşik olarak öğrenir (LeCun vd. 2010, Farabet vd. 2013).

Derin öğrenme çok katmanlı yapay sinir ağıdır ve çalışma prensibi en temel hali ile yapay sinir ağları çalışma prensibine benzemektedir.

Herbir ardışık katman daha karmaşık özellikleri öğrenebilmek için bir önceki katmandaki özellikleri kullanır. Derin öğrenme, ileri besleme ile (feedforward network) görüntüden özellikleri çıkarır. Geri besleme ile (backpropagation) ise öğrenme yapılır.

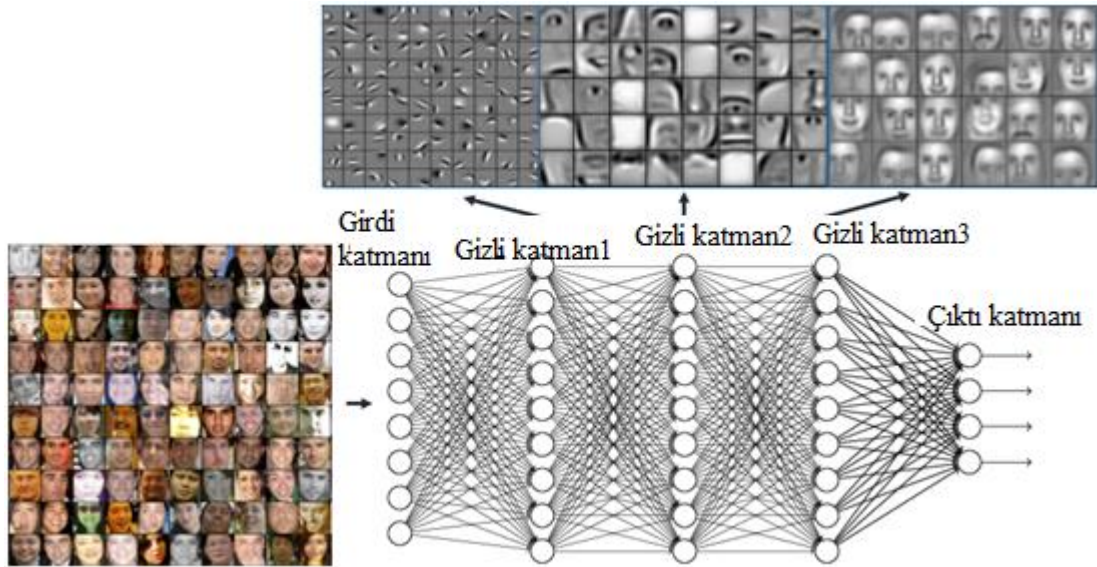
Bir derin öğrenme türü olan Konvolüsyon sinir ağının eğitim süreci özetle şu şekildedir: Ağın eğitimi öncesinde; filtre, parametre ve ağırlık değerleri rastgele atanır. Bu değerler eğitim süreci boyunca değişmez. Eğitim verileri setinde bulunan görüntüler ağa girdi olarak verilir. Girdi, ileri besleme aşaması (konvolüsyon, Relu, havuzlama ve tam bağlı katmanlardan ilerleyerek)ile çıktı katmanına gelir ve her bir sınıf için çıktı olasılık değeri hesaplanır. Ağırlık değeri rastgele atandığı için, çıktı değeri de rastgele olacaktır. Çıktı katmanında aşağıda belirtilen denklem 3.5 kullanılarak toplam hata hesaplanır.

$$\text{Toplam Hata} = \sum \frac{1}{2} (\text{Hedef olasılığı} - \text{Çıktı olasılığı})^2 \quad (3.5)$$

Ağ eğitimi için kullanılan popüler algoritmalarından birisi geri yayılım algoritmasıdır. Hataları, çıktı katmanından girdi katmanına doğru geri yönde giderek azaltmaya çalıştığı için bu şekilde adlandırılmıştır. Sınıflandırma işlemi hata oranını azaltmak için ağırlık değerlerinin güncellenmesi işlemidir. Ağ bu şekilde eğitilmektedir. Geri yayılım

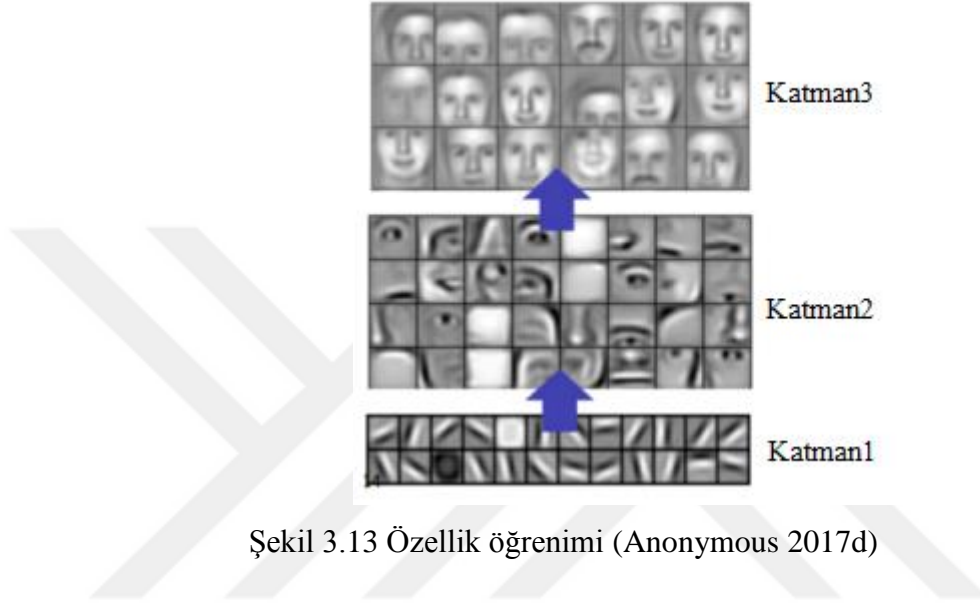
işlevi kullanılarak, ağıdaki tüm ağırlıklara göre hata eğimleri hesaplanır ve çıktı hatasının en aza indirilmesi için bütün filtre ağırlık ve parametre değerleri gradient descent yöntemi kullanılarak güncellenir (Keleşoğlu ve Fırat 2006). Bu algoritma kullanılarak, ağıdaki tüm ağırlıklara göre hata eğimleri hesaplanır ve çıktı hatasının en aza indirilmesi için bütün filtre ağırlık ve parametre değerleri gradient descent yöntemi toplam hata değerine olan katkılarına orantılı olarak kullanılarak güncellenir. Bu işlem eğitim veri setinde bulunan bütün değerler için uygulanır. Ağı sahip olduğu ağırlık ve parametre değerlerinin, eğitim setinde bulunan bütün görüntüleri doğru olarak sınıflandırabilmesi için optimize edilmesi, ağı eğitilmesi demektir (Anonymous 2017d).

Öğrenme aşamasında ağı, değerleri filtreleri sayesinde öğrenmektedir. Ağı, ham veriden kenarları öğrenir, öğrendiği kenarları kullanarak şekilleri öğrenir, şekilleri kullanarak daha üst düzey özellikleri öğrenir. En son katmanda ise sınıflandırma işlemini gerçekleştirmektedir. Şekil 3.12’de şekilsel olarak gösterilen bu hiyerarşik yapı nesne tanımda derin öğrenme algoritmasını güçlü yapmaktadır.



Şekil 3.12 Derin öğrenme mimarisi (Anonymous 2017e)

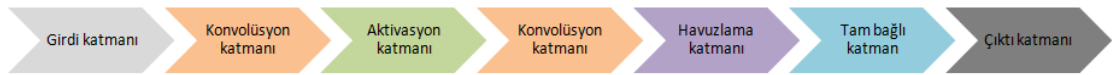
Derin öğrenme algoritması girilen örnekleri katmanlara ayırır; bununla birlikte en düşük katmandan başlayarak örüntüyü tanımaya çalışır ve bir prototip oluşturur. Şekil 3.13’ de gösterildiği gibi ağ, 1.katmanda kenarları öğrenir, öğrendiği kenarları kullanarak ikinci katmanda şekilleri öğrenir, şekilleri kullanarak 3.katmanda daha üst düzey özellikleri öğrenir.



Şekil 3.13 Özellik öğrenimi (Anonymous 2017d)

3.5.1 Derin öğrenme mimarisi

Konvolüsyon sinir ağları mimarisi genellikle girdi katmanını takip eden konvolüsyon katmanı (convolution layer), aktivasyon katmanı (ReLU), havuzlama katmanı (poolinglayer) ve tam bağlı katmanından (fully connected layer) oluşmaktadır.



Şekil 3.14 Konvolüsyonel sinir ağı mimarisi örneği

Konvolüsyon katmanı orijinal resmi, orijinal piksel değerlerinden final sınıf sonucuna katman katman dönüştürmektedir. Yalnızca konvolüsyon ve tam bağlı katmanlar ağırlık değerlerine sahiptir ve bu ağırlıklar eğitim aşamasında öğrenilmektedir. Konvolüsyon

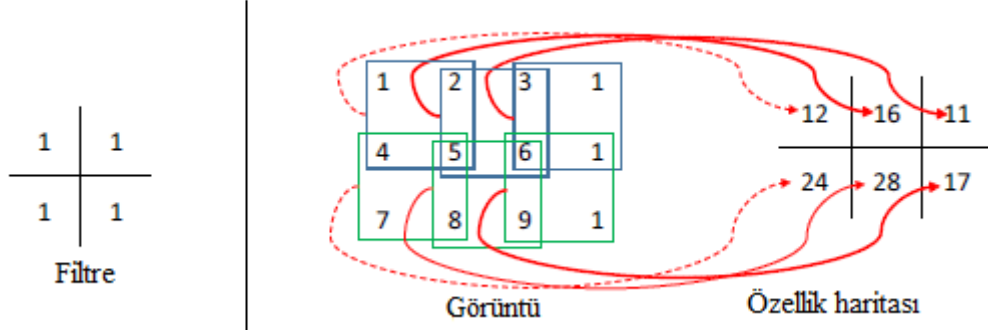
ve tam bağı katmanlarında parametrelerin ağırlık ve bias değerlerinde ve girdilerin aktivasyon fonksiyonlarında değişim gerçekleşmektedir. Öte yandan, aktivasyon ve havuzlama katmanları, sabit bir fonksiyon uygulamaktadır.

Konvolüsyon, havuzlama ve ReLU katmanları, öğrenilebilir özellik çıkarıcı olarak görev yaparken, tam olarak bağı katmanlar, bir makine öğrenme sınıflandırıcısı gibi davranmaktadır. Ayrıca, ağı önceki katmanları görüntülerin genel kalıplarını kodlarken, daha sonraki katmanlar görüntülerin ayrıntı modellerini kodlar. Konvolüsyon ve tam bağı katmanlar parametrelere sahiptir ancak aktivasyon ve havuzlama katmanlarının parametresi yoktur.

Konvolüsyon, tam bağı ve havuzlama katmanı ek hiper parametrelere (derinlik, filtre sayısı, filtre boyutu, adım, dolgu sayısı) sahipken aktivasyon katmanı değildir (Wu 2016, Anonymous 2017f).

Konvolüsyon katmanı konvolüsyon sinir ağının temelini oluşturmaktadır ve başlıca görevi girdi görüntüsünün özelliklerini çıkarmaktır. Konvolüsyon katmanında; her bir filtre (kernel ya da nöron) girdi üzerindeki farklı bir özelliği tespit etmektedir. Her bir konvolüsyon katmanında farklı bir filtre uygulanmaktadır ve bu sonuçlar özellik çıkarımı için birleştirilmektedir. Bir filtrenin sonucu, bir üst katmandaki filtrenin girdisidir. Alt katmandaki özellikleri içermektedir. Belirli boyutta filtre resim üzerinde gezdirilerek özellik öğrenimi yapılmaktadır. Öğrenme aşamasında ağı, değerleri filtreleri sayesinde öğrenmektedir. Orijinal resim piksel değeri ile filtre değerleri çarpılıp toplanarak tek bir değer elde edilir. Bu işlem girdi hacmindeki her konum için tekrarlanmaktadır. En sonunda elde edilen değerler sayı dizisine özellik haritası (activation map, feature map) denilmektedir. Her filtre, farklı özellik haritası oluşturmaktadır yani görüntünün farklı özelliklerini çıkarmaktadır. İlk katmandaki filtreler, girdi görüntüsünün çevresini sarar ve aradığı özel özellik girdi hacminde olduğunda etkinleştirir. Şekil 3.15’ de sözel olarak anlatılan konvolüsyon işlemi şekilsel olarak gösterilmiştir. Her görüntü piksel değerlerinin matrisi olarak ifade edilmektedir. Girdi görüntüsü 3×4 boyutlarında ve 2×2 boyutlarında filtre piksel değerlerinin matrisi

şeklinde ifade edilmiştir. Gri renkli görüntülerde bu değer 0-255 arasındaki sayılardan oluşabilir (Wu 2016).



Şekil 3.15 Konvolüsyon işlemi (Wu 2016)

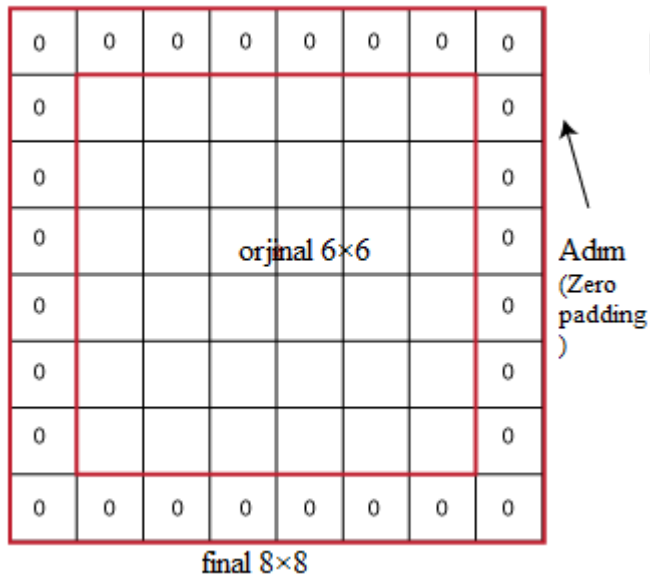
Konvolüsyon ve Korelasyon işlemleri görüntüden bilgi çıkarmak için kullanılan en temel işlemlerdir. Korelasyon işlemi, iki ve ya daha fazla değişken arasındaki ilişki derecesi ve türünü belirlemek için kullanılır. Değişkenler karşılaştırılır ve ne kadar benzer olduğu belirlenir. Korelasyon +1 ile -1 arasında değişebilir. +1'e yakın değerler yüksek pozitif korelasyonu gösterirken ve -1'e yakın değerler yüksek derecede negatif korelasyonu göstermektedir (Jacobs 2005).

Konvolüsyon katmanındaki her bir filtre sadece, girdi hacmindeki yerel bir alana bağlıdır. Bu bağlantının uzaysal boyutu **alıcı alan** (receptive field) olarak ifade edilmektedir ve bu alanın boyutu ve derinliği filtrenin boyutu ve girdinin derinliği ile aynı olmalıdır. Konvolüsyonel sinir ağı, kullanacağı filtre değerlerinin ne olacağını eğitim aşamasında tek başına öğrenmektedir. Ancak eğitim sürecinden önce ağ mimarisi, kullanılacak filtre sayısı, boyutu gibi parametrelerin belirlenmesi gerekmektedir. Ağın sahip olduğu filtre sayısı ne kadar fazla ise özellik çıkarımında, görünmeyen kalıpları tanımda o kadar başarılı olmaktadır. Özellik haritasının boyutu derinlik, adım ve dolgu parametreleri ile kontrol edilmektedir. Bu parametrelerin değerlerinin eğimden önce belirlenmesi gerekmektedir (Anonymous 2016d).

Adım (stride) ve dolgu (zero-padding) , filtre boyutu seçildikten sonra belirlenmek zorunda olan, her katmanın davranışını değiştirmek için değiştirilebilen 2 temel parametredir. Derinlik, adım (stride), dolgu (zero-padding) çıktı değerini belirleyen hiper parametre olarak da değerlendirilmektedir (Wuo 2016).

Derinlik, konvolüsyon katmanında kullanılan filtre sayısına tekabül eder. **Adım**, girdi üzerinde gezdirilecek filtrenin her adımda hareket etmesi gereken piksel sayısını ifade eder. Örneğin adım sayısı 2 ise, girdi üzerinde hareket eden filtre 1 kerede 2 piksel sıçrama yapmaktadır. Böylece mekansal olarak daha küçük çıktı hacimleri üretilmektedir ve çıktı hacmi tam sayı olarak elde edilir.

Dolgu kavramı girdi hacmi ile çıktı hacminin aynı boyutta olmasını sağlamaktadır. P değeri 1 olduğunda, girdinin bütün çevresine 0 koyulur, p değeri 0 olduğunda 0 koyulmaz (Anonymous 2017d).

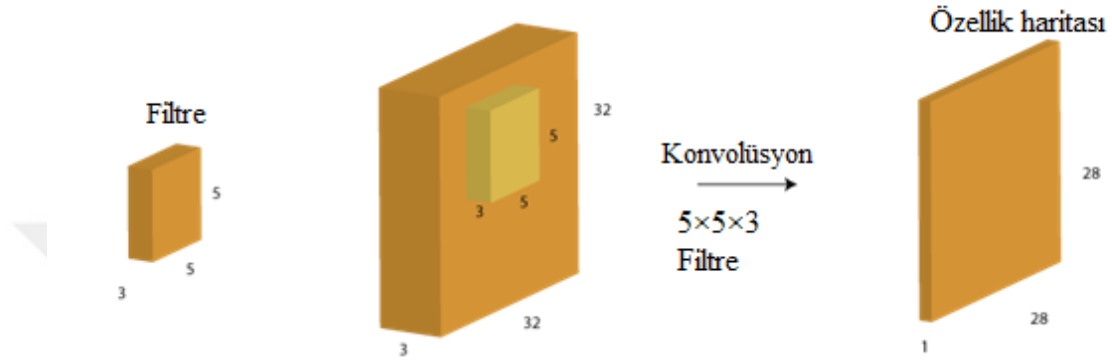


Şekil 3.16 Adım işlemi

Girdi matrisinin sınır elemanlarına filtre uygulanabilmesi için, bu matrisin etrafı sıfırlarla doldurulmaktadır. Bu işlem özellik haritalarının boyutunun kontrol edilmesini,

sağlamaktadır. Girdi matrisi etrafına dolgu eklemek geniş konvolüsyon, eklememek ise dar konvolüsyon olarak adlandırılmaktadır (Kalchbrenner vd. 2014).

Bir konvolüsyon katmanında filtrenin nasıl çalıştığı şekil 3.17’de şekilsel olarak gösterilmiştir.



Şekil 3.17 Konvolüsyon örneği (Anonymous 2017g)

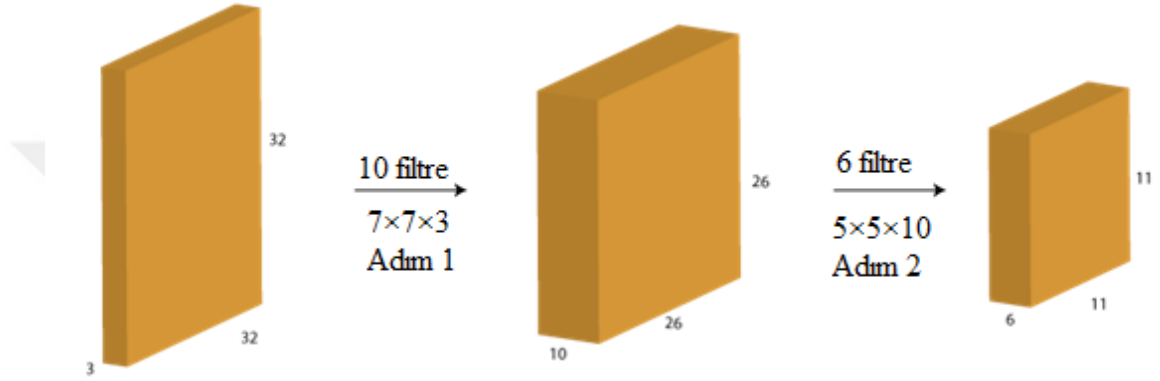
Örnek girdi boyutları 32×32×3 olsun. 32×32 değeri, girdinin genişlik ve yüksekliğini 3 değeri ise derinlik değerini ifade etmektedir. Buradaki 3 görüntülerin temelini oluşturan kırmızı, yeşil ve mavi renklerine karşılık gelmektedir. Örnek filtre boyutu ise 5×5×3 piksel değerinde olsun. Bir filtre üzerinde dolaştığı görüntünün genişlik ve yükseklik değerinden küçük boyutta olur ancak filtre derinliği ile girdi derinliği eşit olmak zorundadır. Filtre bir dizi ağırlık değeridir. Şekilde görüldüğü gibi filtre görüntünün sol üst köşesinden başlayarak sol alt köşeye kadar tarama yapar. Her konumda piksellerin ağırlıkları toplamı

$$W^T \times X + b \quad (3.6)$$

Denklem 3.6’da belirtilen formül ile hesaplanır ve yeni bir değer elde edilir. Bu değerler özellik haritasını oluşturur. (Örnekte gösterilen filtre ağırlık değeri: 5×5×3=75 +1 (bias) dir.) yukarıdaki örnekte özellik haritası boyutu 28×28×1 olmuştur. Derinliğin 1 olmasının sebebi tek filtre kullanımıdır. Genişlik ve yükseklik değerleri ise

$$\frac{I - F}{S} + 1 \quad (3.7)$$

hesaplaması ile bulunmaktadır. Burada I görüntü boyutu, F filtre boyutu, S ise adım sayısını ifade etmektedir. Görüntü boyutu 32 ×32 piksel filtre boyutu 5×5 piksel ve adım sayısı 1 olduğunda; özellik haritası boyutu 28 olmaktadır (Anonymous 2017g).



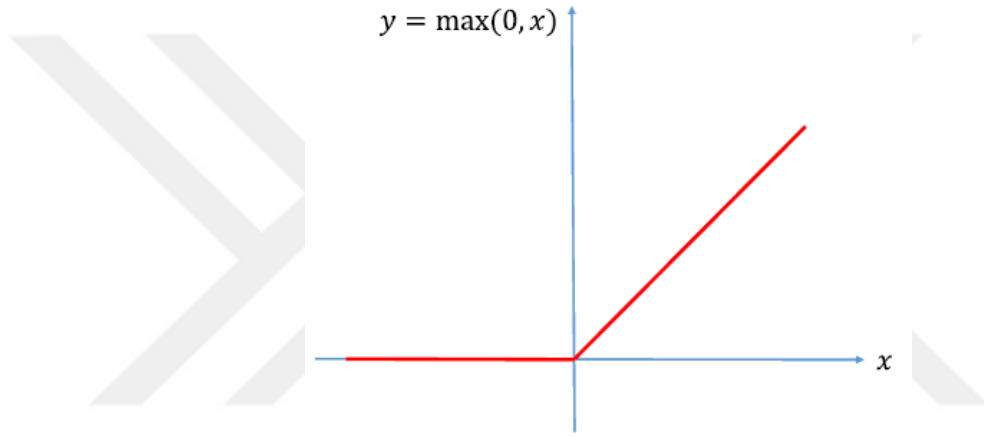
Şekil 3.18 Filtre Kullanım örneği (Anonymous 2017g)

Şekil 3.18 konvolüsyon katmanına bir diğer örnektir. İlk katmanda 10 adet filtre kullanıldığı için katmanın çıktısının derinliği 10 olmuştur. Aynı şekilde ikinci katmanda 6 filtre uygulanmış katmanın çıktı derinliği 6 olmuştur.

Konvolüsyon katmanında değerlerin formülleri özetle şu şekildedir: Girdi boyut hacmi: $W_1 \times H_1 \times D_1$ olsun. W; genişlik, H yükseklik ve D derinlik değeridir. K filtre sayısı, F filtre boyutu, S adım değeri ve P dolgu değeri olsun. Elde edilen çıktı boyut hacmi $W_2 \times H_2 \times D_2$ ise; $W_2 = (W_1 - F) + 2 \times P / S + 1$, $H_2 = (H_1 - F) + 2 \times P / S + 1$, $D_2 = D_1$, Toplam parametre sayısı = $((F \times F \times D) \times K)$, Toplam parametre sayısı = her bir filtre için parametre sayısı (her bir filtrenin hacmi) × filtre sayısı (Anonymous 2017f).

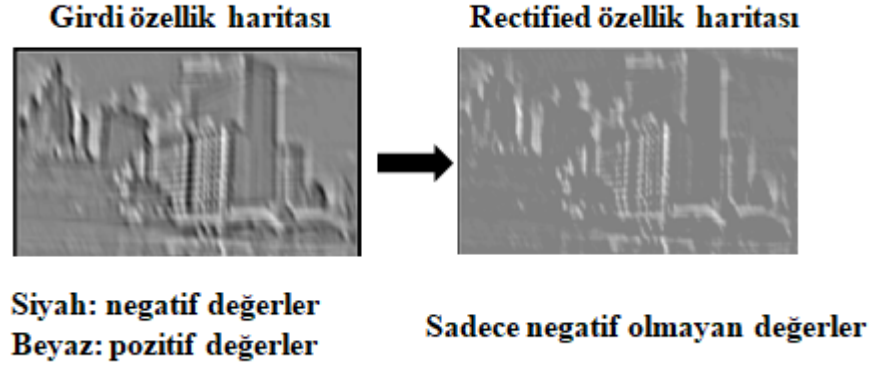
Derin öğrenme algoritmalarında konvolüsyon katmanından hemen sonra aktivasyon fonksiyonu olarak genellikle RectifiedLinearUnit (ReLU) fonksiyonu kullanılmaktadır ve **ReLU katmanı** olarak adlandırılmaktadır. Bu katmanda herhangi bir parametre

öğrenimi yapılmamaktadır. Bu fonksiyonun uygulanmasındaki amaç, konvolüsyon katmanları sırasında doğrusal işlemler uygulanan sisteme doğrusal olmayanlık getirebilmektir (Wuo 2016). Tanh ve sigmoid fonksiyonları da aktivasyon fonksiyonu olarak uygulanırken hesaplamalardaki verimliliği sayesinde ağıın eğitim süresini hızlandırdığı için ReLU fonksiyonu kullanılmaya başlanmıştır. Gerçek dünya verileri çoğunlukla doğrusal olmayan verilerdir. Ağıın bu verileri daha rahat öğrenebilmesi için, ReLU fonksiyonu ağına doğrusal olmayanlığı tanıtmaktadır. ReLU fonksiyonu grafiğı şekil 3.19’da gösterilmiştir.



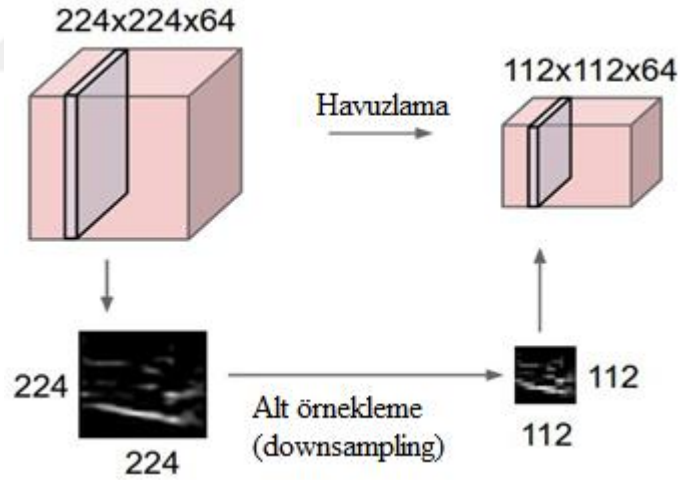
Şekil 3.19 Aktivasyon fonksiyonu (Wuo 2016)

Şekil 3.20’de ReLU işleminin açıkça anlatılmıştır. ReLU fonksiyonu özellik haritasındaki her bir piksele uygulanarak, negatif olan değerleri 0 ile değiştirmektedir. Bu işlem uygulandıktan sonra elde edilen çıktıya “Rectified” özellik haritası denilmektedir.



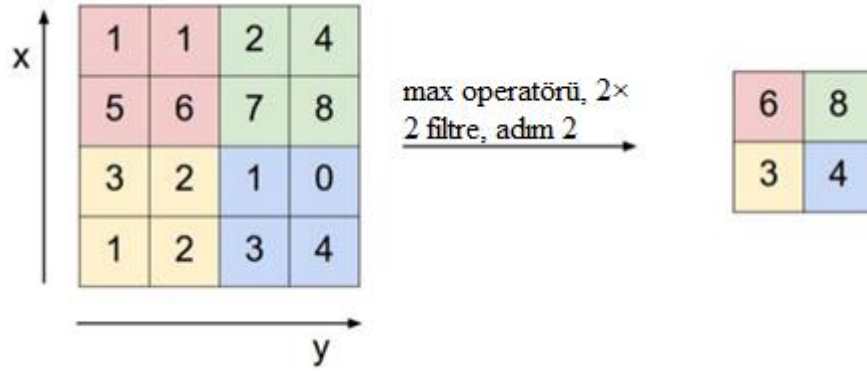
Şekil 3.20 ReLU işlemi (Anonymous 2017d)

Konvolüsyon ve ReLU işlemlerinden sonra genellikle **Havuzlama katmanı** uygulanmaktadır. Havuzlama işleminin temel amacı, girdi görüntüsünün uzaysal boyutunun yavaş yavaş azaltılması ve böylece parametrenin sayısının ve modelin hesaplama karmaşıklığını daha da azaltmasıdır.



Şekil 3.21 Havuzlama katmanı işlevi (Anonymous 2017f)

ReLU işleminden sonra elde edilen Rectified özellik haritasına max, average, toplam gibi havuzlama operatörü uygulanmaktadır. Max ve ortalama operatörleri sıklıkla kullanılmaktadır. Şekil 3.22'de max operatörünün çalışma mantığı anlatılmaktadır.



Şekil 3.22 Havuzlama katmanı (Anonymous 2017f)

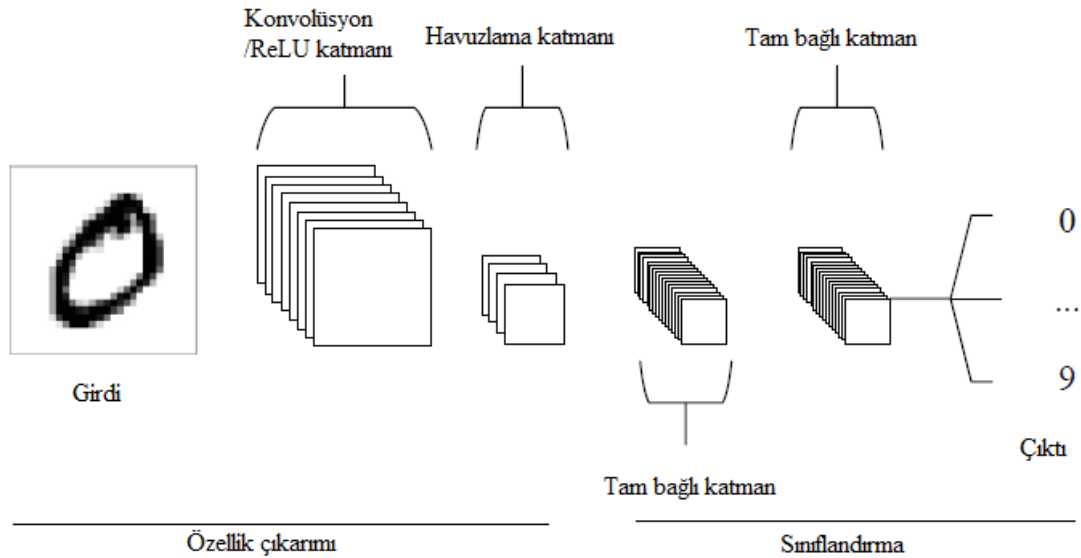
Şekilde gösterildiği gibi, her filtrenin sonucuna max operatörünü uygulayarak girdiler alt kategorilere ayrılmıştır. 2 x 2 boyutundaki pencere, özellik haritası üzerinde kaydırılıp her bir bölgede maksimum değeri alınır. Yukarıda şekil 3.24’de gösterildiği gibi, bu şekilde özellik haritasının boyutu azalmaktadır. Ancak derinlik değeri sabit kalmaktadır. Bu sayede özellik haritası boyutu daha küçük ve yönetilebilir hale gelmektedir. Yerel bir alanda maksimum ya da ortalama değer alındığı için, girdi görüntüsündeki küçük değişimlere, bozulmalara karşı ağın değişmez olması sağlanmaktadır. Yani girdi değerindeki küçük bozulma, havuzlama katmanının çıktısını değiştirmemektedir (O’Shea ve Nash 2015).

Bir önceki katmana göre çıktı boyutunu azaltılır ancak en belirgin özellikler tutulur. İşlevi: ağıdaki parametrelerin ve hesaplamaların miktarını azaltmak ve dolayısıyla ezberlemeyi (overfitting) kontrol etmek için girdi boyutunu aşamalı olarak azaltmaktır. Sınıflandırma için gerekli olan sabit boyutlu çıktı matrisini sağlar. Havuzlama operatörü her filtre için bir değer döndürdüğü için, filtre sayısı boyutunda bir çıktı elde edilir. Bu sayede değişken boyutlu filtreler kullanılıp sabit boyutlu çıktı matrisi elde edilir (Anonymous 2017h).

Tam bağlı katmanı, genellikle konvolüsyon ve havuzlama katmanlarının sonunda kullanılmaktadır. Bu katmanda her bir piksel normal bir sinir ağında olduğu gibi ayrı bir nöron olarak düşünülür ve tahmin edilecek sınıf sayısı kadar nöron içerir.

Konvolüsyon, ReLU ve havuzlama katmanları sonucunda elde edilen çıktı; girdi görüntüsüne ait dağınık özellikler içermektedir. Bu özelliklerin tümünün bir sonraki aşamada güçlü yeteneklere sahip özellikleri oluşturmak için kullanımı bu katmanın amacıdır. Bu amaçla, tam bağlı katman kendinden önceki katmanlardan gelen üst düzey özellikleri kullanarak, girdi görüntüsünü sınıflandırmaktadır. Ayrıca, bu katman sayesinde üst düzey özelliklerin doğrusal olmayan kombinasyonlarının öğrenilmesi sağlanmaktadır.

Tam bağlı(fullyconnected) olma hali; bir sonraki katmanda meydana gelecek hesaplama (çıkı bulma) bir önceki katmandaki bütün girdileri gerektirir yani bir önceki katmanda bulunan her bir nöronun, bir sonraki katmanda bulunan her bir nörona bağlı olma hali olarak ifade edilmektedir (Wuo 2016).



Şekil 3.23 Tam bağlı katmanı (O'Shea ve Nash 2015)

Normalleştirme katmanı; KSA mimarilerinde, bazen biyolojik beyninde gözlemlenen inhibisyon şemalarını uygulama amacıyla kullanılmak üzere pek çok tip normalleştirme katmanı önerilmiştir. İnhibisyon kavramı, bir nöronun etrafındaki komşularını bastırma kapasitesini ifade eder. Bu katmanlardan bir tanesi ve AlexNet mimarisinde kullanılan Local Response Normalization katmanıdır. Bu katman, inhibisyonu uygular ve bu

işlem ReLU katma nöronları için yararlıdır. Çünkü ReLU nöronları sınırsız etkinliğe sahiptir ve bunu normalleştirmek için LRN' ye ihtiyacımız vardır. Yüksek frekanslı özellikleri büyük tepki ile tespit etmek amacıyla, aktif nöronun yerel mahallesinde normalleşirsek komşularıyla karşılaştırıldığında daha hassas olur. Aynı zamanda bu katman, herhangi bir yerel komşulukta eşit olarak verilen yanıtları azaltacaktır. Bütün değerler büyükse, o zaman bu değerleri normalleştirmek hepsini azaltır. Bu nedenle, temelde bir çeşit inhibisyonu teşvik etmek ve nispeten daha büyük aktifleştirmelerle nöronları yükseltmek istenmektedir. Aynı kanal içinde normalleştirme veya kanallar arasında normalleştirme olmak üzere iki normalizasyon türü vardır. ReLU katmanı doygunluk önlemek için giriş normalleştirme gerektirmeyen özelliğe sahiptir. Eğer en azından bazı eğitim örnekleri bir ReLU'ya pozitif bir girdi oluşturursa, bu nöronda öğrenme gerçekleşecektir. Bununla birlikte, yine de yerel normalleştirme şemasının genellemeye yardımcı olduğunu bilinmektedir. Filtre haritalarının sıralanışı eğitim başlamadan önce belirlenir. Bu tür tepki normalizasyonu, farklı filtreler kullanılarak hesaplanan nöron çıktıları arasında büyük faaliyetler için rekabet yaratarak, gerçek nöronlarda bulunan türden esinlenerek bir yanal inhibisyon şekli uygulamaktadır (Krizhevsky vd. 2012).

Dropout metodu; ağırlık eğitim verilerine çok “uygun” olmamasını sağlar ve dolayısıyla aşırı uyum probleminin hafifletilmesine yardımcı olur ve her gizli nöronun çıkışını 0.5 olasılık ile sifira ayarlamaktan oluşur. Bu şekilde “bırakılan” nöronlar, ileri geçiş katkıda bulunmaz ve geri yayılıma katılmazlar. Bu nedenle, bir girdi her sunulduğunda, sinir ağı farklı bir mimari örnekler, ancak tüm bu mimariler ağırlık paylaşır. Bu teknik, nöronların kompleks uyumlarını azaltır, çünkü bir nöron diğer belirli nöronların varlığına güvenemez. Bu nedenle, diğer nöronların birçok farklı rastgele altkümeleri ile birlikte yararlı olan daha sağlam özellikleri öğrenmek zorunda kalır. Test zamanında, tüm nöronları kullanılır, ancak üstel olarak çok fazla bırakma ağırları tarafından üretilen tahmini dağılımların geometrik ortalamasının alınması için makul bir yaklaşımı olan 0.5 ile çıktıları çarpılır. Bu işlem sadece ağırlık eğitimi sırasında uygulanır, dolayısıyla bu aşamadaki nöronlara uygulanmaktadır. AlexNet mimarisinde bu teknik ilk iki tam bağlı katmanda kullanılmıştır (Krizhevsky vd. 2012).

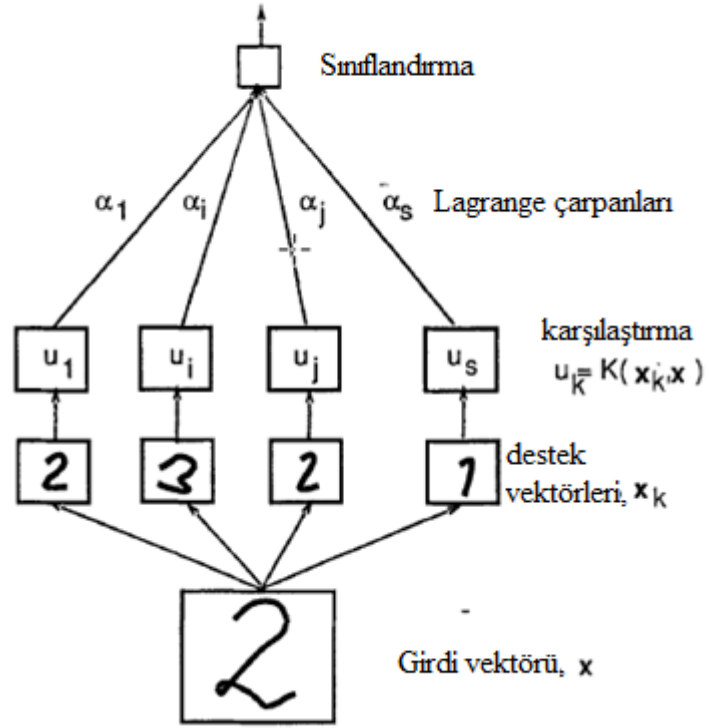
3.6 Destek Vektör Makineleri

Destek vektör makineleri, sınıflandırma ve regresyon analizi problemlerinin çözümü için geliştirilmiş, örüntü tanıma alanında etkili, başarılı makine öğrenimi algoritmalarından biri olan gözetimli (supervised) öğrenme modelleridir.

DVM'in temelleri Vapnik ve Cortes'in 1995 yılında yaptığı çalışmaya dayanmaktadır. Destek vektör makineleri, iki gruplu sınıflandırma problemleri için geliştirilmiş bir öğrenme algoritmasıdır. Makine kavramsal olarak belirtilen fikri uygular: giriş vektörleri çok yüksek boyutlu bir özellik alanına doğrusal olmayan şekilde eşlenir. Bu özellik alanında doğrusal bir karar yüzeyi oluşturulmuştur. Karar yüzeyinin özel özellikleri, algoritmanın yüksek genelleme kabiliyetini sağlar. Destek vektör makinesi arkasındaki fikir, eğitim verisinin hatasız olarak ayrılabilmesi için daha önce uygulanmıştı. Ancak Vapnik ve Cortes çalışmasında ayrılabilir olmayan eğitim verilerinde Destek vektör makinelerinin yüksek genelleme yeteneği gösterilmiştir (Vapnik ve Cortes 1995).

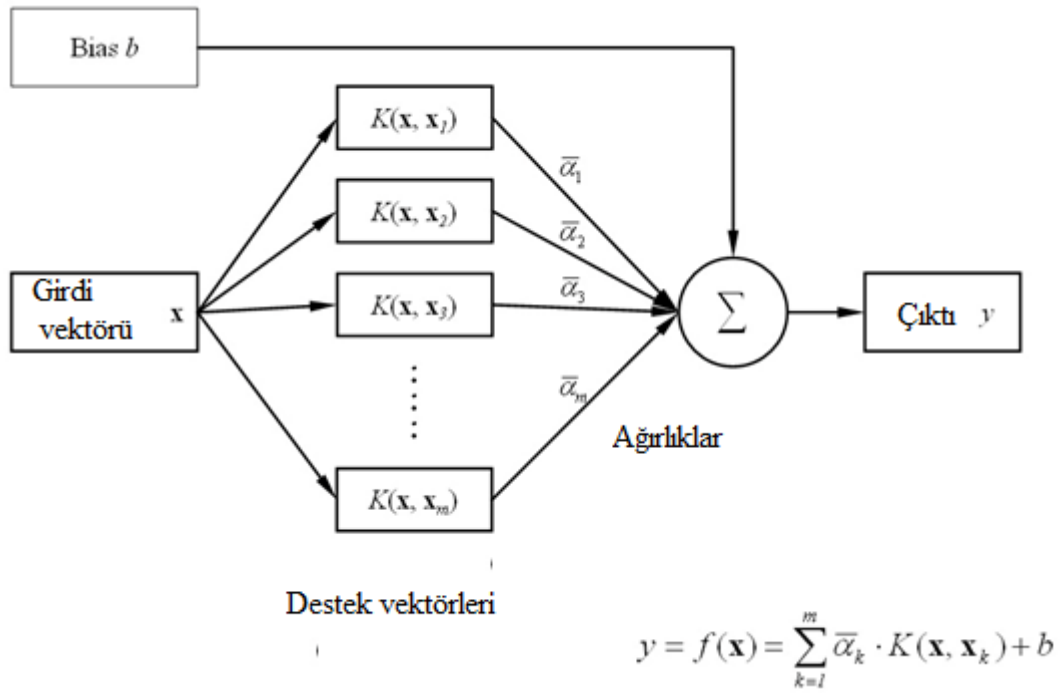
DVM'de amaç, farklı sınıflara ait destek vektörleri arasındaki mesafeyi maksimum yapacak en iyi hiper düzlemin bulunmasıdır (Ayhan ve Erdoğan 2014).

Destek vektör makineleri ağ mimarisi şekil 3.24'de belirtilmiştir.



Şekil 3.24 Destek vektör makinelerinde sınıflandırmanın kavramsal açıklaması (Vapnik ve Cortes 1995)

Destek vektör makinesi ile bilinmeyen bir örneği sınıflandırma işlemi ilk olarak örneği yüksek boyutlu özellik alanına çevirmek ile başlar. Bu özellik alanına yerleştirilen en uygun hiperdüzlem çıktıyı belirler (Vapnik ve Cortes 1995).



Şekil 3.25 Destek vektör makineleri çalışma prensibi

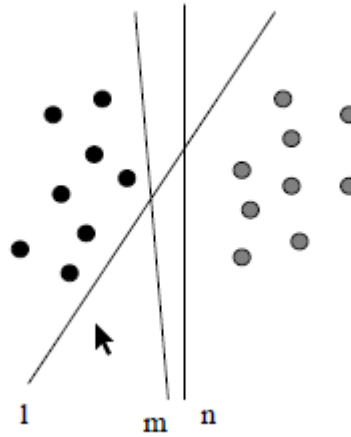
Yukarıdaki şekilde gösterildiği gibi, $K(x, x_m)$ çekirdek fonksiyonlarını ve α Lagrange çarpanlarını ifade etmektedir. Çekirdek fonksiyonları yardımıyla girdilerin iç çarpımları hesaplanmaktadır. Lagrange çarpanları ise ağırlıkları ifade etmektedir. Şekil 3.25’de bulunan denklemde belirtildiği gibi çıktı değeri, ağırlıklar ile girdilerin iç çarpımı toplamlarına bias (b) değerinin eklenmesi ile elde edilmektedir.

DVM iki sınıflı ve çok sınıflı sınıflandırma probleminin çözümü için geliştirilmiş makine öğrenmesi algoritmalarıdır (Ayhan ve Erdoğan 2014).

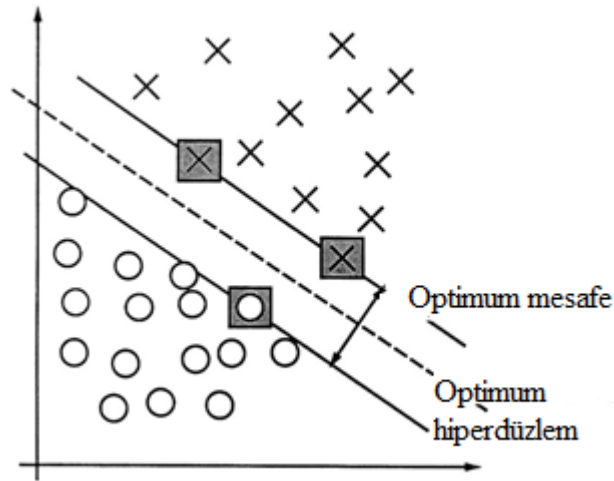
3.6.1 İki sınıflı doğrusal destek vektör makineleri

Farklı sınıflara ait verileri birbirinden en iyi şekilde ayıracak $w \cdot x + b$ (hiperdüzlem karar fonksiyonu), doğrusal hiper düzleminin bulunmasına yardımcı olan, sonsuz sayıdaki doğru içerisinde marjini en yüksek yapacak olan doğrunun seçilmesi hedefleyen denetimli öğrenme algoritmaları sınıfına ait makine öğrenimi algoritmasıdır (Ayhan ve Erdoğan 2014).

Şekil 3.26 anlaşılacağı gibi iki sınıfı birbirinden ayıran birçok doğrusal sınıflandırıcı vardır ancak yalnızca bir tanesi marjini (her sınıfın en yakın veri noktası arasındaki mesafeyi) en yüksek yapmaktadır. Bu sınıflandırıcıya optimal ayırıcı hiperdüzlem (optmal seperating hyperplane) denir (Guo vd. 2000).



Şekil 3.26 İki boyutlu uzayda ayrılabilir DVM örneği

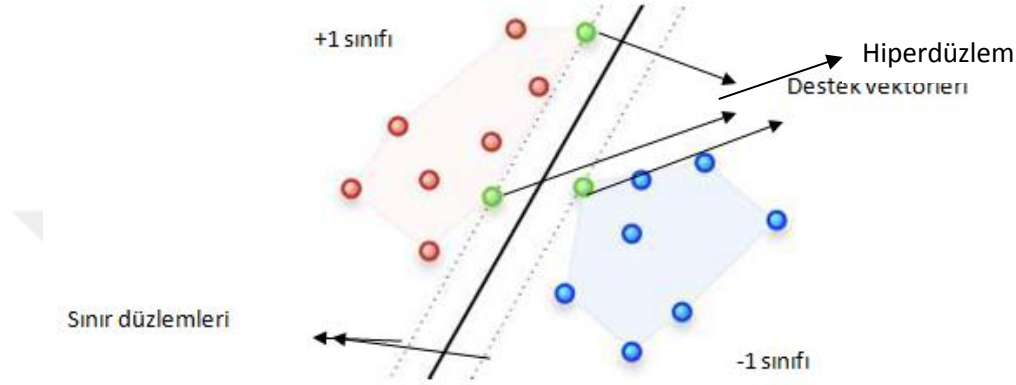


Şekil 3.27 İki boyutlu uzayda ayrılabilir DVM örneği

Sınıflandırma problemleri için hiperdüzlem karar fonksiyonu,

$$D(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b, \quad \mathbf{w} \in \mathcal{X}, \quad b \in \mathfrak{R} \quad (3.8)$$

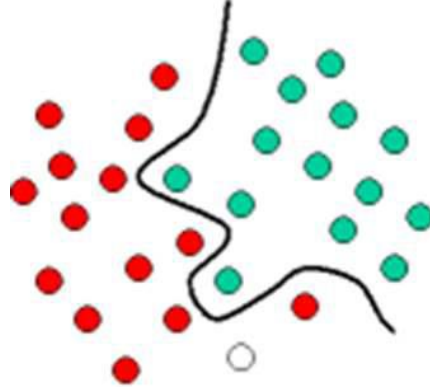
Denklem 3.8' de belirtildiği gibi tanımlanır. \mathcal{X} girdilerin uzayını (örneğin \mathcal{R}^d) ve $\langle w, x \rangle$, \mathcal{X} uzayında iç çarpımını, n eğitim verisi sayısını, $(x_1, y_1), \dots, (x_n, y_n)$, $x \in \mathcal{X}$, $y \in \{+1, -1\}$ örneklerin ait olduğu sınıfı ifade eder. Eğitim süreci sonunda, w ve b katsayı tahminleri kullanılarak, yeni gözlenen x örüntüleri için fonksiyonuna göre çıktı üretilir (Başer ve Apaydın 2015).



Şekil 3.28 İki sınıflı problem örneği (Başer ve Apaydın 2015)

3.6.2 İki sınıflı doğrusal olmayan destek vektör makineleri

Destek Vektör Makineleri, doğrusal olmayan örnek uzayını, örneklerin doğrusal olarak ayrılabilceği bir yüksek boyuta aktararak, farklı örnekler arasındaki maksimum sınırın bulunması esasına dayanır. Bu alanda karşılaşılan problemlerin büyük çoğunluğu, birçok farklı bileşenden oluşan problemlerdir ve doğrusal olarak ayrılmış bir yapı halinde değildirler (Güneş ve Yiğit 2012). Doğrusal olmayan DVM, veri setinin doğrusal bir fonksiyonla tam veya belirli bir hata ile ayrılamaması durumunda kullanılan algoritmalardır. Gerçek yaşam problemlerinde bir veri setinin hiper düzlem ile doğrusal olarak ayrılması çoğunlukla mümkün olmamaktadır.



Şekil 3.29 İki Sınıflı Problem İçin Doğrusal Ayrılmama Durumu (Başer ve Apaydın 2015)

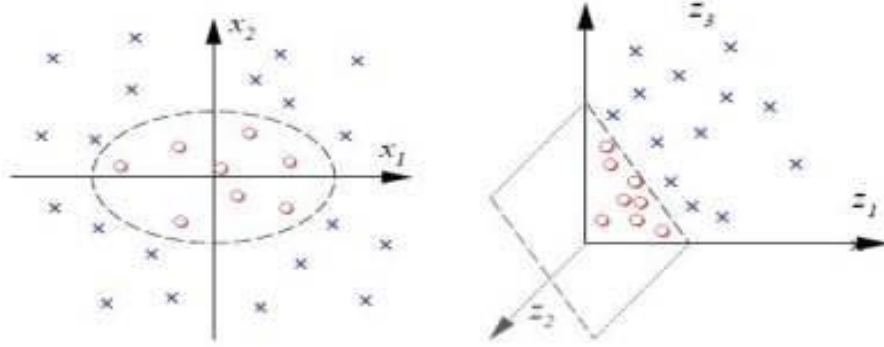
Verilerin doğrusal ayrılmaması durumunda p-boyutlu girdi vektörü x 'in P-boyutlu özellik vektörü Φ 'ye dönüştürülmesi gerekmektedir. Yani daha yüksek boyutlu bir uzaya dönüştürülmektedir. Bu işlemin gerçekleştirilebilmesi için, optimal ayırma düzleminin özellik uzayında tanımlanabilmesi gerekir. Bu amacı gerçekleştirmek için doğrusal olmayan haritalama yaklaşımından yararlanılır. “Doğrusal olmayan haritalama”, orijinal girdi uzayı x 'in bir Hilbert uzayı olan daha yüksek boyutlu F özellik uzayına dönüştürülerek doğrusal ayrımının gerçekleştirilmesi için kullanılan bir yaklaşımdır (Ayhan ve Erdoğan 2014)

Eğitim verisinin başka bir uzayın altkümesi olacak biçimde dönüştürülmesi makine öğrenmesi alanında uzun zamandır bilinen ve uygulanan bir yaklaşımdır. Burada \mathcal{X} girdi uzayı, $\mathcal{F}=\{\Phi(\mathbf{x}):\mathbf{x}\in\mathcal{X}\}$ ise nitelik uzayı olarak adlandırılır. $\Phi:\mathcal{X}\rightarrow\mathcal{F}$ doğrusal olmayan bir dönüşümü göstermek üzere, doğrusal olmayan sınıflandırma problemlerinde karar fonksiyonu

$$D(\mathbf{x})=\sum_{i=1}^n w_i \phi_i(\mathbf{x}) + b \quad (3.9)$$

biçiminde tanımlanır. Burada, m nitelik uzayının boyutunu göstermek üzere $\Phi(\mathbf{x})=[\phi_1(\mathbf{x}) \phi_2(\mathbf{x}) \dots \phi_m(\mathbf{x})]'$ dir (Başer ve Apaydın 2015).

Aşağıdaki şekilde iki boyutlu veri seti üç boyutlu özellik uzayına taşınarak veri setinin doğrusal ayrımı Doğrusal olmayan haritalama yaklaşımı ile sağlanmıştır (Ayhan ve Erdoğan 2014).



Şekil 3.30 İki boyutlu veri setinin üç boyutlu özellik uzayına taşınması (Cortes ve Vapnik 1995)

3.6.3 Çok sınıflı destek vektör makineleri

Çok sınıflı model tanıma sistemleri, iki sınıflı DVM'lerin birleşimi ile elde edilebilir. Destek vektör makinelerinde çok sınıflı problemleri çözmek için uygulanan bire karşı hepsi (one vs all) ve ikili (pairwise, one vs one) olmak üzere iki temel yaklaşım vardır. Bire karşı hepsi yaklaşımında k sayıda sınıf için k sayıda DVM eğitilmektedir. Her bir DVM tek bir sınıfı kalan tüm sınıflardan ayırmaktadır (Heisele vd. 2001). K sayılı sınıflı sınıflandırma için, k sayıda ikili sınıflandırıcılar (binary classifier) oluşturulur. M inci sınıftan gelen veri m inci ikili sınıflandırıcı ile pozitif örnek olarak eğitilirken, kalan (k-1) sınıf negatif örnek olarak eğitilir. Test sırasında sınıf etiketi, en yüksek çıkış değerini veren ikili sınıflandırıcı tarafından belirlenir (Wang ve Xue 2014).

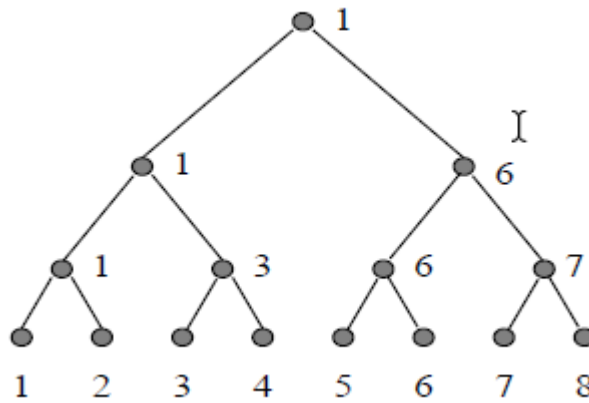
$$g^j(x) = b^j + \sum_{i=1}^m k(x + x_i) y_i \alpha^j \quad (3.10)$$

Bir sınıfı diğerinden ayırmak için eğitilecek ikili sınıflandırıcılar kümesi çok sınıflı sınıflandırıcıyı oluşturur. Yukarıdaki formül x noktasının hiper düzleme olan uzaklığını

ifade eden değeri vermektedir. Bu değere güven değeri denir. X noktası güven değerini en büyük yapan sınıfa atanır.

İkili yaklaşımda k kare sayıda DVM eğitilmektedir. Her DVM, bir çift sınıfı ayırır. Her olası sınıf çifti için sınıflandırıcı eğitilir. M sınıf için $(m-1)m/2$ tane ikili sınıflandırıcı elde edilir. Elde edilen her sınıflandırıcı, bilinmeyen x noktasını sınıflandırma için eğitilir. x noktasının kaç kere hangi sınıflandırıcıya atandığı hesaplanır. Hangi sınıf etiketi en fazla ise bilinmeyen x noktası o sınıftadır demektir.

Bazı yüz tanıma uygulamalarında, test aşamasını kolaylaştırmak için sınıflandırıcılar aşağıdan yukarıya (bottom-up) ağaç yapıları halinde düzenlenir ve her ağaç düğümü DVM yi temsil eder (Heisele vd. 2001). Aşağıdaki şekilde görüldüğü gibi, sınıflandırma için aşağıdan yukarıya doğru ikili bir ağ oluşturulur. Şekildeki örnekte veri kümesinde 8 sınıf olduğu varsayılmıştır. Sınıfları temsil eden sayılar rastgele yerleştirilmiştir. Ağacın en alt düzeyinden başlanarak, ikili sınıflar arasında karşılaştırma yapılarak, bir sınıf numarası seçilir ve kazanan olarak adlandırılır. Kazanan sınıf test aşamasında, bir üste geçmiş olur. Böylece kazanan sınıf en sonunda en ikili karar ağacının en üstünde kalacaktır (Guo vd.2000).



Şekil 3.31 İkili karar ağacı (Guo vd. 2000)

Sınıflandırma performansı açısından iki yaklaşım arasında teorik bir analiz yoktur. Eğitim için harcanan efor baz alındığında, bire karşı hepsi yaklaşımı diğerine göre daha çok tercih edilmektedir. Çünkü bu yaklaşımda n sayıda DVM eğitilirken diğerinde n kare sayıda DVM eğitilmektedir. Yüz tanıma alanında yapılan uygulamalarda iki yaklaşımda benzer sınıflandırma performansı gösterdiği görülmektedir (Heisele vd.2001) .İkili yaklaşımda, uygulama esnasında oluşturulan sınıflandırıcı sayısı diğerine göre çok büyüktür. Ancak sınıflandırıcının boyutu daha küçüktür buda eğitimin hızlı olmasını sağlamaktadır (Wang ve Xue 2014). Ancak yüz tanıma uygulamalarında sınıf sayısı fazla olduğu için DVM sayısının sınıf sayısı ile doğrusal olduğu bir e karşı hepsi yaklaşımı tercih edilmektedir (Heisele vd. 2001).

3.7 Tez Kapsamında Geliştirilen Uygulamanın Açıklaması

Klasik ve derin öğrenme yöntemleri ile geliştirilen algoritmanın eğitimi ve test edilmesi için AT&T (Anonymous 2017i), faces96 (Anonymous 2017i), ve LFW (Anonymous 2017j), olmak üzere üç farklı veri seti kullanılmıştır. 40 farklı kişinin toplam 400 resminden oluşan AT&T veri setinde bulunan her bir resim 92×112 pikseldir. 152 farklı kişinin toplam 3040 resminden oluşan faces96 veri setinde bulunan her bir resim 196×196 pikseldir.

Derin öğrenme ile yapılan yüz tanıma uygulamalarında sıkça kullanılan, 62 farklı kişinin toplam 2992 resminden oluşan “Labelled Faces in the Wild” (LFW) veri seti kullanılmıştır. Orijinal lfw veri setinde 5749 farklı kişinin toplam 13.233 resmi bulunmaktadır. Ancak geliştirilen uygulama için en az 20 farklı resmi bulunan kişiler kullanılmıştır.

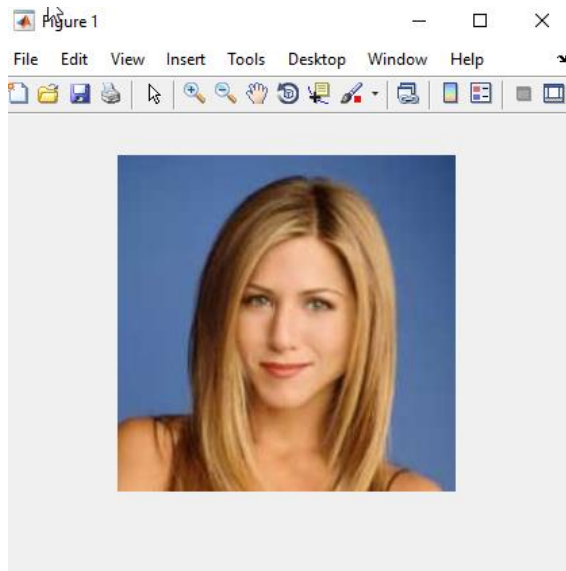
Resimde yüz tespiti sürecinde yüzün bulunması ve ilgili bölgenin hesaplanması işlemi gerçekleştirilmiştir. Bu aşamada Viola&Jones algoritması kullanılmıştır. Ön hazırlık sürecinde görüntü; gürültü, aydınlatma, poz/dönüş gibi konuların üstesinden gelmek için ön işleme tabi tutulabilir. Öznitelik çıkarımı kısmında HOG ve derin öğrenme olmak üzere iki ayrı teknik kullanılmıştır. HOG metodu kullanılarak özellik çıkarımı işlemi için; resimdeki her piksele, onu doğrudan çevreleyen piksellere bakılır. Buradaki

amaç; mevcut pikselin, onu doğrudan çevreleyen piksellerle karşılaştırıldığında ne kadar karanlık olduğunu bulmaktır. Daha sonra görüntünün hangi yönde daha koyulaştığını gösteren bir ok çizilir. Bu oklar görüntünün ışıktan karanlığa akışını göstermektedir. Böylece orijinal görüntü, görüntü parlaklığından bağımsız olarak resmin temel özelliklerini yakalayan bir HOG gösterimine dönüştürülür. Derin öğrenme, bir yüzün hangi bölümlerinin ölçülmesi gerektiğini belirleyebilir. Her yüz sayısal değerli bir özellik vektörü ile ifade edilir. Bu özellikleri öğrenebilmesi için ağ eğitilir. Daha sonra bu özellikler ağa kaydedilir. Elde edilen özellikler çok sınıflı DVM metodu kullanılarak sınıflandırılmıştır. DVM sınıflandırıcısı, veritabanı özellik vektörünü test verisi özelliği vektörüyle eşleştirmek için kullanılır. Veritabanından en iyi eşleşen yüzleri bulur.

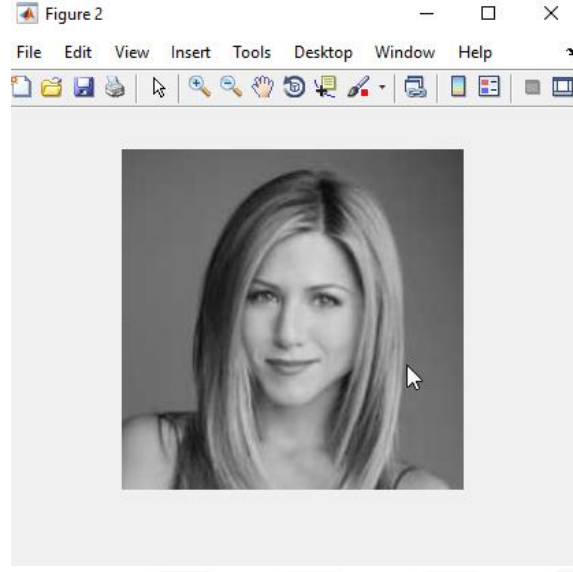
3.7.1 Klasik yöntem ile yüz tanıma algoritması

Klasik yöntem ile gerçekleştirilen yüz tanıma uygulaması aşamaları şu şekildedir:

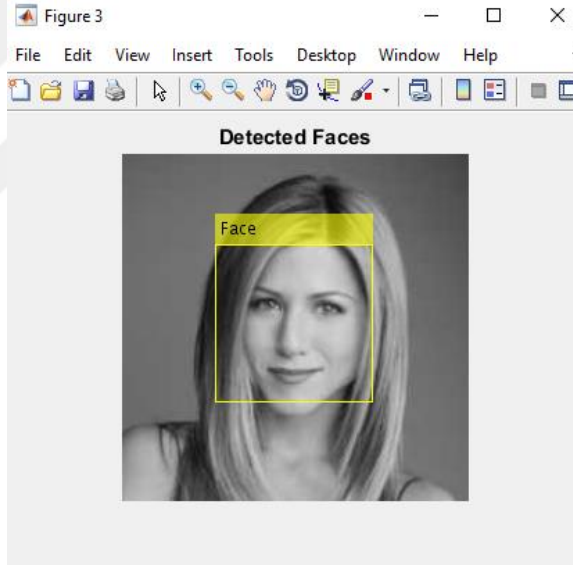
- Şekil 3.32’ de gösterildiği gibi orijinal resim girdi olarak alınmış, şekil 3.33’ deki gibi görüntü gri renge çevrilmiş ve şekil 3.34’ de gösterildiği gibi yüz tespiti aşamasında Viola&jones algoritması kullanılarak yüz tespit edilmiştir.



Şekil 3.32 Orijinal resim



Şekil 3.33 Gri renge çevrilmiş resim



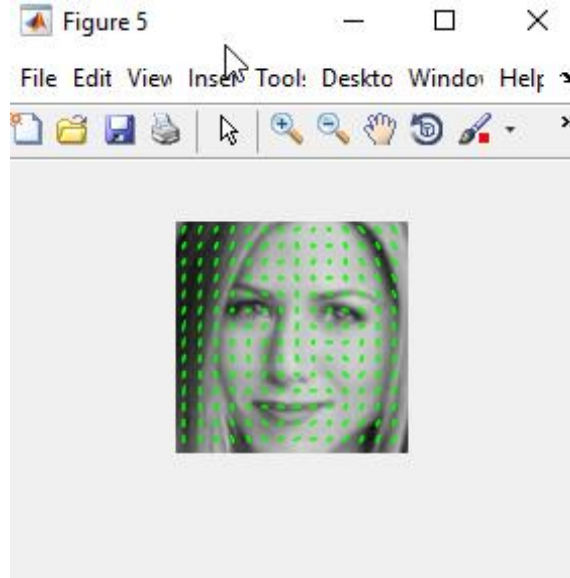
Şekil 3.34 Yüzün tespit edilmesi

- Yüz görüntüsünün hizalanması ve ön hazırlık kısmında; tespit edilen yüzler kırılmış ve 100×100 piksel olmak üzere yeniden boyutlandırılmıştır.



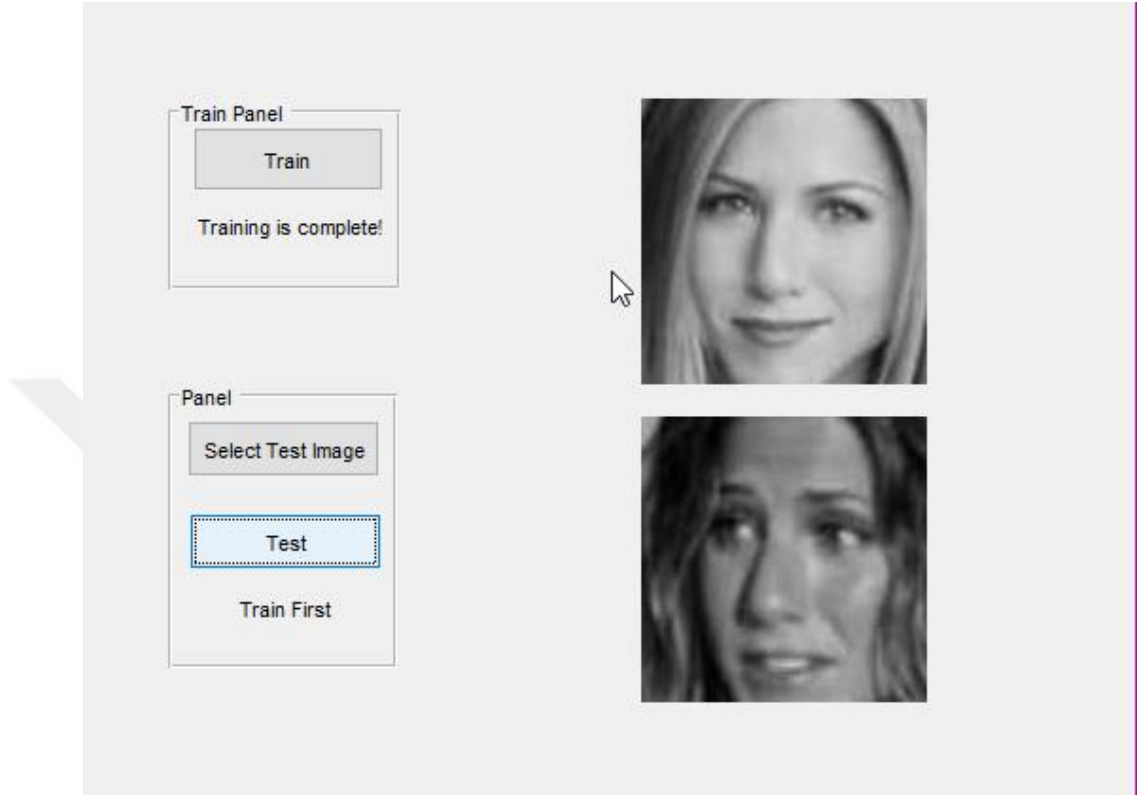
Şekil 3.35 Kırpılmış resim

- Veri setinde bulunan resimler sırasıyla % 60 -% 40, % 70-% 30, % 80-% 20, % 90-% 10 eğitim ve test verisi olmak üzere ayrılmıştır.
- Öznitelik çıkarımı kısmında; HOG algoritması kullanılarak özellik vektörü elde edilmiştir.



Şekil 3.36 HOG özellikleri

- Elde edilen özellik vektörü ile DVM sınıflandırıcısı eğitilerek yüz tanıma işlemi gerçekleştirilmiştir.



Şekil 3.37 Yüz tanıma işlemi

3.7.2 Derin öğrenme ile yüz tanıma algoritması

Makine öğrenmesi teknikleri, bilgisayarlarla görme alanında görüntülerde ilgi çekici nesnelere tespit etmek ve nesnelere kategorilerini sınıflandırmak veya nesnelere tanımlamak için kullanılmaktadır. Makine öğrenme teknikleri; noktaları, bölgeleri veya nesnelere temsil eden özellikleri çıkarır ve bu özellikleri sınıflandırma yapan bir modeli eğitmek için kullanırlar.

Geleneksel makine öğrenme tekniklerinde öznelik çıkarımı çok zaman alan ve manuel olarak yapılan bir işlemdir. Genellikle her bir görüntünün bir veya daha fazla görüntü işleme operasyonu ile işlenmesini içerir; örneğin her görüntünün ayırtedici bilgilerini

çıkarmak için gradient hesaplanması yönteminin kullanılması gibi. Ancak **Derin öğrenme** algoritmaları özellik, işaret ve görevleri doğrudan görüntü, metin ve ses arasından öğrenebilir ve böylece manuel özellik seçimi ihtiyacını ortadan kaldırır.

Derin öğrenme ile yüz tanıma algoritmasının oluşturulmasında, 1000 kategoride 1.2 milyondan fazla etiketli, yüksek çözünürlüklü resim içeren bir veritabanı olan ImageNet' te önceden eğitim görmüş bir Konvolüsyonel sinir ağı sınıflandırıcısı olan AlexNet mimarisinin basit bir nesne tespit ve tanıma uygulaması esas alınmıştır (Krizhevsky vd. 2012). Belirtilen örnekte, videoda bir hayvanı algılamak için bir algoritma eğitilmiş ve hayvanın kedi veya köpek olarak doğru bir şekilde etiketlenmesi sağlanmıştır (Anonymous 2017k). Bunun için sınıflandırma yapabilen ve ham görüntüden özellik çıkarabilen derin öğrenmenin özel bir türü olan konvolüsyon sinir ağı kullanılmıştır.

Konvolüsyonel sinir ağları, farklı bir sorunu çözmek için (örneğin, yüz tanıma işlemi) kullanılacak yeni bir sınıflandırıcıyı eğitmek için gerekli olan özellikleri çıkarmayı öğrenebilirler.

Yapılan tez çalışmasında, belirlenen veri setlerinde bulunan görüntülerden ayırt edici özellikleri çıkarmak için Konvolüsyon sinir ağı kullanılmış ve daha sonra sınıflandırma yapabilmek için gerekli bir makine öğrenme algoritması eğitmek için MATLAB uygulaması kullanılmıştır.

ImageNet, kabaca 22.000 kategoriye ait 15 milyondan fazla etiketli yüksek çözünürlüklü görüntülerden oluşan veri kümesidir. Verisetinde bulunan resimler webden toplanmış ve Amazon'un Mechanical Turk crowd aracını kullanarak insan etiketleyicileri tarafından etiketlenmiştir. Pascal Visual Object Challenge'in bir parçası olarak 2010'dan itibaren, ImageNet Büyük Ölçekli Görsel Tanıma Yarışması (ImageNet Large-Scale Visual Recognition Challenge / ILSVRC) adı verilen yıllık bir yarışma gerçekleştirilmiştir. Gerçekleştirilen yarışmada, ImageNet in alt kümesi olarak her birinde 1000 resim bulunan 1000 kategorilik bir veri seti kullanılmıştır. Toplamda, kabaca 1.2 milyon eğitim resmi, 50.000 eğitim görüntüsü ve 150.000 test görüntüsü

vardır. ImageNet, deęişken çözünürlüklü görüntülerden oluşurken, kullanılan sistem sabit bir girdi boyutu gerektirmektedir. Bu nedenle görüntüler 227×227 piksellik çözünürlük deęerinde sabitlenmiştir. Görüntülere boyutlandırma dışında başka bir önışlem yapılmamıştır. Böylece aę, piksellerin ham RGB deęerleri üzerinden eğitilmiştir.

60 milyon parametre ve 650.000 nöron içeren sinir aęı, beş konvolüsyon katmanından oluşur; bunlardan bazılarını takip eden maksimum havuz katmanları, 3 tam baęlı katman ve 1000 softmax kullanılarak sonlandırılmıştır. Daha hızlı eğitim yapmak için doymuş olmayan (non saturating) nöronlar ve etkili konvolüsyon işlemi için çok verimli bir GPU uygulaması kullanılmıştır. Tam baęlanmış tabakalardaki aşırı uyumsuzluğu (overfitting) azaltmak için, sinir aęının sonunda tam baęlı katmana uygulanan son zamanlarda geliştirilen, "Dropout" adlı normalleştirme yöntemini kullanılmış ve bu yöntemin çok etkili olduęu kanıtlanmıştır (Krizhevsky vd. 2012). Bu katman yalnızca eğitim aşamasında kullanılmaktadır, test aşamasında kullanılmamaktadır. Çizelge 3.1'de uygulamada kullanılan, açıklaması yapılmış AlexNet derin sinir aęı mimarisi özetlenmiştir.

Çizelge 3.1 AlexNet mimarisi

Katman Adı	Yapısı
Girdi görüntüsü	RGB, 227×227 piksel
Conv1	Konvolüsyon katmanı, 11×11×3 boyutlarında 96 filtre, adım 4×4, dolgu 0×0
Relu1	ReLU katmanı
Norm1	Eleman başına 5 kanal ile çapraz kanal normalizasyonu
Pool1	3 × 3 max pooling operatörü, adım 1×1, dolgu 0×0
Conv2	Konvolüsyon katmanı, 5×5×48 boyutlarında 256 filtre, adım 1×1, dolgu 2×2
Relu2	ReLU katmanı
Norm2	Eleman başına 5 kanal ile çapraz kanal normalizasyonu
Pool2	3 × 3 max pooling operatörü, adım 2×2, dolgu 0×0
Conv3	Konvolüsyon katmanı, 3×3×256 boyutlarında 384 filtre, adım 1×1, dolgu 1×1
Relu3	ReLU katmanı
Conv4	Konvolüsyon katmanı, 3×3×192 boyutlarında 384 filtre, adım 1×1, dolgu 1×1
Relu4	ReLU katmanı
Conv5	Konvolüsyon katmanı, 3×3×192 boyutlarında 256 filtre, adım 1×1, dolgu 1×1
Relu5	ReLU katmanı
Pool5	3 × 3 max pooling operatörü, adım 2×2, dolgu 0×0
FC6	4096 tam bağlı nöron
FC7	4096 tam bağlı nöron
FC8	1000 tam bağlı nöron
Prob	Softmax fonksiyonu

- Önceden eğitim görmüş AlexNet mimarisi indirilir ve yüklenir.
- Ağ'ı Sinir Ağ Toolbox™ 'ı (Neural Network Toolbox™) kullanarak Seri Ağ (Series Network) olarak MATLAB'e aktarılır ve KSA mimarisini görüntülenir. Series Network nesnesi KSA'yı temsil etmektedir.
- Veri seti ağa eklenir. AlexNet ağı 227x227 piksellik ve renkli görüntüler üzerinde eğitildiğinden tüm eğitim görüntüleri aynı çözünürlüğe ayarlanır. AlexNet girişte 3 kanal görüntüsü gerektirmektedir. Lfw ve faces96 verisetleri zaten renkli görüntülerden oluşmaktadır ancak at&t veriseti gri renkli görüntülerden oluştuğu için tek kanalıdır, ağa tarafından işlenebilmesi için görüntü diğer iki kanala basitçe kopyalanmış ve üç kanallı bir görüntü elde edilmiştir. 2. 3. ve 5. Konvolüsyon katmanları ve tam bağlı katmanlarda bias değeri 1, diğer katmanlarda ise 0 olarak belirlenmiştir. Standart sapma değeri ve ilk öğrenme oranı 0.01 olarak belirlenmiştir.
- Resimler eğitim ve test verileri olmak üzere ikiye ayrılır ve resimler rastgele seçilir.
- Eğitim setinde bulunan görüntülerin özellikleri KSA üzerinden, önceden eğitim görmüş AlexNet mimarisi ile çıkarılır ve bu özellikler DVM sınıflandırıcısını eğitmek ve test etmek için kullanılır. Ağın başlangıcındaki katmanlar, kenarlar ve lekeler gibi temel görüntü özelliklerini yakalar. Bu ilkel (primitive) özellikler daha sonra, daha üst düzey görüntü özellikleri oluşturmak için önceki özellikleri birleştiren daha derin ağ katmanları tarafından işlenir. Bu üst düzey özellikler, tanıma görevleri için daha uygundur. Çünkü tüm ilkel(primitive) özellikleri daha zengin bir görüntü sunumuyla birleştirirler. Bir KSA' nın her katmanı bir girdi resmine (input image) yanıt üretirken, yalnızca birkaç katman öznetelik çıkarımı için uygundur. Bu katmanları tanımlamak için kesin bir formül yoktur, ancak genellikle sınıflama katmanının başlamasından hemen önceki katmanla

başlamak iyi bir yerdir. Bu çalışmada da bu katman 'fc7' olarak adlandırılır. Bu katmanı kullanarak eğitim özellikleri çıkarılmıştır.

- Çıkarılan özellikler eğitilmiş bire karşı hepsi tasarımını kullanan DVM sınıflandırıcısına gönderilir, sınıflandırıcının tahmin ettiği sonuç ile gerçek sonuç karşılaştırılır, iki sonuç aynı ise işlem doğru gerçekleştirilmiştir farklı ise yanlış gerçekleştirilmiştir.



4. BULGULAR

Bu uygulama Intel Core i5-7200U, 2.50 GHz CPU ve 12 GB RAM özelliklerine sahip standart bir dizüstü bilgisayarda geliştirilmiştir. Bu bölümde geleneksel makine öğrenme tekniği ve derin öğrenme tekniği kullanılarak eğitilen yüz tanıma algoritması başarı oranları sonuçları incelenecektir.

Geleneksel makine öğrenme tekniği kullanılarak eğitilen algoritma ile üç ayrı veri seti eğitilmiş ve test edilmiştir. Eğitim ve test veri setlerinde bulunan verilerin sayılarına göre başarı oranları karşılaştırılmıştır. Eğitim veri setindeki verilerin sayısı arttıkça başarı oranının arttığı görülmüştür. Sonuçlar şekil 4.1'de bulunan grafikte gösterilmiştir.

Derin öğrenme tekniği kullanılarak geliştirilen algoritmada, bir önceki yöntemde olduğu gibi üç farklı veri seti kullanılarak ağ eğitilmiş ve test edilmiştir. Derin öğrenme mimarisi olarak AlexNet mimarisi ile öznetelikler çıkarılmıştır. Bu yöntemde de eğitim veri setinde bulunan örneklerin sayısı arttıkça başarının arttığı gözlenmiştir. İki yöntemin başarı oranı karşılaştırıldığında derin öğrenme tekniği ile geliştirilen yöntemin üç veri setinde de daha başarılı sonuç verdiği görülmüştür. Bu bağlamda öznetelik çıkarımı aşamasında, derin öğrenmeyi diğer yöntemlerden de ayıran, kendi kendine öğrenebilmesi özelliğinin başarılı olduğu kanıtlanmıştır. İki farklı yöntem kullanılarak üç farklı veri seti üzerinde görülen başarı oranları ayrı ayrı şekil 4.4- 4.6'da gösterilmiştir.

At&t, faces96 ve lfw olmak üzere üç farklı veri setinde, eğitim ve test veri setinde bulunan verilerin sayılarına göre elde edilen başarı grafikleri aşağıdaki gibidir.

Çizelge 4.1 At&t veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi

Eğitim ve test verileri dağılım yüzdesi	Toplam test verisi sayısı	Doğru bilinen veri sayısı	Başarı yüzdesi
%60- %40	160	122	%76.25
%70- %30	120	96	%80
%80-%20	80	68	%85
%90-%10	40	34	%85

Çizelge 4.2 Faces96 veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi

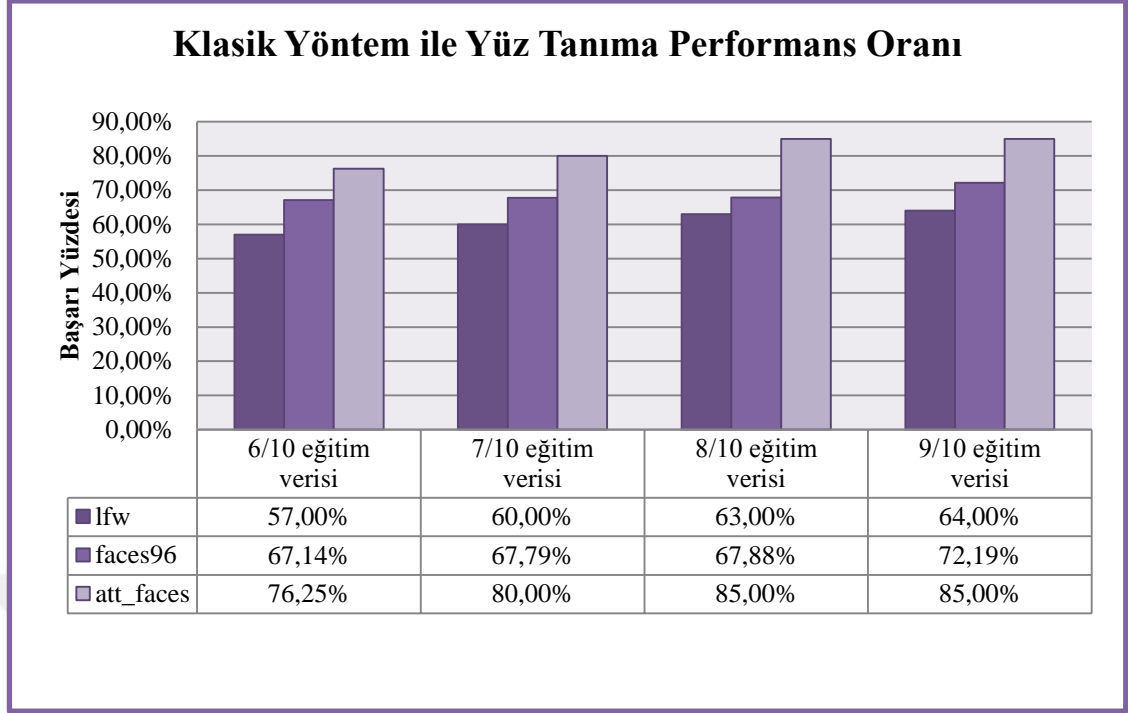
Eğitim ve test verileri dağılım yüzdesi	Toplam test verisi sayısı	Doğru bilinen veri sayısı	Başarı yüzdesi
%60- %40	1208	811	%67.14
%70- %30	906	587	%67.79
%80-%20	604	410	%67.88
%90-%10	302	218	%72.19

Çizelge 4.3 Lfw veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi

Eğitim ve test verileri dağılım yüzdesi	Toplam test verisi sayısı	Doğru bilinen veri sayısı	Başarı yüzdesi
%60- %40	1221	691	%57
%70- %30	923	553	%60
%80-%20	625	391	%63
%90-%10	331	213	%64

Klasik yöntem kullanılarak att_faces veri seti eğitimi yaklaşık olarak 5 dakika içinde, tamamlanmıştır. Faces_96 veri seti yaklaşık olarak 13 dakikada ve lfw veri seti 20 dakikada eğitimini tamamlamıştır. Derin öğrenme tekniği kullanıldığında ise bu süreler att_faces veri seti için 15 dakika, faces96 veriseti için 35 dakika, lfw veri seti için 60 dakika olarak ölçülmüştür.

Klasik yöntem kullanılarak eğitilen uygulamanın 3 veri seti üzerindeki başarı oranları grafiği şekil 4.1’de gösterilmiştir.



Şekil 4.1Klasik yöntem ile yüz tanıma performans oranı

Çizelge 4.4 At&t veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi

Eğitim ve test verileri dağılım yüzdesi	Toplam test verisi sayısı	Doğru bilinen veri sayısı	Başarı yüzdesi
% 60- %40	160	153	%95.63
% 70- %30	120	117	%97.50
% 80-%20	80	79	%98.75
% 90-%10	40	39	%97.50

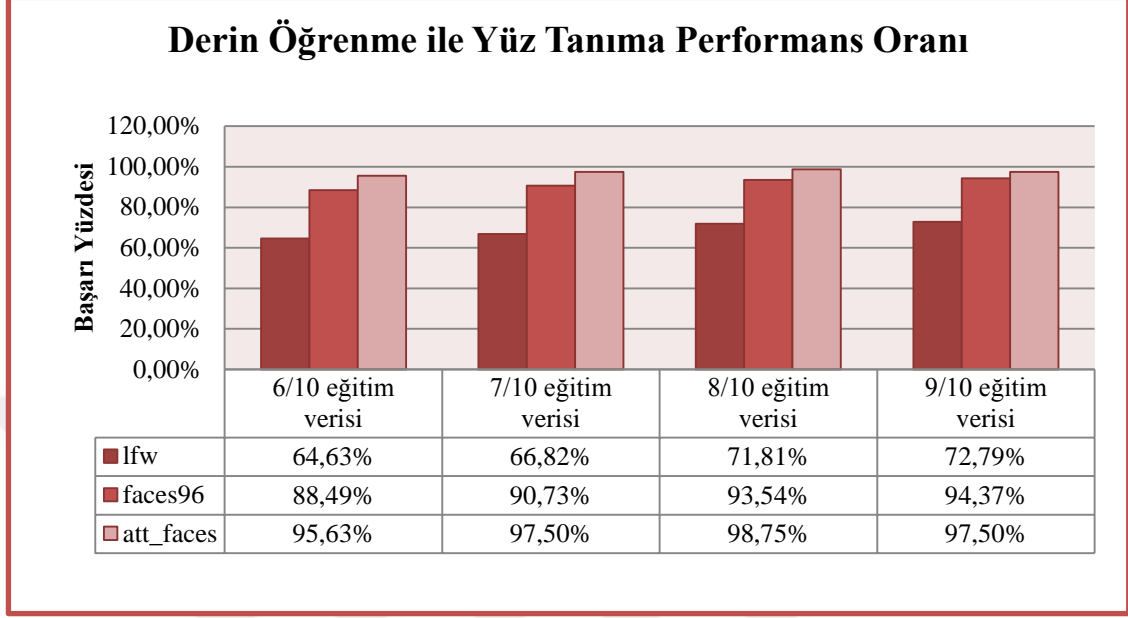
Çizelge 4.5 Faces96 veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi

Eğitim ve test verileri dağılım yüzdesi	Toplam test verisi sayısı	Doğru bilinen veri sayısı	Başarı yüzdesi
%60- %40	1208	1069	%88.49
%70- %30	906	822	%90.73
%80-%20	604	565	%93.54
%90-%10	302	285	%94.37

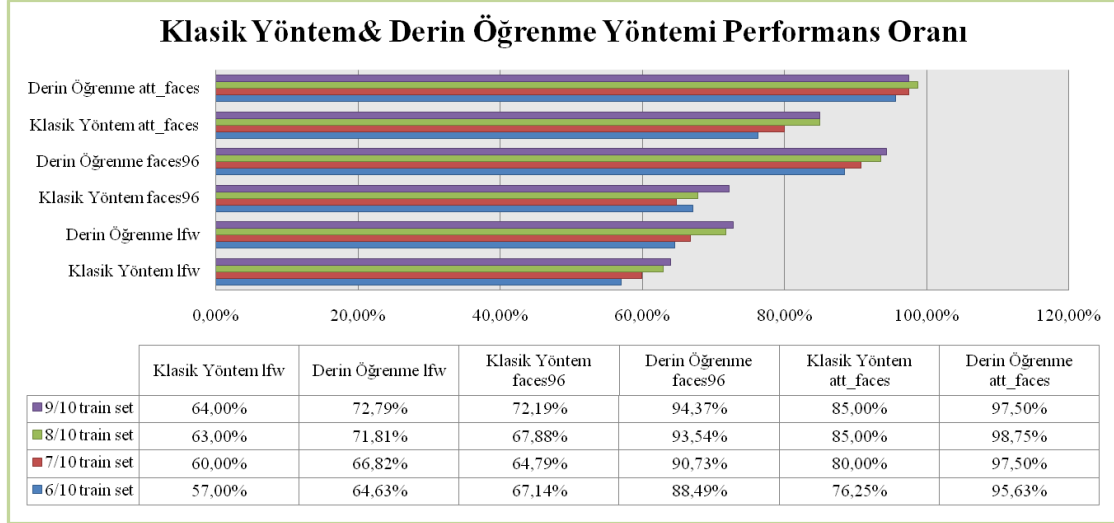
Çizelge 4.6 Lfw veri seti eğitim ve test verileri dağılımı ve başarı yüzdesi

Eğitim ve test verileri dağılım yüzdesi	Toplam test verisi sayısı	Doğru bilinen veri sayısı	Başarı yüzdesi
%60- %40	1196	762	%64.63
%70- %30	898	600	%66.82
%80-%20	596	428	%71.81
%90-%10	294	214	%72.79

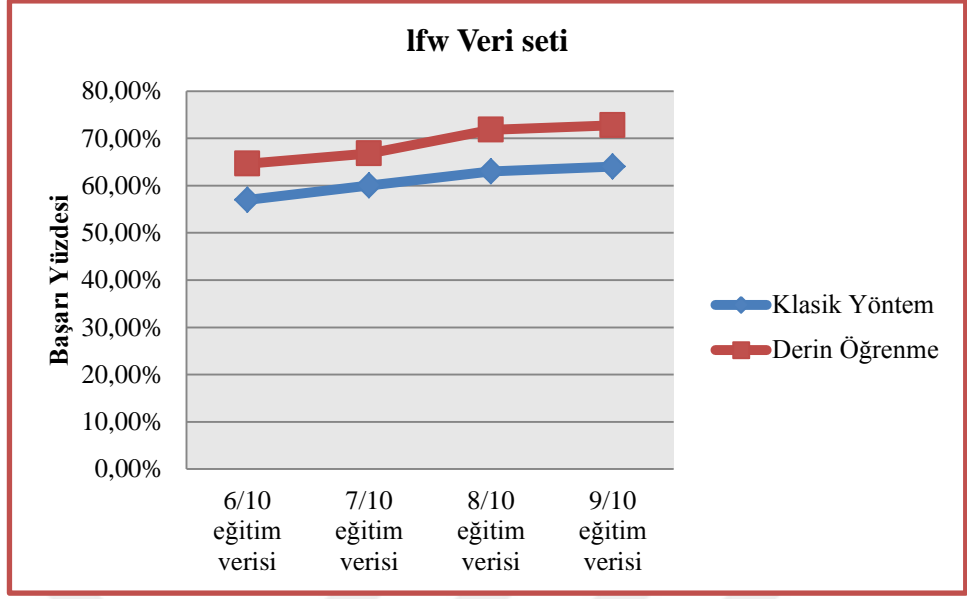
Derin öğrenme yöntemi kullanılarak eğitilen uygulamanın 3 veri seti üzerindeki başarı oranları grafiği şekil 4.2’de gösterilmiştir.



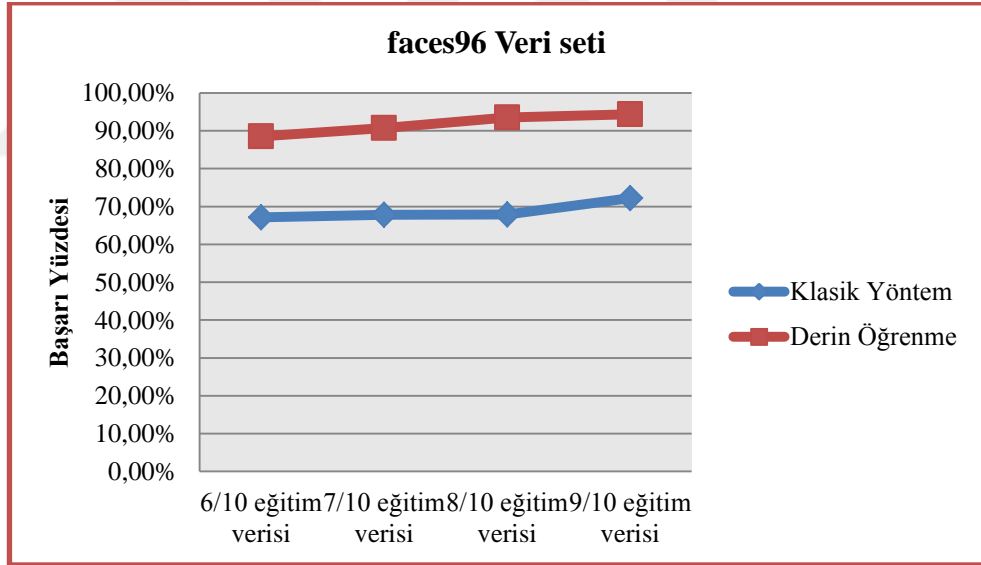
Şekil 4.2 Derin öğrenme ile yüz tanıma performans oranı



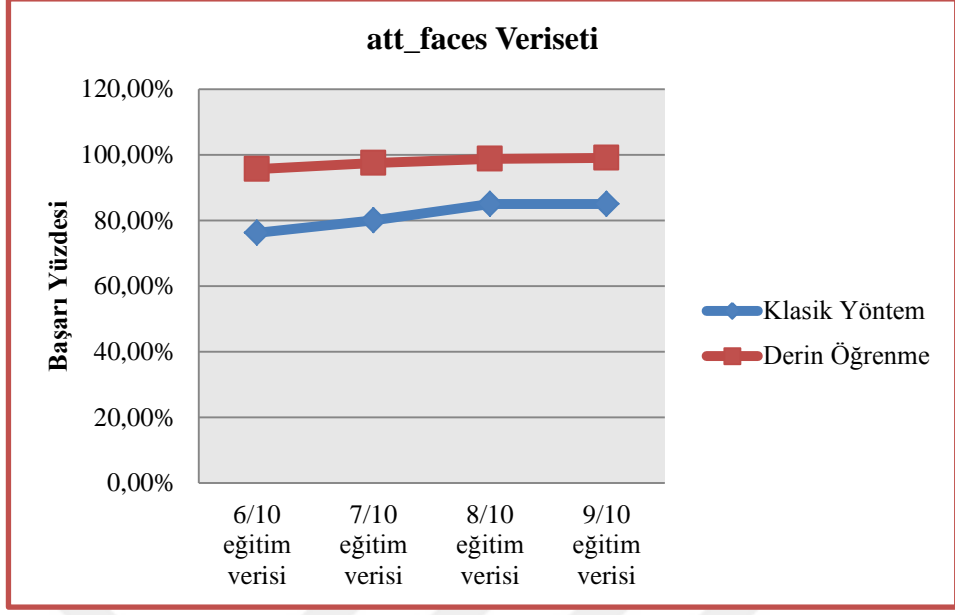
Şekil 4.3 Yöntemlerin başarı oranları karşılaştırılması



Şekil 4.4 lfw veri setinde iki yöntem karşılaştırılması



Şekil 4.5 faces96 veri setinde iki yöntem karşılaştırılması



Şekil 4.6 att_faces veri setinde iki yöntem karşılaştırılması

5. SONUÇ ve TARTIŞMA

Bu yüksek lisans tez çalışmasında resim üzerinden elde edilen yüz tanıma çalışmalarında gerçekleştirilen genel konsept, yüz tanıma aşamaları ve kullanılan yöntemlerin detaylı açıklaması ile derin öğrenme mimarisi, çalışma prensibi ile ilgili çalışmalardan bahsedilmiştir.

Tez kapsamında, geleneksel makine öğrenme tekniği ve derin öğrenme tekniği kullanılarak bir yüz tanıma algoritması eğitilmiş ve iki ayrı yöntemin başarı oranlarını karşılaştırılmıştır. Geleneksel makine öğrenme tekniği ile geliştirilen yüz tanıma algoritması kapsamında; HOG özellik çıkarıcı ile elde edilen özellikler, DVM sınıflandırıcısı kullanılarak sınıflandırılmış, derin öğrenme tekniği kullanılarak geliştirilen yüz tanıma algoritmasında ise KSA özellik çıkarıcı olarak kullanılmış ve elde edilen özellikler aynı şekilde DVM sınıflandırıcısı ile sınıflandırılarak tanıma işlemi gerçekleştirilmiştir. İki farklı yöntem kullanılarak geliştirilen yüz tanıma algoritmaları 3 farklı veri seti üzerinde test edildiğinde derin öğrenme tekniği kullanılarak geliştirilen algoritmanın diğer yöntemlere göre başarısının yüksek olduğu görülmüştür.

Öznitelik çıkarımı işlemi, geleneksel makine öğrenme tekniklerinde manuel olarak yapılan bir işlemdir ancak derin öğrenme algoritmaları kullanıldığında, algoritmaları özellik, işaret ve görevleri doğrudan görüntü, metin ve ses arasından kendi kendine öğrenebildiği için manuel özellik seçimi ihtiyacını ortadan kaldırmaktadır.

Derin öğrenmenin mevcut hesaplama ve verilerin miktarındaki artışlardan kolayca yararlanabilmesi ve elle çok az mühendislik gerektirmesi nedeniyle yakın gelecekte çok daha fazla başarıya sahip olacağı düşünülmektedir. Bu amaçla tez çalışmasında derin öğrenme tekniği seçilmiştir. Yapılan çalışmanın geliştirilmesi amacıyla önceden eğitim görmüş ağ yerine, farklı mimariye sahip bir ağ kurulup veri seti ile eğitilip sonuçlar karşılaştırılabilir.

KAYNAKLAR

- Albiol, A., Monzo, D., Martin, A., Sastre, J. and Albiol, A. 2008. Face recognition using HOG–EBGM. *Pattern Recognition Letters*, 29(10), 1537-1543.
- Anonymous 2016d. Web Sitesi: <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>, Erişim Tarihi: 17.06.2017
- Anonymous 2017a. Web Sitesi: <https://software.intel.com/en-us/node/529070>, Erişim Tarihi: 02.01.2017
- Anonymous 2017b. Web Sitesi: <http://www.psikolojik.gen.tr/yapay-sinir-aglari.html>, Erişim Tarihi: 22.06. 2017
- Anonymous 2017c. Web Sitesi: <http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/>, Erişim Tarihi: 22.06.2017
- Anonymous 2017e. Web Sitesi: <http://www.rsipvision.com/exploring-deep-learning/>, Erişim Tarihi: 15.03.2017
- Anonymous 2017f. Web Sitesi: <http://cs231n.github.io/convolutional-networks/>, Erişim Tarihi: 26.02.2017
- Anonymous 2017g. Web Sitesi: <https://www.analyticsvidhya.com/blog/2016/04/deep-learning-computer-vision-introduction-convolution-neural-networks/>, Erişim Tarihi: 01.07.2017
- Anonymous 2017h. Web Sitesi: <http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/>, Erişim Tarihi: 18.04.2017
- Anonymous 2017i. Web Sitesi: <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>, Erişim Tarihi: 12.03.2017
- Anonymous 2017j. Web Sitesi: <http://cswww.essex.ac.uk/mv/allfaces/faces96.html>, Erişim Tarihi: 12.03.2017
- Anonymous 2017k. Web Sitesi: <https://www.mathworks.com/company/newsletters/articles/deep-learning-for-computer-vision-with-matlab.html>, Erişim Tarihi: 08.04.2017
- Arel, I., Rose, D. C. and Karnowski, T. P. 2010. Deep machine learning-a new frontier in artificial intelligence research. *IEEE Computational Intelligence Magazine*, 5(4), 13-18.

- Ayhan, S. ve Erdoğan, Ş. 2014. Destek vektör makineleriyle sınıflandırma problemlerinin çözümü için çekirdek fonksiyonu seçimi. Eskişehir Osmangazi Üniversitesi İktisadi ve İdari Bilimler Dergisi, 9(1).
- Başer, F. ve Apaydın, A. 2015. Sınıflandırma Amaçlı Destek Vektör Makinelerinin Lojistik Regresyon İle Karşılaştırılması. Anadolu University of Sciences & Technology-B: Theoretical Sciences, 3(2).
- Bordes, A., Chopra, S. and Weston, J. 2014. Question answering with subgraph embeddings. arXiv preprint arXiv:1406.3676.
- Brunelli, R. and Poggio, T. 1993. Face recognition: Features versus templates. IEEE transactions on pattern analysis and machine intelligence, 15(10), 1042-1052.
- Campadelli, P. and Lanzarotti, R. 2005. A face recognition system based on local feature characterization. In Advanced Studies in Biometrics (pp. 147-152). Springer Berlin Heidelberg.
- Cheung, B. 2012. Convolutional neural networks applied to human face classification. In Machine Learning and Applications (ICMLA), 2012 11th International Conference on (Vol. 2, pp. 580-583). IEEE.
- Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. and Kuksa, P. 2011. Natural language processing (almost) from scratch. Journal of Machine Learning Research, 12(Aug), 2493-2537.
- Comon, P. 1994. Independent component analysis, a new concept. Signal processing, 36(3), 287-314.
- Cortes, C. and Vapnik, V. 1995. Support-vector networks. Machine learning, 20(3), 273-297.
- Dayhoff, J. E. and DeLeo, J. M. 2001. Artificial neural networks. Cancer, 91(S8), 1615-1635.
- De Carrera, P. F. and Marques, I. 2010. Face recognition algorithms. Master's thesis in Computer Science, Universidad Euskal Herriko.
- Deng, L. 2014. A tutorial survey of architectures, algorithms, and applications for deep learning. APSIPA Transactions on Signal and Information Processing, 3, e2.
- Déniz, O., Bueno, G., Salido, J., & De la Torre, F. 2011. Face recognition using histograms of oriented gradients. Pattern Recognition Letters, 32(12), 1598-1603.
- Déniz, O., Castrillon, M. and Hernández, M. 2001. Face Recognition Using Independent Component Analysis and Support Vector Machines. In International Conference

- on Audio-and Video-Based Biometric Person Authentication (pp. 59-64). Springer, June, Berlin Heidelberg.
- Ding, C. and Tao, D. 2016. A comprehensive survey on pose-invariant face recognition. *ACM Transactions on intelligent systems and technology (TIST)*, 7(3), 37.
- Farabet, C., Couprie, C., Najman, L. and LeCun, Y. 2013. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8), 1915-1929.
- Guo, G., Li, S. Z. and Chan, K. 2000. Face recognition by support vector machines. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on* (pp. 196-201). IEEE.
- Günes, A. ve Yiğit, T. 2012, April. Handwritten digit recognition with accelerated Support Vector Machines. In *Signal Processing and Communications Applications Conference (SIU), 2012 20th* (pp. 1-4). IEEE.
- Heisele, B., Ho, P. and Poggio, T. 2001. Face recognition with support vector machines: Global versus component-based approach. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on* (Vol. 2, pp. 688-694). IEEE.
- Heisele, B., Ho, P., & Poggio, T. 2001. Face recognition with support vector machines: Global versus component-based approach. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on* (Vol. 2, pp. 688-694). IEEE.
- Hu, G., Yang, Y., Yi, D., Kittler, J., Christmas, W., Li, S. Z. and Hospedales, T. 2015. When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (pp. 142-150).
- Ilonen, J., Kämäräinen, J. K., & Kälviäinen, H. 2005. Efficient computation of Gabor features. Lappeenranta University of Technology.
- Jacobs, D. 2005. Correlation and convolution. *Class Notes for CMSC*, 426.
- Jafri, R. and Arabnia, H. R. 2009. A survey of face recognition techniques. *Jips*, 5(2), 41-68.
- Jonsson, K., Matas, J., Kittler, J. and Li, Y. P. 2000. Learning support vectors for face verification and recognition. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on* (pp. 208-213). IEEE.
- Kalchbrenner, N., Grefenstette, E. and Blunsom, P. 2014. A Convolutional Neural Network for Modelling Sentences. *Acl*, 655–665.

- Keleşoğlu, Ö. ve Fırat, A. 2006. Tuğla Duvardaki ve Tesisattaki Isı Kaybının Yapay Sinir Ağları İle Belirlenmesi. *Fırat Üniv. Fen ve Müh. Bil. Der.*, 18(1), 133-141.
- Kirby, M. and 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.
- Krizhevsky, A., Sutskever, I. and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).
- LeCun, Y., Bengio, Y. and Hinton, G. 2015. Deep learning. *Nature*, 521(7553), 436-444.
- LeCun, Y., Kavukcuoglu, K. and Farabet, C. 2010, May. Convolutional networks and applications in vision. In *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on* (pp. 253-256). IEEE.
- LeCun, Y. and Ranzato, M. (2013, June). Deep learning tutorial. In *Tutorials in International Conference on Machine Learning (ICML'13)*.
- Lu, J., Plataniotis, K. N. and Venetsanopoulos, A. N. 2003. Face recognition using kernel direct discriminant analysis algorithms. *IEEE Transactions on Neural Networks*, 14(1), 117-126.
- Mikolov, T., Deoras, A., Povey, D., Burget, L. and Černocký, J. 2011, December. Strategies for training large scale neural network language models. In *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on* (pp. 196-201). IEEE.
- Nefian, A. and Hayes III, M. H. 1999. A hidden Markov model-based approach for face detection and recognition (Doctoral dissertation, School of Electrical and Computer Engineering, Georgia Institute of Technology).
- O'Shea, K. and Nash, R. 2015. An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.
- Öztemel, E. 2003. Yapay Sinir Ağları. PapatyaYayincilik, Istanbul.
- Pervaiz, A. Z. 2010, March. Real time face recognition system based on EBGM framework. In *Computer Modelling and Simulation (UKSim), 2010 12th International Conference on* (pp. 262-266). IEEE.
- Phillips, P. J. 1998. Support vector machines applied to face recognition.
- Samaria, F. S. and Harter, A. C. 1994, December. Parameterisation of a stochastic model for human face identification. In *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on* (pp. 138-142). IEEE.

- Schmidhuber, J. 2015. Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117.
- Schroff, F., Kalenichenko, D. and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 815-823).
- Sharif, M., Mohsin, S. and Javed, M. Y. 2012. A survey: face recognition techniques. *Research Journal of Applied Sciences, Engineering and Technology*, 4(23), 4979-4990.
- Sun, Y., Chen, Y., Wang, X. and Tang, X. 2014. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems* (pp. 1988-1996).
- Sun, Y., Wang, X. and Tang, X. 2014. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1891-1898).
- Syafeeza, A.R., et al. 2016. A MATLAB-Based Convolutional Neural Network Approach for Face Recognition System *J Bioinfo Proteomics Rev* 2(1): 1-5
- Taigman, Y., Yang, M., Ranzato, M. A. and Wolf, L. 2014. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1701-1708).
- Tolba, A. S., El-Baz, A. H. and El-Harby, A. A. 2006. Face recognition: A literature review. *International Journal of Signal Processing*, 2(2), 88-103.
- Tompson, J. J., Jain, A., LeCun, Y. and Bregler, C. 2014. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems* (pp. 1799-1807).
- Turk, M. and Pentland, A. 1991. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1), 71-86.
- Varol, A. ve Cebe, B. 2011, September. Yüz tanıma algoritmaları. In *5th International Computer&Instructional Technologies Symposium*.
- Vijayakumari, V. 2013. Face recognition techniques: A survey. *World journal of computer application and technology*, 1(2), 41-50.
- Vilardi, A. L. 2016. Very deep convolutional neural networks for face identification (Master's thesis, Universitat Politècnica de Catalunya).
- Viola, P. and Jones, M. J. 2004. Robust real-time face detection. *International journal of computer vision*, 57(2), 137-154.

Wang, Y. Q. 2014. An analysis of the Viola-Jones face detection algorithm. *Image Processing On Line*, 4, 128-148.

Wang, Z. and Xue, X. 2014. Multi-class support vector machine. In *Support Vector Machines Applications* (pp. 23-48). Springer International Publishing.

Wiskott, L., Krüger, N., Kuiger, N. and Von Der Malsburg, C. 1997. Face recognition by elastic bunch graph matching. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7), 775-779.

Wu, J. 2016. *Introduction to Convolutional Neural Networks*. 27 April, China.



ÖZGEÇMİŞ

Adı Soyadı : Fatıma Zehra ÜNAL

Doğum Yeri : Ankara

Doğum Tarihi : 02.04.1989

Medeni Hali : Evli

Yabancı Dili : İngilizce

Eğitim Durumu

Lise : Kalaba Anadolu Lisesi (2007)

Lisans : Çankaya Üniversitesi Bilgisayar Mühendisliği, Çift Anadal (2013)
Çankaya Üniversitesi Matematik- Bilgisayar Bölümü, Anadal
(2013)

Yüksek Lisans : Ankara Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Bölümü (Eylül 2014-)

Çalıştığı Kurumlar

Ankara Ticaret Odası (2014- 2016)