

**ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

İNSANIN YÜZ MİMİKLERİNDEN DURUM ANALİZİ

Büşra EMEK SOYLU

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**ANKARA
2017**

Her hakkı saklıdır

TEZ ONAYI

Büşra EMEK SOYLU tarafından hazırlanan “İnsanın Yüz Mimiklerinden Durum Analizi Yapma” adlı tez çalışması 28/07/2017 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. Hacer YALIM KELEŞ

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.

28/07/2017



Büşra EMEK SOYLU

ÖZET

Yüksek Lisans Tezi

İNSANIN YÜZ MİMİKLERİNDEN DURUM ANALİZİ YAPMA

Büşra EMEK SOYLU

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Yrd. Doç. Dr. Mehmet Serdar GÜZEL

Yüz mimikleri, kişiler arası ilişkilerde önemli rol oynayan, sözlü olmayan evrensel işaretlerdir. İnsanlar konuşmalarının belirli bir bölümünü vurgulamak ve duygularını göstermek için mimiklerini kullanırlar. İnsanoğlu bu mimikleri kavramada herhangi bir zorluk çekmezken, bu durum makineler için geçerli olmayıp, halen daha başarılı ve daha güvenilir mimik tanıma sistemleri üzerinde çeşitli araştırmalar yapılmaktadır.

Yapılan tezin amacı, insana ait çeşitli yüz resimlerinden mimik durum analizi yapmaktır. Yüz mimikleri için ‘mutlu’, ‘üzgün’, ‘korkmuş’, ‘sinirli’, ‘şaşkın’, ‘iğrenmiş’, nötr’ olmak üzere yedi adet evrensel mimik kullanılmıştır. Yapılan çalışma dört ana bölüm olarak incelenebilir:

1. Resmin ön işlemlerden geçirilmesi: Yüz mimik analizinde görüntülerin bilgisayar tarafından işlenebilecek sayısal veriler haline dönüştürülmesi gerekmektedir. Sayısal verilerin anlamlandırılması için bu sayısal veriler görüntü işlemenin temel adımlarından geçirilmiştir.
2. Yüz tespiti: Analiz yapılacak resimden sadece yüzün tespit edilip, diğer faktörlerin (örneğin, saç, boyun vs.) çıkarılması çalışma zamanını olumlu etkileyecektir. Viola ve Jones algoritması kullanılarak resimden yüz tespit edilmiş, dikdörtgen ile yüz kırpılmıştır.
3. Geliştirilen birinci yaklaşımda HOG yöntemi kullanılarak öznitelikler çıkarılmış, DVM ile sınıflandırma yapılmıştır.
4. Geliştirilen ikinci yaklaşımda Derin öğrenmenin özelleşmiş bir modeli olan ‘Konvolüsyonel Sinir Ağları’ ile öznitelikler çıkarılıp, bu öznitelikler yedi mimik sınıfından birine sınıflandırılmıştır. Daha sonra bu iki yaklaşım karşılaştırılmıştır.

Temmuz 2017, 93 sayfa

Anahtar Kelimeler: Viola ve Jones, HOG, DVM, derin öğrenme, öznitelik.

ABSTRACT

Master Thesis

MAKİNG SITUATION ANALYSIS FROM HUMAN FACİAL GESTURES

Büşra EMEK SOYLU

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

Supervisor: Asst. Prof. Dr. Mehmet Serdar GÜZEL

Facial gestures are universal, non-verbal signs play an important role in interpersonal relationships. People use gestures to emphasize a specific part of their speech and to show their emotions. While human beings do not have any difficulty in the concept of these gestures, this does not apply to machines, but various researches are still being carried out on more successful and more reliable mimic recognition systems.

The aim of the thesis is to detect facial gestures from various face pictures belonging to human. Seven different mimic classes were defined for facial gestures: ‘happy’, ‘sad’, ‘frightened’, ‘angry’, ‘surprised’, ‘disgusted’, ‘neutral’. The work to be done can be examined in four main sections:

1. Pre-processing of face pictures: In face gesture analysis, images must be transformed into numerical data that can be processed by computer. For the interpretation of numerical data, these numerical data must be passed through the basic steps of image processing.
2. Face detection: Determining only the face from the image to be analyzed and removing other factors (e.g. hair, neck, etc.) will seriously affect the working time. Viola and Jones algorithm was used to detect the face from the pictures and crop the face with a rectangle.
3. In the first approach to be developed, features were extracted by using ‘Histogram of Oriented Gradients’ method and classified with ‘Support Vector Machines’.
4. The second approach to be developed is to extract the features with the Deep Learning Model ‘Convolutional Neural Networks’ and classify them into one of seven mimic classes. These two approaches were compared.

July 2017, 93 pages

Key Words: Viola and Jones, HOG, SVM, deep learning, feature.

TEŞEKKÜR

Bu çalışmamın yürütülmesi aşamasında; değerli bilgi, öneri ve yardımlarını esirgemeyerek bana yol gösteren danışman hocam Sayın Yrd. Doç. Dr. Mehmet Serdar GÜZEL (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) başta olmak üzere çalışma süresince tüm zorlukları benimle göğüsleyen ve hayatımın her evresinde bana destek olan değerli eşim Kazım SOYLU'ya, manevi desteklerini esirgemeyen annem, babam ve kardeşlerime sonsuz teşekkürlerimi sunarım.

Ayrıca çalışmam boyunca her an yanımda olan değerli dostlarım Merve TOPAL, Fatıma Zehra ÜNAL, Işıl KARABEY, tüm mesai arkadaşlarım ve dostlarıma minnettarlığımı belirterek, en derin duygularla teşekkür ederim.

Büşra EMEK SOYLU

Ankara, Temmuz 2017

İÇİNDEKİLER

TEZ ONAY SAYFASI	
ETİK.....	i
ÖZET	ii
ABSTRACT	iii
TEŞEKKÜR	iv
KISALTMALAR DİZİNİ	vii
ŞEKİLLER DİZİNİ	viii
ÇİZELGELER DİZİNİ	x
1. GİRİŞ	1
1.1 Yüz Mimik Analizinin Tarihçesi.....	2
1.2 Önerilen Yaklaşım	3
1.3 Yüz Mimik Tanıma Sisteminde Zorluklar	4
1.4 Genel Çerçeve	5
2. KURAMSAL TEMELLER ve KAYNAK ÖZETLERİ.....	7
2.1 Makine Öğrenmesi	8
2.1 Geleneksel Yöntemler	10
2.1.1 Yerel/Öznitelik tabanlı yaklaşımlar	10
2.1.2 Bütüncül/Görünüm tabanlı yaklaşımlar	12
2.1.3 Hareket tabanlı yaklaşımlar	14
2.1.4 Model tabanlı yaklaşımlar.....	15
2.1.5 Hibrit tabanlı yaklaşımlar	17
2.2 Yapay Sinir Ağları	19
2.2.1 Derin öğrenme tabanlı yaklaşımlar	25
2.3 Sınıflandırma	33
2.3.1 Prototipik mimikler	33
2.3.2 Eylem birimlerini kullanma	33
3. GELİŞTİRİLEN YÖNTEMLER.....	35
3.1 Yüz Tespiti	37
3.1.1 Viola ve Jones algoritması	38
3.2 Ön İşlemler	42
3.2.1 Gri tonlamalı görüntüye dönüştürme	43
3.2.2 Yeniden boyutlandırma	43
3.3 HOG-SVM Tabanlı Yüz Mimik Tanıma Sistemi.....	44
3.3.1 HOG özniteliklerinin kullanımı	44
3.3.2 DVM ile sınıflandırma	48
3.4 Derin Öğrenme ile Yüz Mimikleri Durum Analizi	56
3.4.1 Konvolüsyonel sinir ağları.....	56
3.4.2 Geliştirilen KSA mimarisi	66
4. ARAŞTIRMA BULGULARI ve DENEYSEL SONUÇLAR	75
4.1 Uygulama Platformu.....	75
4.2 Veri Kümeleri.....	76
4.2.1 JAFFE veri seti.....	76
4.2.2 KDEF veri seti	77
4.2.3 MUG veri seti.....	78
4.3 Geliştirilen İki Yaklaşımın Karşılaştırılması	78

4.4 KSA Değişen Parametre Değerlerine Göre Elde Edilen Grafikler	79
4.4.1 Epok sayısı başarı oranı	80
4.4.2 KSA kayıp fonksiyonu ve iterasyon arasındaki ilişki	81
4.5 KSA Hata Matrisleri.....	82
5. SONUÇ.....	86
5.1 Öneriler.....	86
KAYNAKLAR	87
ÖZGEÇMİŞ.....	93



KISALTMALAR DİZİNİ

2B	İki boyutlu
3B	Üç boyutlu
AGM	Aktif Görünüm Modeli
AU	Action Units
BN	Batch Normalization
CK	Cohn Kanade
CLBP	Compound Local Binary Pattern
DBM	Derin Boltzmann Makineleri
DBN	Deep Belief Network
DİN	Derin İnanç Ağları
DSA	Derin Sinir Ağları
DVM	Destek Vektör Makinesi
FACS	Facial Action Coding System
FC	Fully Connected
HLAC	Higher Order Local Autocorrelation
HOG	Histogram of Oriented Gradients
KBM	Kısıtlı Boltzmann Makineleri
KSA	Konvolüsyonel Sinir Ağları
LBP	Local Binary Pattern
LDA	Lineer Diskriminant Analizi
PCA	Principal Component Analysis
PHOG	Pyramid of Histogram of Gradients
ReLU	Rectified Linear Unit
SGD	Stokastik Gradyan Düşüşü
SVM	Support Vector Machine
TBA	Temel Bileşen Analizi
U-LTP	Uniform Local Ternary Pattern
YMA	Yüz Mimik Analizi
YSA	Yapay Sinir Ağları

ŞEKİLLER DİZİNİ

Şekil 1.1 Bir görüntünün farklı pozlardaki hali (Khatrı vd. 2014)	4
Şekil 1.2 Engellenmiş görüntüler	5
Şekil 1.3 Bir görüntünün farklı ışıklandırılması	5
Şekil 2.1 Yüz mimikleri durum analizi yaklaşımları	7
Şekil 2.2 Facial Affect veritabanındaki resimler üzerinde önemli yüz noktalarının ve bu nokta kümelerinin merkezlerinin işaretlenmesi (Gizatdinova ve Surakka 2006)	11
Şekil 2.3 Yüz mimikleri optik akış hesaplaması	15
Şekil 2.4 Candide modeli	17
Şekil 2.5 Tipik bir yapay sinir ağı yapısı	19
Şekil 2.6 Geri beslemeli sinir ağı	22
Şekil 2.7 Perseptron yapısı	23
Şekil 2.8 Çok katmanlı perseptron yapısı	24
Şekil 2.9 Basit bir otomatik kodlayıcı yapısı	27
Şekil 2.10 Boltzmann makine yapısı	27
Şekil 2.12 FACS eylem birimlerinin bazı örnek kombinasyonları	34
Şekil 3.1 Geliştirilen yaklaşımın genel bir akış şeması	36
Şekil 3.2 Yüz tespiti yapılmış görüntülere birkaç örnek (Rowley vd. 1998)	37
Şekil 3.3 Öznitelik çıkarmak için kullanılan dikdörtgen çeşitleri (Viola ve Jones 2001)	39
Şekil 3.4 İntegral görüntünün oluşturulma esası (Viola ve Jones 2001)	40
Şekil 3.5 AdaBoost tarafından seçilen birinci ve ikinci öznitelikler (Viola ve Jones 2001)	41
Şekil 3.6 Algılama kademesinin şematik tasviri (Viola ve Jones 2001)	42
Şekil 3.7 HOG öznitelik seviyeleri	46
Şekil 3.8 Yönelim histogramları (Gill 2013)	47
Şekil 3.9 Her bir hücreden çıkarılan yerel HOG'ları birleştiren global bir HOG (Dahmane 2011)	48
Şekil 3.10 İki boyutlu bir uzayda ayrılabilir problem örneği (Cortes ve Vapnik 1995)	50
Şekil 3.11 Çekirdek fonksiyonları ile verinin daha yüksek bir boyuta dönüştürülmesi (Dima 2016)	51
Şekil 3.12 Bire karşı hepsi ve bire karşı bir yaklaşımlarının sınıflandırılmayan bölgeleri (Liu vd. 2010)	54
Şekil 3.13 İki sınıflı sınıflandırıcı akış diyagramı	55
Şekil 3.14 Çok sınıflı sınıflandırıcı akış diyagramı	55
Şekil 3.15 Konvolüsyon işlemi ile elde edilen öznitelikler	58
Şekil 3.16 ReLu fonksiyonu	62
Şekil 3.17 Havuzlama işlemi	63
Şekil 3.18 Geliştirilen KSA mimarisi	67
Şekil 3.19 Stokastik gradyan düşüşü	73
Şekil 4.1 JAFFE veritabanından örnek resimler	77
Şekil 4.2 KDEF veritabanından örnek resimler	77
Şekil 4.3 MUG veritabanından örnek resimler	78
Şekil 4.4 HOG-SVM bazlı yaklaşımın 3 veritabanı için başarı oranları	79

Şekil 4.5 Derin öğrenme tabanlı yaklaşımın 3 veritabanı için başarı oranları	80
Şekil 4.6 HOG-SVM ve Derin öğrenme tabanlı KSA yaklaşımlarının karşılaştırılması	80
Şekil 4.7 JAFFE veri seti için epok ve başarı oranı grafiği	81
Şekil 4.8 KDEF veri seti için epok ve başarı oranı grafiği	81
Şekil 4.9 MUG veri seti için epok ve başarı oranı grafiği	82
Şekil 4.10 JAFFE veri setinin kayıp fonksiyon-iterasyon grafiği.....	82
Şekil 4.11 KDEF veri setinin kayıp fonksiyon-iterasyon grafiği.....	83
Şekil 4.12 MUG veri setinin kayıp fonksiyon-iterasyon grafiği.....	84
Şekil 4.13 JAFFE mimik tahmin başarı oranı	85



ÇİZELGELER DİZİNİ

Çizelge 3.1 Test edilen KSA mimarileri	68
Çizelge 3.2 Her üç veritabanı için kullanılan hiperparametreler	69
Çizelge 4.1 Geliştirilen yaklaşımların yaklaşık eğitim süreleri	75
Çizelge 4.1 JAFFE veritabanının hata matrisi	83
Çizelge 4.2 KDEF veritabanının hata matrisi	84
Çizelge 4.3 MUG veritabanının hata matrisi	85



1. GİRİŞ

Mimikler, aksiyel kasların kasılması ve göz kapakları, kaşlar, burun, dudak ve ten dokusu gibi yüze ait özniteliklerin geçici olarak deforme olmasıyla çıkar. Kas aktivitelerine ilişkin bu tipik değişiklikler birkaç saniye sürer.

Yüz mimikleri iletişim sırasında bir kişinin psikolojik durumunu anlamadaki en belirleyici, güçlü, doğal ve sözlü olmayan evrensel işaretlerdir. Psikolog Mehrabian'e (1968) göre konuşulan sözcüklerin anlamının yani mesajın bir bütün olarak konuşmaya oranı yalnızca % 7, ses tonlamasının etkisi % 38'dir. Buna karşılık yüz mimikleri etkili iletişime %55 oranında katkıda bulunur. Bu da mimiklerin insan iletişiminde önemli bir rol oynadığını göstermektedir.

Teknolojinin gelişimiyle birlikte, bilgisayarlar yaşam ve çalışma alanlarının vazgeçilmez bir parçası haline gelmeye başlamış ve daha çok insanla daha fazla etkileşime geçmiştir. Bu yüzden insanlarla etkin bir şekilde etkileşim kurmak için, bilgisayarların da yüz mimiklerini tanıma kabiliyeti kazanması beklenmiştir. Bilim adamları farklı, karmaşık koşullarda bile mimikleri kolayca tanıyabilecek kadar sofistike olan insan beynini taklit ederek 'Yüz Mimikleri Durum Analiz' teknikleri geliştirmişlerdir. Yüz mimik analizinin amacı; yüz hareketlerini ve yüze ait öznitelik değişikliklerini görsel bilgilerden otomatik olarak analiz etmek ve tanımadır (Zeng vd. 2009). Nihayetinde günümüzde geline nokta yüz mimik analizinin amacı; görüntü işleme ve insan-bilgisayar etkileşimi (Pantic ve Rothkrantz 2003) alanında doğal etkileşimleri gerçekleştirecek, minimum hata verecek bir sistem geliştirmektir. Bu alanda eğer bilgisayar kullanıcıların mimiklerini algılayabiliyorsa ve bu mimiklerin yardımıyla niyetlerini anlayabiliyorsa, sistemin kullanıcılara doğru yanıt verebilmesi ve öneriler doğrultusunda yardımcı olması kolaylaşır.

Yüz mimikleri durum analizi, geniş uygulama olanakları nedeniyle bilgisayarda görme ve makine öğrenme alanında çok aktif ve açık araştırma alanıdır. Mimik tanımaya olan ilginin birçok sebebi vardır. Bunlardan en önemlileri; yapay zekâ ve örüntü tanımanın

gelişimi, yüz tespiti, yüz izleme, yüz tanıma gibi ilgili araştırma alanlarında yapılan ilerlemeler ve nispeten ucuz hesaplama gücünün mevcut olmasıdır.

Akıllı bir makinenin insan mimiklerini tanıma kabiliyeti, bir kişinin diğerine karşı yansıttığı benzer niyetleri daha kolay anlama, yorumlama, bunlara en hatasız şekilde cevap verme yetisine sahip olması demektir. Bu, insan bilgisayar iletişiminde; öğrenme ortamı, eğlence, müşteri hizmetleri, bilgisayar oyunları, güvenlik, eğitici yazılımlar gibi uygulamalara katkıda bulunur. Yüz mimikleri tanıma sistemlerinin animasyonlar, psikiyatri, eğitim yazılımları, hassas müzik, tıp bilimi, adli tıp, cezai mülakat, kolluk kuvvetleri, otomatik özel ders sistemleri vb. gibi birçok farklı alanda uygulamaları bulunmaktadır.

1.1 Yüz Mimik Analizinin Tarihçesi

Yüz mimik analizinin temeli on dokuzuncu yüzyılda atılmıştır. Darwin (1872) eserinde, tüm insanların ve hatta hayvanların da, duygularını benzer davranışlarla belli ettiğini iddia etmiştir. Bu eser için beraber çalıştığı bilim insanı Fransız Doktor Duchenne, insan yüzünün 60'tan fazla duyguyu iletebildiğini ve her birinin ayrı bir kas seti gerektirdiğini söylemiştir. Darwin ise yüz kaslarının beraber çalışarak sadece birkaç ana duyguyu dışa vurduğunu savunmuştur. Duchenne, savını kanıtlamak için katılımcılarına, yüzlerinde sabit bir kasılma yaratacak kadar elektrik akımı vermiş ve belirli kasları kasarak 'özgün' olduğunu düşündüğü 60 duygu ifadesi yaratmış ve bunları fotoğraflamıştır. Darwin ise, kendi evinde yaptığı bir deneyle bu ifadelerin insanlar tarafından algılanıp algılanmadığını ölçmeye girişmiştir. İnsanlar, elektrik akımı yardımıyla yaratılmış duygular içinden sadece evrensel ve ortak olan 6 duyguyu tanıyabilmişlerdir. Geri kalan fotoğraflardaki ifadeler çoğu kişi tarafından anlam verilememiş ya da farklı yorumlar yapılmıştır (<http://www.uplifers.com> 2017). Amerikan Psikolog Ekman ve Friesen (1971) altı temel duyguyu tanımlamışlardır: mutluluk, üzüntü, korku, tiksinti, sürpriz ve öfke. Ekman (1978) Darwin'in ve Duchenne'in araştırmalarını temel aldığı, yüzdeki mikro ifadelerle dayanarak insan duygularını okumaya yönelik kodlu yaklaşım; Yüz Hareket Kodlama Sistemi (FACS) adını verdiği çalışmasını sunmuştur. Bu sistem yüz hareketlerini tanımlamak için yüzü

bir dizi Eylem Birimi (AU) olarak bölmüştür. Suwa vd. (1978) bir görüntü dizisinden otomatik mimik analizine ilişkin bir ön araştırma sunmuşlardır. Mase ve Pentland (1991), FACS hareketini belirlemek için 8 yönlü optik akış değişimi kullanmışlardır. 2005 yıllarına kadar çoğu mimik tanıma 2 boyutlu statik görüntülere dayanmaktaydı. Qin ve Zeng-fu (2005) Bilim ve Teknoloji Üniversitesi bilim insanları, mimik tanıma yöntemini temel alan bir 3 boyutlu yüz modelini öne sürmüşlerdir. Bahsedilen bu çalışmalar yüz mimik analizi için dönüm noktaları olmuştur. Günümüze kadar yüz mimik analizi için çeşitli yöntemler uygulanarak geliştirilmiştir (Bkz. Bölüm 2).

1.2 Önerilen Yaklaşım

Son yıllarda, insan-bilgisayar etkileşim sistemlerinin (Pantic ve Rothkrantz 2003) tasarımı yüz mimiklerinin önemi nedeniyle, farklı yaklaşımlar öne sürülmüş, birçok makine öğrenme algoritması geliştirilmiştir. Bu tez kapsamında, insanın içinde bulunabileceği ‘mutlu’, ‘üzgün’, ‘sinirli’, ‘korkmuş’, ‘tiksinmiş’, ‘şaşkın’ ve ‘nötr’ olmak üzere 7 adet mimik durumununiki tür analizi gerçekleştirilecek ve bu iki yöntem karşılaştırılacaktır. Bu yaklaşımlardan önce ön işlemler yapılarak görüntüler işlem yapılmaya hazır hale getirilecektir. İlk olarak Viola ve Jones algoritması ile üzerinde çalışılacak yüz resimlerinden sadece yüz bölgesi çıkarılacaktır. Daha sonra elde edilen görüntüler bir sonraki adımların daha hızlı ve başarılı olması için normalleştirilecektir. Bu amaç doğrultusunda yapılan ilk analiz sistemi HOG-DVM tabanlı yüz mimik durum analizidir. Her bir mimik için alınan bu sabit görüntülerin öznitelikleri klasik bir öznitelik çıkarma yöntemi olan Histogram of Oriented Gradients (HOG) kullanılarak mimikleri temsil eden en iyi öznitelik kümeleri oluşturulacaktır. Temel örüntü işleme aşamalarından özellik çıkarma işlemi, tüm örüntülerin belirli sınıflandırıcılara tabi tutulmadan önce karşı karşıya kaldığı bir işlemdir ve doğru yapılması çalışmanın başarı oranını doğrudan etkileyecektir. Seçilen öznitelik kümeleri öznitelik seçme algoritmalarından yine klasik bir yöntem olan Destek Vektör Makineleri (DVM) ile 7 adet mimik sınıfından birine sınıflandırılacaktır.

Geliştirilecek ikinci sistem yakın zamanda oldukça sık kullanılan derin öğrenme ile mimik durum analizi gerçekleştirilecektir. Bu yöntemde derin öğrenmenin özelleşmiş

bir yapısı olan Konvolüsyonel Sinir Ağları (KSA) ile öznitelik çıkarımı yapıp yine bu ağlarla sınıflandırma yapılacaktır. Sonuçta geliştirilecek bu iki sistem ile bir başarı oranı çıkarılıp, bu iki yöntem karşılaştırılacaktır.

1.3 Yüz Mimik Tanıma Sisteminde Zorluklar

Her insanın görünümü yaş, ırk, cinsiyet gibi kavramlara bağlı olarak değişebilmektedir. Fakat yüz mimikleri evrenseldir ve insanlar birbirlerinin duygularını mimikler sayesinde kolayca anlayabilmektedir. Bununla birlikte, temel yüz mimiğini farklı kişiler arasında sınıflandırmak bilgisayarda görme sistemi için zorlu bir görevdir. Yüz mimiklerinin bir bilgisayar sistemi için tanınması ile ilgili zorluklar, aşağıdaki etkenlere bağlanabilir:

Poz: Yüzün görüntüleri, önden veya yandan yüz duruşlarına bağlı olarak değişir. Yüz, farklı bir açıya sahip olabilir; bu nedenle, göz veya burun gibi yüz özelliklerinin bir kısmı veya tamamı kapanabilir. Bu zorluğun üstesinden gelmek için çevirme, döndürme ve ölçekleme gibi iyi ön işleme tekniklerini uygulanabilir.



Şekil 1.1 Bir görüntünün farklı pozlardaki hali (Khatrı vd. 2014)

Gizleme (Occlusion): Bir görüntüde yüzün, başka yüzler, yüz maskesi, saç, gözlük gibi başka nesneler tarafından kısmen tıkanması mimiklerin özniteliklerinin çıkarması aşamasında zorluk yaratacaktır.



Şekil 1.2 Engellenmiş görüntüler

Aydınlatma: Görüntülerin farklı ışıklarla çekilmesi, bazı resimlerin oldukça fazla ışıktaki olup, bazı resimlerin yetersiz ışık alması gibi faktörler sebebiyle genellikle mimik öznelikleri hatalı olarak algılanmaktadır. Bu durumda yüz mimiklerinin tanıma oranı düşük çıkacaktır.



Şekil 1.3 Bir görüntünün farklı ışıklandırılması

Görüntüdeki bu aydınlatma değişimini telafi etmek için, DCT normalleştirme, histogram eşitleme, sıra normalleştirme gibi görüntü ön işleme yöntemleri öznelik çıkarımı öncesinde uygulanabilir (Khatri vd. 2014).

1.4 Genel Çerçeve

Tezin ikinci bölümünde yüz mimikleri tanıma sistem yaklaşımları kategorize edilmiş ve literatürde bu yaklaşımlarla ilgili yapılan çalışmalar özetlenmiştir.

Üçüncü kısımda proje kapsamında gerçekleştirilen ön adımlardan ve Viola ve Jones algoritması ile yüz tespiti bahsedilmiş, ele alınan iki yüz mimik tanıma sistemi yaklaşımı detaylı bir şekilde incelenmiştir.

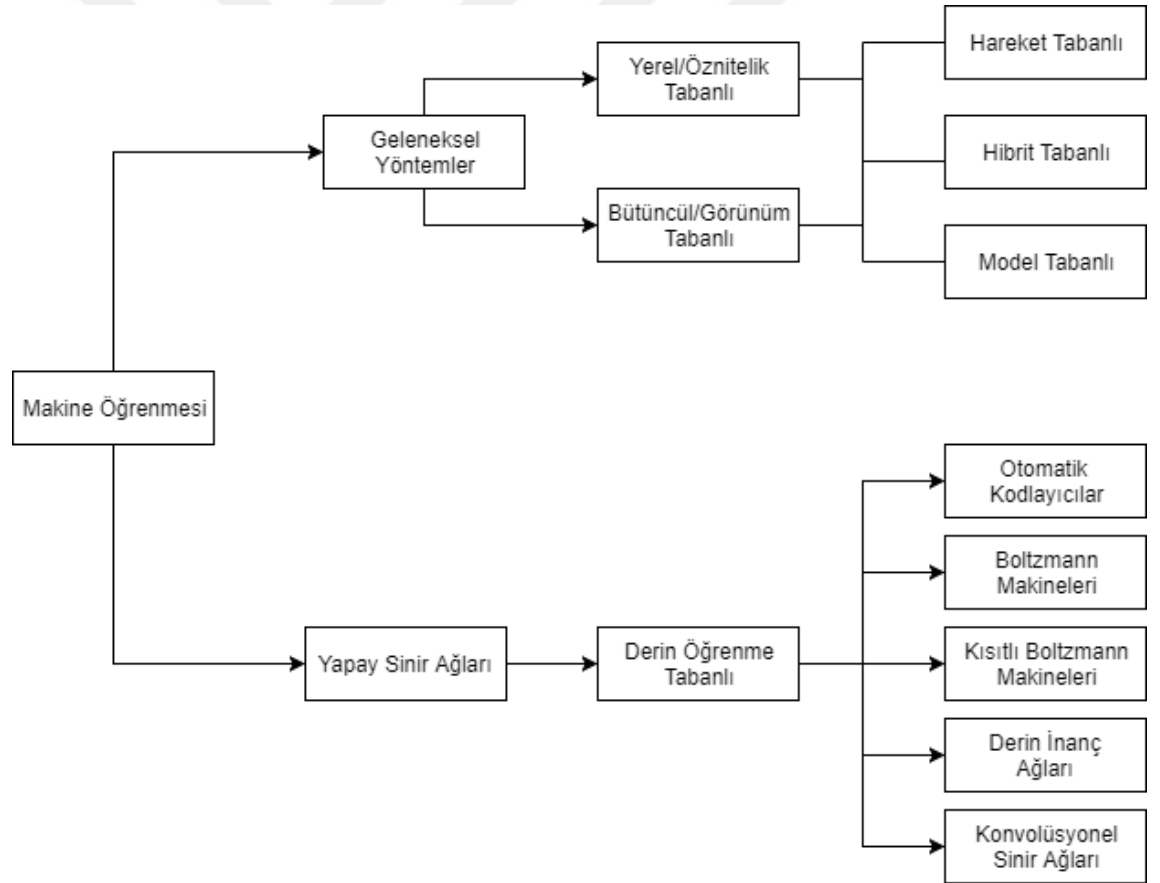
Dördüncü bölümde bu iki sistem karşılaştırılmış deneysel sonuçlar tartışılmıştır.

Sonuç bölümünde tez kapsamında geliştirilen yaklaşımlar özetlenmiş, gelecekte bu çalışmayla ilgili önerilerden bahsedilmiştir.



2. KURAMSAL TEMELLER ve KAYNAK ÖZETLERİ

Bu bölümde, insan yüz mimiklerinden durum analizi yapma yaklaşımları öznitelik çıkarma yöntemlerine ve sınıflandırma yöntemlerine göre kategorize edilmiş ve bu konuyla ilgili daha önceden yapılan çalışmalardan kısaca bahsedilmiştir. Fakat yapılan araştırmalara göre öznitelik çıkarma yöntemlerini ve sınıflandırma yöntemlerini kesin bir gruba dâhil etmek mümkün değildir. Örneğin; Principal Component Analysis (PCA) öznitelik çıkarma yöntemi görünüm tabanlı yaklaşımlarda kullanılır beyanı doğru değildir. Çünkü PCA görünüm tabanlı yaklaşımda kullanıldığı gibi öznitelik tabanlı yaklaşımda da kullanılmaktadır. Bu yüzden yüz mimikleri analiz yaklaşımları literatür çalışması metot bazlı değil yaklaşım bazlı incelenmiştir.



Şekil 2.1 Yüz mimikleri durum analizi yaklaşımları

Bu kısımda yüz mimik tanıma sistem yaklaşımları iki ana kategoriye ayrılmıştır: geleneksel yöntemler ve yapay sinir ağları. Geleneksel yöntemler ise iki temel yaklaşıma ayrılmaktadır: yerel veya öznel tabanlı yaklaşımlar, bütüncül veya görünüm tabanlı yaklaşımlar. Bu ana yaklaşımlar ise, hibrit tabanlı yaklaşımlar, model tabanlı yaklaşımlar, hareket tabanlı yaklaşımlar olarak yeniden kategorileştirilmiştir. Diğer yandan son zamanlarda daha aktif kullanılan yapay sinir ağlarının altında derin öğrenme tabanlı yaklaşımlar incelenmiştir. Derin öğrenme tabanlı yaklaşımlardan ise otomatik kodlayıcılar, Boltzmann makineleri, kısıtlı Boltzmann makineleri, derin inanç ağları ve konvolüsyonel sinir ağları incelenmiştir.

2.1 Makine Öğrenmesi

Yapay zekâ, “bir bilgisayarın ya da bilgisayar denetimli bir makinenin, genellikle insana özgü nitelikler olduğu varsayılan akıl yürütme, anlam çıkartma, genelleme ve geçmiş deneyimlerden öğrenme gibi yüksek zihinsel süreçlere ilişkin görevleri yerine getirme yeteneğidir” (Nabiyev 2010). Yapay zekâ alanında yapılan ilk çalışmada “Bilgisayarlar düşünebilir mi?” sorusuna açıklık getirmek isteyen Turing (1950) soruların ve yanıtlarının tuşlar ve ekran aracılığıyla yapıldığı bir soru yanıt oyununda, cevapları kontrol ederek, bu yanıtların sahibinin makine veya insan olduğunu anlamasını sağlayan bir test geliştirmiştir. Böylece bir makinenin düşünmesi ile ilgili açıklamaların mantıksal olarak mümkün olup olmadığını araştırmıştır.

Makine öğrenmesi, belirli bir görevi yerine getirmek için, geçmişteki verileri kullanarak gelecek için tahminlerde bulunan, görevi nasıl yerine getireceğine dair öğrenme yeteneği kazandırılmış, büyük miktarlarda veri ve algoritma kullanan eğitilmiş makine uygulamaları olup yapay zekânın bir alt dalıdır. Bu alanda yapılan ilk çalışmalardan biri Samuel (1959) makine öğrenme yöntemi ile geliştirdiği bir dama oyunudur. Çalışmada dama oyununun bir bilgisayar tarafından programlanabilip gerçek dünya uygulaması gibi oynamayı öğrenmesi için iki adet makine öğrenme tekniği geliştirilmiştir.

Makine öğrenmesi temelde gözetimli (supervised) ve gözetimsiz (unsupervised) öğrenme olmak üzere iki gruba ayrılır.

- **Gözetimli öğrenme**

Seçilen algoritmaya uygun olarak ilgili veriler hazırlandıktan sonra, ilk aşamada verinin bir kısmı modelin öğrenilmesi, diğer kısmı ise modelin geçerliliğinin test edilmesi için ayrılır. Modelin öğrenilmesi, öğrenim kümesi kullanılarak gerçekleştirildikten sonra, test kümesi ile modelin doğruluk derecesi belirlenir (Lektemur 2017).

Bu çalışmada geliştirilen her iki sistemde de gözetimli öğrenme söz konusudur. Tüm resimlerin bir kısmı eğitim için bir kısmı test için ayrılmıştır. Eğitim resimleri ise, “üzgün”, “mutlu” vs. olarak gruplara ayrılıp etiketlenmiştir. Daha sonra test resimleri ile eğitilmiş resimler test edilmiş ve doğruluk derecesi belirlenmiştir.

- **Gözetimsiz öğrenme**

Verilerin izlenmesi ve bu örneklerin özellikleri arasındaki benzerliklerden hareket ederek sınıfların tanımlanması amaçlanmaktadır (Lektemur 2017). Burada mevcut verilerin özelliklerine göre kümeleme ve boyutsal azaltım yöntemi ile belli gruplar oluşturan öğrenme şeklidir.

Bu öğrenme şeklinde bu gruplama resmin “üzgün”, “mutlu” vs. olduğunu bilemeyecektir, ancak örnek bir resim verildiğinde bunun hangi kümeleme içerisinde olduğunu bularak benzerlerini getirebilir.

Günümüzde farkında olmadan birçok alanda makine öğrenme teknolojisi kullanılmaktadır. Makine öğrenme sistemleri, web aramalarından sosyal ağlardaki içerik filtrelemesine, e-ticaret web sitelerindeki önerilere, kameralar ve akıllı telefonlar gibi tüketici ürünlerinde giderek artan bir büyümeye ve sağlık alanında hastalıkların daha iyi anlaşılabilmesini sağlayan birçok gelişim ile modern toplumun birçok yönünü güçlendirmektedir. Ayrıca görüntüdeki nesneleri tanımlamak, konuşmayı metne dönüştürmek, haber öğelerini, kullanıcıların ilgi alanına giren yayınları veya ürünleri

eşleştirmek ve ilgili arama sonuçlarını seçmek için kullanılır. Yine sağlık alanında gen ve DNA dizilişi gibi bilgilerin işlenmesinde de bu sistem kullanılmaktadır.

2.1 Geleneksel Yöntemler

Temel bileşenler analizi, gabor dalgacıkları, lineer ayrık analizi, yerel ikili örüntü gibi algoritmalar çok eskilerden ortaya atılan, çıkarıldığı anda ses getiren başarılı algoritmalarlardır. Fakat her biri diğerinin eksikliğini tamamlamak için çıkarılan bu algoritmalar dizisine yeni algoritmalar eklenmektedir. Bu kısımda yüz mimikleri tanıma sistemleri öznitelik çıkarma aşamasında klasik algoritmalarla yapılan çalışmalardan bahsedilmiştir.

- **Öznitelik çıkarma**

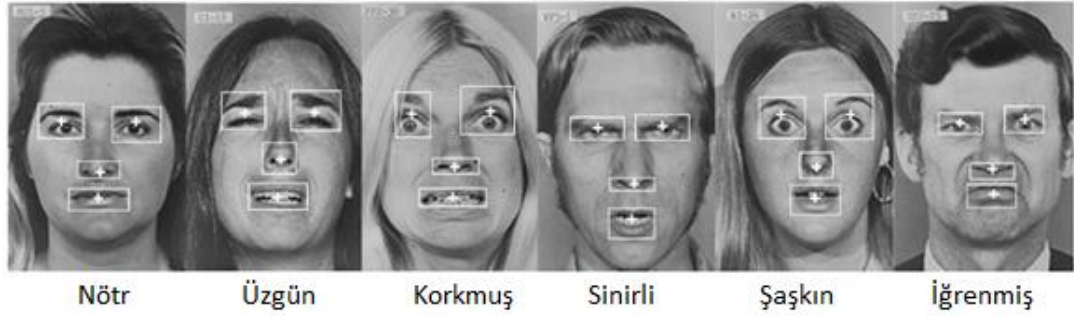
Yüze ait öznitelikler, kaşlar, gözler, burun ve ağzın belirgin özellikleridir. Mimikler, bu özniteliklerin biçimsizleşmesi ile oluşur ve yüzde geçici olarak görünür. Başarılı bir yüz mimik tanıma sistemi için en kritik konu, sağlam ve etkin bir yüz öznitelik tanımlayıcısı tasarlamak ve öznitelik çıkarmaktır. Bu aşamada kullanılan teknik, yüz mimikleri tanıma sisteminin başarısını etkilemektedir. Sağlam ve etkili öznitelikleri çıkarmak için birçok yaklaşım, algoritma geliştirilmiştir. Kumari vd. (2015), yüz özniteliklerini çıkarmak için kullanılan en popüler tekniklerden bahsetmişlerdir.

2.1.1 Yerel/Öznitelik tabanlı yaklaşımlar

Öznitelik tabanlı yöntemde ele alınacak görüntü; nokta, eğri vb. gibi geometrik parametreler yardımı ile temsil edilir. Bu temsil, gözler, burun, kulaklar, ağız vb. gibi yüzün farklı bileşenlerinin bulunması ve bunların ilgili konumlarının yer değişimlerinin ölçülmesi şeklindedir. Yüz geometrisini temsil eden yüze ait bu bileşenler, mimik öznitelik vektörleri oluşturmak için çıkarılır. Yani genellikle yüz bölgesi analizlerinde öznitelik ölçümünün görevi; önemli nokta bulma ve izleme ile bağlantılıdır. Yüz

görüntülerinde yüze ait öznitelik noktalarının yerinin saptanması, çok sayıda yüz görüntüsünün yorumlanma görevi için önemli bir aşamadır.

Gizatdinova ve Surakka (2006) çalışmalarında, kaş, göz, burun altı ve ağız bölgesi olmak üzere 4 adet önemli yüz noktalarını tespit edip işaretlenmiştir. Daha önce Golovan (2000) önerdiği metot üzerinden iki iyileştirme yapılmıştır. İlk iyileştirme, görüntünün kenar haritalarını oluşturmak için kullanılan yönlü kenar sayısının azaltılmasıdır. Bu azaltma, yöntemin hesaplama yükünü azaltmıştır. İkinci iyileştirme, yüzün işaretlenen önemli yerlerinin yönelim modelinin oluşturulmasıdır. Bu model, görüntüde işaretlenmiş noktanın varlığını doğrulamak için kullanılmıştır.



Şekil 2.2 Facial Affect veritabanındaki resimler üzerinde önemli yüz noktalarının ve bu nokta kümelerinin merkezlerinin işaretlenmesi (Gizatdinova ve Surakka 2006)

Elde edilen sonuçlar, görüntünün kenar haritalarını oluşturmak için yönlü kenarların temel öznitelik olarak seçilmesinin, yüz mimikleri ve görüntü boyutundaki değişimlere bakılmaksızın göz tespiti için belirli bir aralıkta değişmezlik sağladığını ortaya koymuştur. Bununla birlikte, alt yüzün önemli noktalarını saptamak mimik ve görüntü boyutundaki değişikliklerden etkilenmiştir. Mimik değişikliği ve görüntü boyutundaki azalma ağız ve burun bölgelerinin ortalama algılama oranlarını sırasıyla % 86 ve % 78'e kadar düşürmüştür.

Kotsia vd. (2007) çalışmalarında, yüz mimikleri analizi için, yüz görüntüsünden bir dizi yüz noktası tespit ettikten sonra bir Kanade-Lucas-Tomasi (KLT) takip cihazı kullanmışlardır. Öznitelik olarak; birinci ve en büyük yoğunluk çerçeveleri arasındaki düğüm koordinatlarının farklılıkları olarak tanımlanan belirli düğümlerin geometrik yer

değiřtirmesi çıkarılmıřtır. Çıkarılan bu öznitelikler, hem altı temel yüz mimiklerini tanımak için kullanılan DVM sınıflandırıcısına hem de seçilen yüz eylem birimlerini sınıflandıran Facial Action Unit (FAU) algılayıcısına girdi olmuřtur. Cohn Kanade (CK) veri tabanında test edilen sonuçlara göre, DVM'leri kullanarak yüz mimiklerinin tanıma başarı oranı FAU algılamaya dayalı yüz mimikleri tanıma başarı oranından fazla çıkmıřtır.

Öznitelik tabanlı yüz mimikleri tanıma yaklaşımlarında, herhangi yüz doku bilgisi alınmadan, yalnızca yüz görüntü dizisinden çıkarılan geometrik bilgiler kullanılarak gerçekleştirilir.

Ghimire ve Lee (2013) çalışmalarında, dinamik yüz mimiklerini tanımak için kademeli geometrik öznitelikler çıkarmıřlardır. Önerilen yüz mimik tanıma sistemi, ilk çerçevede yüz yer işaretlerini başlatmak için elastik demet grafięi kullanan tam otomatik bir sistemdir. Daha sonra bu öznitelikleri çoklu sınıf AdaBoost ve DVM ile sınıflandırmıřlar ve bu iki yöntemi karşılařtırmıřlardır. Nihayetinde CK+ yüz mimik veritabanındaki sonuçlara göre, DVM AdaBoost metodundan daha iyi tanıma başarıısı elde etmiřtir.

2.1.2 Bütüncül/Görünüm tabanlı yaklaşımlar

Yüz görüntüsü bütün bir birim olarak gösterilebiliyorsa buna bütünsel gösterim denmektedir. Görünüm öznitelikleri; kırışık, çıkıntı ve çizgi gibi yüzün ten dokusunu temsil ederler. Görünüm tabanlı yaklaşımlar bir öznitelik vektörü çıkarmak için tüm yüze uygulanır ve dolayısıyla ayrıntılı bilgi verir.

Görünüm tabanlı yöntemlerde, öznitelik bileřenlerin řekli veya ilgili uzaklıęı yerine piksel deęerlerine önem verilir. Bir başka deyiřle; bu yöntemde temsil için uzaysal noktalar kullanmak yerine, piksellerin yoğunluk, histogram gibi hareket parametreleri kullanılır. Bu parametreler, yüzün ilgili bölgelerinin zaman ve renklerine göre deęiřir.

Bartlett vd. (2004) çalışmalarında, yüz mimikleri tanıma problemine uygulanan, DVM ve Lineer Diskriminant Analizi (LDA) gibi makine öğrenme yöntemlerinin sistematik bir karşılaştırmasını sunmuşlardır. Bu çalışmalarında sistemi tamamen otomatik yüz hareket kodlamasına uygulamışlardır. Bu, kızılötesi göz pozisyon sinyali olmadan görüntülerin tam otomatik Facial Action Code System (FACS) kodlaması için bilinen ilk sistemdir. Gabor filtrelerinin bir alt kümesini seçerek ve daha sonra seçilen filtrelerin çıktıları üzerinde DVM eğitilerek elde edilen sonuç daha iyi çıkmıştır.

Whitehill ve Omlin (2006) çalışmalarında, FACS eylem birimlerini (AU) sınıflandırmak için Haar özniteliklerini ve Adaboost algoritmasını incelemiştir. Gabor özniteliklerinin DVM ile sınıflandırmanın en son teknoloji yöntemine kıyasla, bu yeni yaklaşımın tanıma doğruluğu ve işlem süresi değerlendirilmiştir. CK mimik veritabanı üzerindeki deneysel sonuçlara göre, Haar+Adaboost metodu standart Gabor+DVM metodu ile benzer AU tanıma doğruluğuna ulaşmıştır, ancak birkaç düzende daha hızlı performans göstermiştir.

Lien vd. (2008) çalışmalarında, düşük çözünürlüklü görüntülerde altı çeşit yüz mimiklerini (öfke, korku, mutluluk, nötr, üzüntü ve sürpriz) tanımlamak için "mimik geçişi" adı verilen yeni bir görünüm tabanlı mimik tanıma yöntemi önermişlerdir. Çalışmalarında, arttırılmış ağaç sınıflandırıcıları ve şablon eşleştirmesi, yüz mimiklerini karakterize edebilen etkili yüz bölgesini bulmak ve kırmak için kullanılmıştır. Ardından, görüntüleri bir dizi mimik geçiş matrisi aracılığıyla dönüştüren yüz mimikleri, yüz mimiklerini tanımlamak için gerçek yüz görüntüleriyle eşleştirilmiştir. Buna ek olarak, gözlerin ve ağzın konumlandırılmasını iyileştirmek için aydınlatma karşılama projeksiyon yöntemi, şablon eşleştirme ve Hough dönüşümü kullanılmıştır. Mimik dönüşüm işlemi için gözlerin ve ağzın konumlarına göre etkili yüz bölgesi çıkarılmıştır. Önerilen sistem, yüz mimiklerini her çerçeve için 0.24 saniye hızıyla ve % 86'nın üzerinde doğrulukla tanıyabilmiştir.

Ahmed vd. (2014), kişiden bağımsız yüz mimikleri tanıma için, LBP'in sınırlarını aşan, Compound Local Binary Pattern (CLBP) ile yapılandırılmış sağlam bir yüz öznitelik tanımlayıcısı sunmuşlardır. Önerilen yöntem, iki gri değer arasındaki farkın büyüklüğü

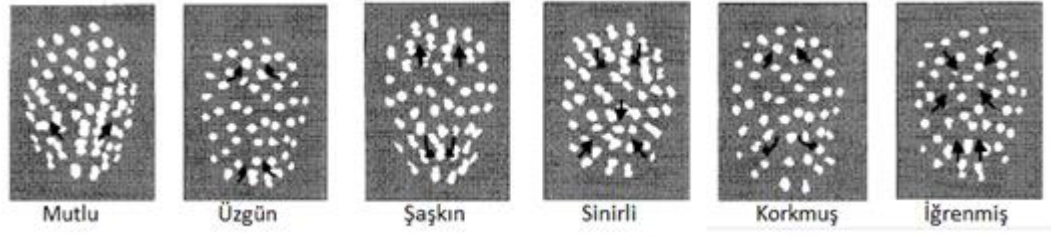
bilgisini orijinal LBP deseniyle birleştiren bir kodlama şemasını kullanır ve bu nedenle, LBP'nin tutarlı olamadığı bazı durumları iyileştirir. Önerilen yöntemin tanıma performansı, DVM sınıflandırıcısı ile CK ve JAFFE veritabanı kullanılarak değerlendirilmiştir. Prototipik mimiklerle yapılan deneysel sonuçlar, CLBP öznitelik tanımlayıcısının, bazı iyi bilinen görünüm tabanlı yöntemlerden daha iyi performans gösteren, yüksek ayrımcılık becerisine sahip, etkin ve verimli bir yaklaşım sağladığını göstermektedir.

2.1.3 Hareket tabanlı yaklaşımlar

Öznitelik tabanlı yaklaşıma göre en önemli adım; bir yüze ait yoğun noktalar kümesini izlemek ve yerelleştirmektir. Yoğunluk ölçümü, piksel değişimi miktarını gösteren önemli bir faktördür. Çoğu geometrik öznitelik tabanlı yaklaşımlar yoğun yüz noktası kümesini izlemek için optik akış veya şekil ve gri tonlama bilgilerinin istatistiksel modeli olan Aktif Görünüm Modeli (AGM) gibi yöntemler kullanırlar. Bu yüz yer işaretlerinin yerleri daha sonra yüze ait özniteliklerin şeklini ve mimik değiştikçe yüze ait özniteliklerin hareketini çıkarmak için hareket tabanlı yaklaşım denilen farklı yollarla kullanılır.

Farklı araştırmacıların hareket tabanlı yaklaşımlara katılımı şöyledir:

Yacoob ve Davis (1996), görüntü dizilerinden 6 adet evrensel yüz mimiklerinin tanınması için yüz dinamiklerinin analizi ve gösterimi için bir yaklaşım sunmuşlardır. Algoritma, insan yüz mimiklerinden kaynaklanan hareketlerin yönünü belirlemek için optik akış hesaplamasını kullanmaktadır. Bu yaklaşım, yüzün ana bölgelerinin nitel olarak izlenmesine ve yüksek yoğunluk gradyan noktalarında akış hesaplamasına dayanmaktadır. Metot bir laboratuvar ortamında 30 kişi ile test edilmiştir ve o zamanki koşullara göre iyi bir sınıflandırma başarısı elde edilmiştir.



Şekil 2.3 Yüz mimikleri optik akış hesaplaması

Dai vd. (2000) çalışmalarında, yüz hareketlerini çıkarmak ve konuşma yeteneği olmayan kişilerin mimiklerini tanımak için bir metot geliştirmişlerdir. Önerilen yöntemde, yüz hareketlerine karşılık gelen görüntü dizisinin optik akışı hesaplanmıştır. Ağız ve göz bölgesindeki S ve Y koordinat eksenleri üzerindeki optik akış projeksiyon histogramına dayanarak, yüz mimiklerinin hareket öznitelikleri yüz mimiklerinin ortak bellek modelini oluşturmak için çıkarılmıştır. Bu modelden yararlanarak yüz mimikleri tanınmıştır.

2.1.4 Model tabanlı yaklaşımlar

Şu ana kadar 2B statik görüntü ile yapılan çalışmalardan bahsedilmiştir ve YMA için iyi tanıma oranları elde edilmiştir. Fakat 3B kafa pozisyonu değiştiğinde mimikleri tanımda yüze ait özniteliklerin şekillendirilmesi en zorlu olaylardır. Çünkü yüz hareketleri statik yüz özniteliklerine konulduğunda yüz mimiklerinin tanınmasında sorun ortaya çıkar. Çoğu araştırmacı model tabanlı yaklaşımlara odaklanmıştır çünkü bu yaklaşımlar yüze ait öznitelik geometrisi hakkında özlü bilgiler verir (Kotsia ve Pitas 2007).

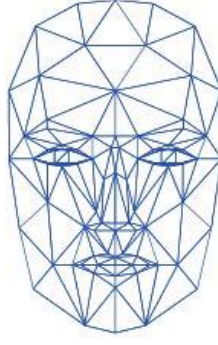
Huang vd. (1999) çalışmalarında, yüz mimikleri özniteliklerini bulmak için nokta dağılım modeli ve gri seviye modelini uygulamışlardır. Ardından yüz öznitelikleri üzerindeki işaretlenmiş belirli noktaların konum değişimlerini 10 eylem parametresi ile tanımlanmıştır. Bu parametrelerin boyutlarını iki boyuta indirgemek ve tanıma sürecini basitleştirmek için eylem parametreleri üzerinde analiz yapmışlardır. Yaptıkları sistem, yüz özniteliklerinin kontur değişimini doğru bir şekilde izleyebilmiştir, ancak tüm eylem birimleri çıkarılmış bilgidan tanımlayamamıştır. Bu nedenle, yüz mimiklerini

etkili bir şekilde tanımlamak için bazı eylem birimleri diğer eylem parametreleri ile yer değiştirilmeye çalışılmıştır. Sonuçta yaptıkları sistemin yüz mimiklerini etkin bir şekilde tanıyabileceğini göstermişlerdir.

Göktürk vd. (2002) çalışmalarında, YMA için görüş açısı ve poz varyasyonundan bağımsız bir şekilde sağlam yeni bir metot önermişlerdir. Bu sistem monoküler bir video dizisinin her karesinden yüz pozu ve şeklini eşzamanlı ve sağlam bir şekilde çıkarmak için yeni bir 3B model tabanlı takip cihazı kullanmıştır. Bu çalışmanın iki ana katkısı vardır. Birincisi, 3B takip yoluyla çıkarılan 3B bilgilerinin, geniş dönüşlü ve çevirisel kafa hareketlerine (kafa dönüşünde 90 dereceye kadar) rağmen sağlam yüz mimik tanımanın mümkün kıldığı gösterilmiştir. İkincisi, DVM'sinin etkin bir sınıflandırma için uygun bir araç olduğu gösterilmiştir. 5 farklı dinamik yüz mimiği için % 91 tanıma oranı elde edilmiştir.

Model tabanlı metotlar morfolojik işlemler için çok yararlıdır. Bunun nedeni, otomatik yüz tanıma gürültülü verileri ortadan kaldırmaktadır (Amberg vd. 2008).

Vretos vd. (2009) çalışmalarında, Candide modelinin köşeleri üzerinde öz (eigen) vektörleri bulmak için PCA kullanarak YMA için bir yöntem sunmuşlardır. Daha sonra, yüz mimiklerini sınıflandırmak için DVM'leri kullanmışlardır. Sonuçlar önerilen yöntemin, model tabanlı yüz mimikleri tanımaya yönelik iyi bir yapı olduğunu açıkça göstermektedir. Elde edilen yüz mimikleri sınıflandırma doğruluğu % 90'dır. İşletilen öznelik alanının en önemli avantajı, düzlem içi döndürme, çevirme ve ölçeklendirmede dayanıklı, sabit olmasıdır. Fakat yöntemin çekirdek ve çekirdek parametrelerine göre DVM konfigürasyonuna duyarlı olmadığını da belirtmek gerekir; bu, yöntemin iyi genelleme niteliklerine sahip olduğunun bir göstergesidir.



Şekil 2.4 Candide modeli

2.1.5 Hibrit tabanlı yaklaşımlar

Bazen tüm yüzün bilgilerini kullanmak gereksizdir çünkü bir tek mimiğin ortaya çıkması yüze ait tüm öznitelikleri değiştirmez; örneğin, gülümseme mimiği tüm öznitelikleri etkilemez, sadece yanak ve dudak vb. görünümü etkiler. Böylece, yerel tabanlı yaklaşımlar bütünsel temelli yaklaşımların aksine yalnızca etkilenen yüz özniteliklerini işlemek için bir yol sağlar. Bazen de yüzün tam olarak ele alınması daha avantajlı olup görünüm tabanlı yöntem öznitelik tabanlı yöntemle göre daha başarılı olmaktadır. Bu noktada hibrit yaklaşım adı verilen yöntem işin içine girmektedir. Hibrit tabanlı yaklaşımlar hem öznitelik tabanlı hem de görünüm tabanlı yaklaşımları birleştirerek daha iyi tanıma oranı elde edebilmektedir.

Zhang vd. (1998), yaptıkları yüz görüntülerinden mimik tanıma çalışmasında, iki tip öznitelik çıkarma yaklaşımını karşılaştırmışlardır. İlki geometrik tabanlı bir yaklaşım olup, yüz üzerindeki bir dizi referans noktaların geometrik konumları çıkarılmıştır. İkincisi görünüm tabanlı yaklaşım; referans noktalardaki çok ölçekli ve çok yönlü gabor dalgacık katsayıları kümesi yüz görüntüsünden çıkarılmıştır. Bunun için iki katmanlı perseptrona dayalı bir mimari geliştirilmiştir. Bu iki yaklaşımın mimik tanıma performansının karşılaştırılması sonucu gabor dalgacık katsayılarının geometrik konumlardan daha güçlü olduğu ortaya çıkmıştır. Ayrıca, perseptronun ilk katmanı öznitelik alanının boyutsallığını doğrusal olmayan bir şekilde küçülttüğünden, iyi bir mimik tanıma oranı elde etmek için, makul bir gizli birim yani uygun boyut üzerinde çalışmışlardır.

Shinohara ve Otsu (2004), fisher ağırlık haritaları (fisher weight map) ve Higher Order Local Autocorrelation (HLAC) öznitelikleri kullanarak bir yüz mimikleri tanıma sistemi geliştirmişlerdir. Geliştirilen sistem, yerel/öznitelik tabanlı ve görüntü tabanlı yaklaşımın birleştirildiği hibrit bir yaklaşımdır. Bu çalışmada HLAC öznitelikleri, bir görüntünün her pikselinde hesaplanır. Bu özellikler, bir öznitelik vektörü elde etmek için bir ağırlık haritası ile bütünleştirilmiştir. Fisher ağırlık haritası olarak adlandırılan bu optimal ağırlık haritası, öznitelik vektörlerinin fisher kriterini maksimize ederek bulunur. Fisher diskriminant analizi, öznitelik vektöründen bir görüntüyü tanımak için kullanılır. Yüz mimiğini tanıma konusundaki deneylere göre, geliştirilen hibrit metodu HLAC öznitelik tabanlı metot ve fisher yüz metodundan daha iyi performans göstermiştir. Önerilen yöntem, diğer yerel öznitelik türleri (örn, gabor dalgacık, doku, optik akış) ve daha karmaşık ama güçlü sınıflandırma yöntemleri (örn, sinir ağları, DVM, çekirdek temelli fisher ayırma analizi) ile kullanılabilir.

Happy (2015) çalışmasında, şekil ve görünüm özniteliklerini birleştirerek hibrit bir öznitelik vektörü oluşturan bir yaklaşımı ele almaktadır. Görünüm özellikleri olarak Yerel İkili Örüntüler (LBP) ve şekil tanımlayıcıları olarak Pyramid of Histogram of Gradients (PHOG) çıkarılmıştır. Yüze ait önemli noktaları tespit ettikten sonra, aktif parçalar (patch) yerleştirilmiş ve bu parçalardan hibrit öznitelikler hesaplanmıştır. Öznitelik çıkarmak için tüm yüz yerine yüzün küçük parçalarının kullanılması, hesaplama maliyetini düşürmüştür ve sınıflandırmada özniteliklerin aşırı uyumunu önlemiştir. LDA kullanılarak, özniteliklerin boyutları azaltılmış ve daha sonra DVM kullanılarak sınıflandırılmıştır. CK+ ve JAFFE veritabanında yapılan sonuçlara göre PHOG sırasıyla % 89.91, % 76.43 başarı elde ederken, LBP sırasıyla % 60.54, % 55.71 elde etmiştir.

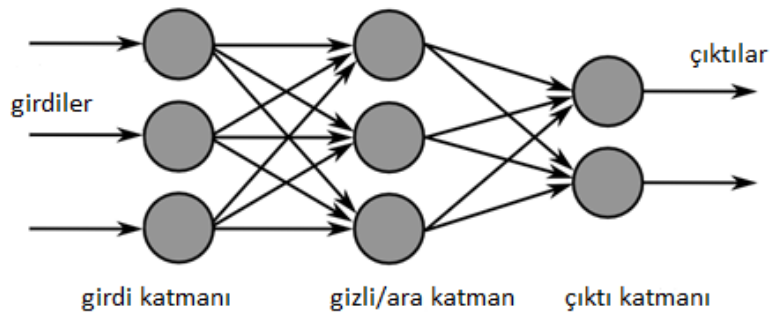
Ghimire vd. (2017), alınan görüntüden mimik tanımak için DVM sınıflandırıcısı ile görünüm ve geometrik özniteliklerin kombinasyonunu kullanan yeni bir yöntem önermişlerdir. Önerilen yüz mimikleri tanıma sisteminin performansı, CK + yüz mimik veri setleri üzerinde test edilmiştir. Genelde, yüz mimiklerinin tanınması için görünüm öznitelikleri, yüz bölgesinin düzenli gridlere bölünmesi (bütüncül gösterim) ile hesaplanır. Ancak, bu çalışmada tüm yüz bölgesi belirli yerel alanlara bölünerek belirli

görünüm öznitelikleri çıkarılmıştır. Aynı zamanda geometrik öznitelikler de ilgili bölgelerden çıkarılmıştır. Önemli yerel bölgeler, öznitelik boyutunun azaltılması ve tanıma doğruluğundaki iyileşme ile sonuçlanan arttırılmış arama yaklaşımı kullanılarak belirlenmiştir. Belirli bölgelerin öznitelikleri kullanılarak oluşturulan yerel bölge tabanlı gösterim tekniği, gride dayalı bütüncül gösterim ile karşılaştırılmıştır. YMA deneysel sonuçları hibrit yaklaşımın daha iyi olduğunu göstermiştir.

Sajjad vd. (2017), yedi adet evrensel yüz mimiklerini sınıflandırarak analiz etmek için, yerel ve bütüncül öznitelikleri birleştirilerek hibrit bir yaklaşım sunmuşlardır. Öznitelik çıkarımı HOG ile Uniform Local Ternary Pattern (U-LTP) birlikte gerçekleştirilmiştir. En uygun HOG parametre seti ULTP ile birlikte analizinden elde edilmiştir. Ayrıca bu ULTP gürültü ve oklüzyonları içeren yüz görüntüleri üzerinde performansı artırmıştır. HOG ve ULTP yoluyla çıkarılan öznitelikler tek bir öznitelik vektörüne birleştirilmiş ve mimik sınıflandırması için bu öznitelik vektörü çok sınıflı bir DVM ile beslenmiştir. JAFFE, MMI, ve CK+ gibi üç veritabanı üzerinde değerlendirilen deneysel sonuçlara göre alınan başarı sırasıyla 95.71, 98.20 ve 99.68 şeklinde olmuştur.

2.2 Yapay Sinir Ağları

Yapay zekâ, girdi ve çıktıları olan, birbirleri ile sıkı bir şekilde ilişkilenmiş işlem elemanları olup, insan beynindeki hücrelerin çalışma prensibini matematiksel olarak modelleyen bir bilgisayar sistemidir.



Şekil 2.5 Tipik bir yapay sinir ağı yapısı

YSA'larında girdi, çıktı ve gizli katman denilen ara katmanlar olmak üzere 3 tip katman vardır. Bilgiler ağa girdi katmanından alınır. Bu katmanlar arasındaki iletişime bağlantı denir. Ağdaki her bağlantının bir ağırlık değeri vardır. Ağın ağırlık değerleri kullanılarak girdi olarak alınan bilgiler ara katmanlarda işlenerek çıktıya dönüştürülür. Ağın doğru çıktılar üretebilmesi ağırlık değerlerinin doğruluğuna bağlıdır. Doğru ağırlıkların bulunması işlemine ağı eğitilmesi denmektedir. Bu değerler başlangıçta rastgele atanırlar. Eğitim aşamasında her örnek gösteriminde ağırlıklar güncellenerek en doğru değerler bulunmaya çalışılır. Bu işlemler ağ eğitim setindeki örneklerin tamamı için tekrarlanır. Böylece eğitim bittikten sonra test seti denenir (Öztemel 2003).

İnsan beyninin hesaplama yeteneğinden esinlenen McCulloch ve Pitts (1943) elektrik devreleriyle basit bir sinir ağı modelleyerek yapay sinir ağlarına ilk adımı atmışlardır. Çalışmalarında, sinir aktivitesinin "hepsi ya da hiç" karakteri nedeniyle, sinirsel olaylar ve aralarındaki ilişkiler önerme mantığı ile işlenmiştir. Olası nörofizyolojik varsayımlar arasındaki birçok özel seçeneğin eşdeğer olduğunu göstermişler; bu da, bir varsayım altında her ağı davranışında, belki de aynı zamanda olmasa da ötekinin altında hareket eden ve aynı sonuçları veren başka bir ağı var olduğu anlamına gelmektedir. Fakat yaptıkları bu basit sistemde öğrenme yoktur.

Farley ve Clark (1954) çalışmalarında, dijital bilgisayar tarafından simülasyonlara ilişkin açıklamalarla birlikte, kendi kendini organize eden sistemlerle ve bunların senteziyle ilgili fikirlerin genel bir tartışması verilmektedir. Daha sonra, gerçek bir sistemin sentezi ve simülasyonu anlatılmaktadır. Başlangıçta rastgele geniş sınırlar içerisinde düzenlenen bu sistem, basit bir öngörülen görevi yerine getirmek üzere kendi kendini organize eder.

Yapay sinir ağları öğrenme biçimine göre gözetimli öğrenme ve gözetimsiz öğrenme (Bölüm 2.1) olarak incelenir. Gözetimli öğrenmede ağa girdi çıktı vektörleri şeklinde ayrıntılı bir biçimde eğitim örnekleri verilmektedir. Gözetimsiz öğrenmede ise ağa girdi bilgileri verilir, ağ önemli özneliliklerini keşfeder ve uygun kategoriler halinde girdi verilerinin nasıl sınıflandırıldığını öğrenir.

Gözetimli öğrenme algoritmaları;

- ✓ Tek katmanlı perseptron
- ✓ Çok katmanlı perseptron
- ✓ Konvolüsyonel sinir ağları

Gözetimsiz ve yarı gözetimli öğrenme algoritmaları;

- ✓ Otomatik kodlayıcılar
- ✓ Boltzmann makineleri
- ✓ Kısıtlı Boltzmann makineleri
- ✓ Derin inanç ağları

Yapay sinir ağları yapılarına göre ileri beslemeli sinir ağları ve geri beslemeli sinir ağları diye ikiye ayrılır.

İleri beslemeli sinir ağları;

- ✓ Tek katmanlı perseptron
- ✓ Çok katmanlı perseptron
- ✓ Otomatik kodlayıcı
- ✓ Derin inanç ağları

Geri beslemeli sinir ağları;

- ✓ Boltzmann makineleri
- ✓ Konvolüsyonel sinir ağları

➤ **İleri beslemeli sinir ağları**

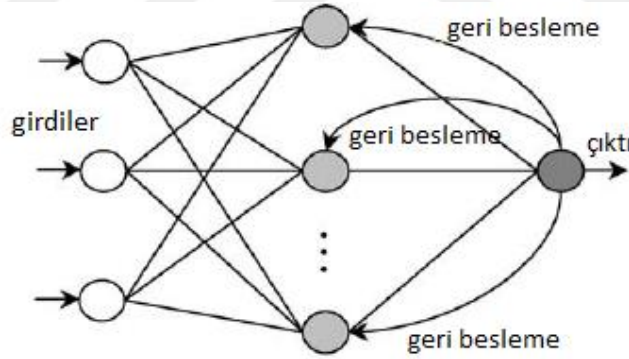
İleri beslemeli ağlarda nöronlar girişten çıkışa doğru düzenli katmanlar şeklindedir. Bir katmandan sadece kendinden sonraki katmanlara bağ bulunmaktadır. Geriye doğru herhangi bir bağlantı söz konusu değildir. Yapay sinir ağına gelen veri sırasıyla girdi katmanı, gizli katmanlar ve çıktı katmanından geçerek sonuç verir.

➤ Geri beslemeli sinir ağı

Sinir ağı eğitilirken hata varsa hataları azaltmak ister. Bu durumda eğitim şu şekilde gerçekleştirilir:

- Bir veri ağı girdi katmanına sunulur. İstenen çıktı, çıktı katmanı tarafından oluşturulana kadar girdi katmandan katmana yayılır.
- Eğer bu çıktı istenen çıktıdan farklıysa, bir hata hesaplanır ve ağ boyunca çıktı katmanından girdi katmanına doğru geriye doğru yayılır. Hata yayıldığı için ağırlıklar değiştirilir.

Bu yapısı ile geri beslemeli yapay sinir ağı doğrusal olmayan dinamik bir davranış göstermektedir.



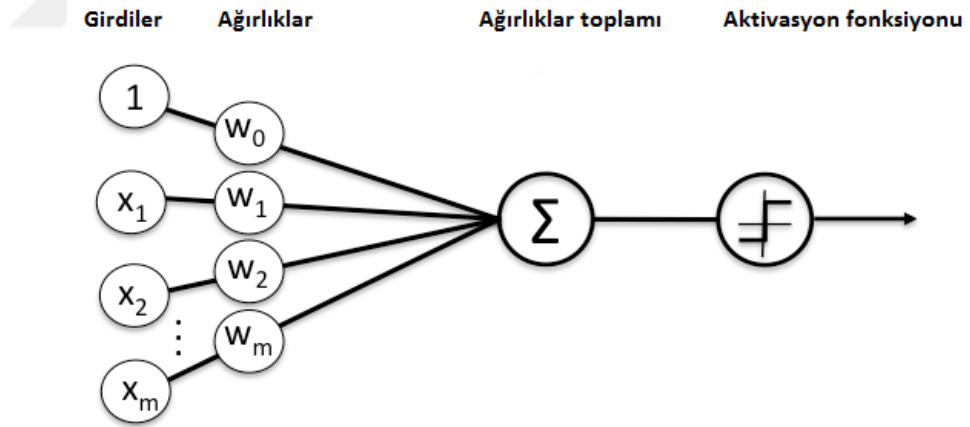
Şekil 2.6 Geri beslemeli sinir ağı

Bu kısımda derin öğrenmenin de anlaşılmasına yardımcı olması beklentisiyle perseptron ve çok katmanlı perseptronlardan kısaca bahsedilmiştir.

- **Perseptron**

Rosenblatt (1958), fiziksel dünyayla ilgili bilgilerin nasıl algılandığına, bilginin hangi biçimde hatırlandığına ve bellekteki bilgileri hatırlama ve davranışa nasıl etki ettiğine ilişkin sorulara cevap bulmak için hipotetik bir sinir sistemi için bir teori geliştirmiştir. Rosenblatt, bu teoriye perseptron adını vermiştir. Teori, biyofizik ve psikoloji arasında bir köprü görevi görmektedir. Öğrenme eğrilerini nörolojik değişkenlerden veya bunun tersinden tahmin etmektedir.

Perseptron, bir sinir hücresinin birden fazla girdiyi alarak bir çıktı üretmesi prensibine dayanır. Perceptrona örneklerin öğretilmesi düzlem üzerinde lineer doğrunun konumunun değiştirilmesiyle olur. “Problemin girdileri ağırlıklar ile çarpılıp toplandıktan sonra elde edilen değerin bir eşik değerinden büyük veya küçük olmasına göre girdinin sınıfı belirlenir. Öğrenme sırasında hem ağırlıklar hem de eşik değeri biriminin ağırlık değeri değiştirilir.” (Öztemel 2012).



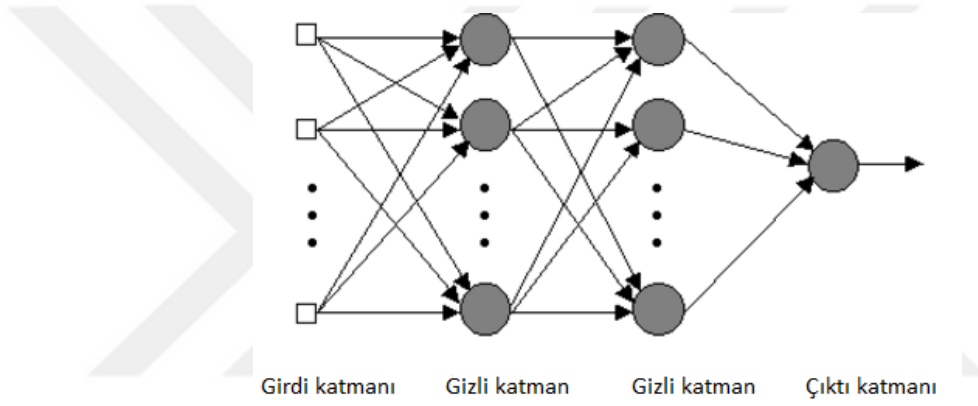
Şekil 2.7 Perseptron yapısı

Perceptron doğrusal bir fonksiyonla iki parçaya bölünebilen problemlerde kullanılabilir. Bu problemlere AND, OR, NOT problemleri örnek verilebilir. Doğrusal olarak ayrılmayan problemleri çözmede başarısızdır. Örneğin XOR problemini çözmede başarısız kalmıştır.

- **Çok katmanlı perseptron**

1950 yılında başlayan yapay sinir ağları ile ilgili çalışmalar bir süre devam etmiştir. Fakat Minsky (1969), perseptron modelinin XOR problemi için çözüm olamayacağını ispat etmiş, Rumelhart vd. (1986) çok katmanlı perseptronlar ile problemin çözülebileceğini ispat etmişlerdir.

İçinde barındırdığı perceptronlardan dolayı doğrusal olmayan sınıflandırmalara katkısı yoktur.



Şekil 2.8 Çok katmanlı perseptron yapısı

Çok katmanlı perseptron sinir ağları, giriş nöronlarının bulunduğu girdi katmanı, çıkış nöronlarının bulunduğu çıktı katmanı ve bir veya daha fazla gizli katmandan oluşan bir modeldir. Amaç girdi katmanından alınan verilerin işlenmesidir. Çıktı katmanında gizli katmandan alınan verilere karşılık her bir girdi için çıktı değerleri hesaplanır. Gizli katman ise girdi katmanından gelen verileri işler. Bir adet gizli katman ile bir problemi çözmek mümkündür. Doğrusal sinir ağlarının mimarisinde gizli katman sayısının belirlenmesi oldukça önemlidir. Bunun yanı sıra, problemde karmaşıklığın artmasıyla gizli katman sayısı artabilir, gizli katman sayısının artırılması durumunda ağın eğitimi için geçen süre artmaktadır. Genellikle ağ mimarilerinin oluşturulmasında, deneme yanılma yöntemi ile uygun olan gizli katman sayısı tespit edilir. Temel amaç, ağın beklenen çıktısı ile ürettiği çıktı arasındaki hatayı en aza indirmektir.

2.2.1 Derin öğrenme tabanlı yaklaşımlar

Derin öğrenme, istenen probleme göre en iyi çözüm üreten, yapay sinir ağları adı verilen beynin yapısı ve fonksiyonundan esinlenen ve çok hızlı bir şekilde büyüyen, makine öğrenmesinin bir alt dalıdır.

Derin öğrenme, son zamanlarda muazzam bir araştırma canlılığı yaşamış ve sayısız uygulamada son teknoloji ürünleri ile birlikte kullanılmıştır. Derin öğrenmenin yüksek boyutlu verilerin karmaşık yapılarını keşfetmekte çok başarılı olduğu ortaya çıkarılmış ve bu nedenle bilim ve iş dünyasında birçok uygulamada kullanılmaya başlanmıştır. Derin öğrenme; bilgisayar görme, nesne tanıma, nesne tespiti, video, konuşma ve ses işleme, doğal dilin işlenmesi, tıbbi, askeri ve mühendislik alanındaki uygulamalar, kodlanamayan DNA'daki mutasyonların gen ekspresyonu ve hastalığına etkileri gibi diğer makine öğrenme tekniklerinde uzun yıllar direnen sorunların çözümünde büyük ilerlemeler kaydetmektedir.

➤ Derin öğrenme çalışma prensibi

Derin öğrenme yöntemleri, her bir gösterimi piksellerden başlayarak, daha yüksek, daha soyut seviyede bir gösterime dönüştüren basit ancak doğrusal olmayan modüller oluşturarak elde edilen birden fazla gösterim düzeyine sahip temsili-öğrenme yöntemleridir. Temsili öğrenme, bir makinenin ham verilerle beslenmesine ve algılama veya sınıflandırma için gerekli olan temsilleri otomatik olarak keşfetmesine olanak tanıyan bir dizi yöntemdir. Yeterli dönüşümün kompozisyonu ile çok karmaşık fonksiyonlar öğrenilebilir. Sınıflandırma görevleri için, daha yüksek temsil katmanları girdinin önemli taraflarını güçlendirir ve alakasız varyasyonları bastırır. Örneğin, bir görüntü, bir dizi piksel değeri biçiminde gelir ve ilk katmandaki öğrenilen öznitelikler, tipik olarak, görüntüdeki belirli yönlerde ve konumlarda kenarların varlığını veya yokluğunu temsil eder. İkinci kat, kenar konumlarındaki küçük değişikliklere bakılmaksızın, kenarların özel düzenlemelerini tespit ederek motifleri algılar. Üçüncü kat, motifleri tanıdık nesnelerin parçalarına karşılık gelen daha büyük kombinasyonlara

toplayabilir ve sonraki katmanlar nesneleri bu parçaların kombinasyonları olarak algılayabilir (LeCun vd. 2015).

➤ **Derin öğrenme yapıları**

Birden çok sinir ağ katmanı kullanan bilgisayarlar derin öğrenme teknolojisi sayesinde, karmaşık durumları en az insanlar kadar iyi görme, öğrenme ve onları çözme becerisine sahiptir. Bu kısımda derin öğrenme mimarisinin dayandığı yapılardan bahsedilmiştir.

• **Otomatik kodlayıcılar**

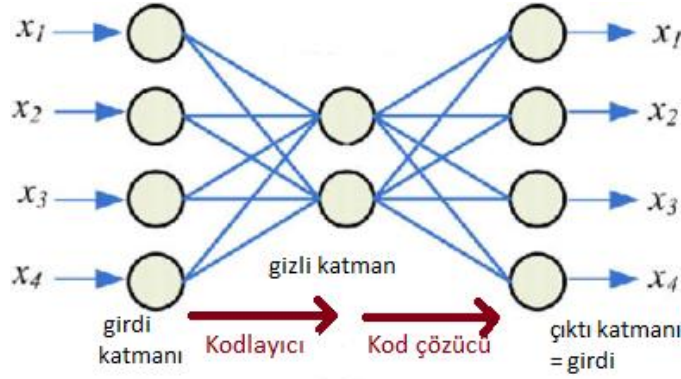
Otomatik kodlayıcı (autoencoder), yüksek boyutlu verilerin düşük boyutlu bir koda dönüştürülmesi için adaptif ve çok katmanlı bir kodlayıcı (encoder) ağ ve benzer şekilde koddan verileri kurtarmak için bir kod çözücü (dekoder) ağ kullanan çok katmanlı sinir ağıdır. Kodlayıcı ve kod çözücü ağlarında, rastgele ağırlıklardan başlayarak, orijinal veriler ile bunların yeniden yapılanması arasındaki tutarsızlığı en aza indirgeyerek birlikte eğitilebilirler. Gerekli gradyanlar, hata türevlerini önce kod çözücü ağından ve daha sonra kodlayıcı ağı üzerinden geri iletmek için zincir kuralı kullanılarak kolayca elde edilir.

Otokodlayıcılar girdi veri setini çıktı veri setine doğrusal olmayan biçimde haritalayarak (mapping) verilerin boyutsallığını azaltırlar. Boyut azaltma yönü ile PCA'lere benzerler fakat ondan daha iyi çalışmaktadırlar. Boyut azaltılması, yüksek boyutlu verilerin sınıflandırılmasını, görselleştirilmesini, iletişimini ve depolanmasını kolaylaştırır.

Bir otokodlayıcı girdi verilerinin temsilden yeniden oluşturulabilmesi için girdiyi kodlaması için eğitilir. Dolayısıyla hedef çıktı girdinin kendisidir.

Hinton ve Salakhutdinov (2006) çalışmalarında, derin otomatik kodlayıcı ağlarının, düşük boyutlu kodları öğrenmesine izin veren ağırlıkları başlatmanın etkili bir yolunu anlatmışlardır. Gradyan düşüşü (gradient descent), bu tür otomatik kodlayıcı ağlarındaki

ağırlıkları hassas ayarlamak için kullanılabilir, ancak bu, yalnızca başlangıç ağırlıkları iyi bir çözüme yakınsa iyi çalışır.

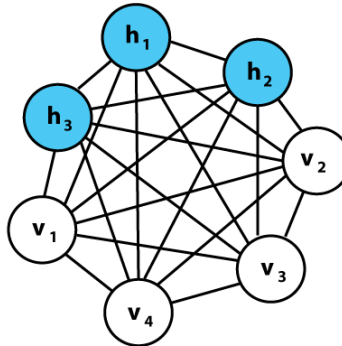


Şekil 2.9 Basit bir otomatik kodlayıcı yapısı

- **Boltzmann makineleri**

Boltzmann makinesi stokastik kararlar veren sinirlerden oluşan simetrik bağlantılı bir yapay sinir ağı olup, paralel bir hesaplama organizasyonudur. Bütün birimler birbirine bağlı olduğundan özyinelemeli bir yapıya sahiptir.

Ackley vd. (1985) çalışmalarında, Boltzmann makinesinin boltzmann dağılımları ve bilgi teorisi arasındaki yakın ilişkiyi kullanan ilginç bir stokastik model sınıfına basit bir örnek olduğunu göstermişlerdir.



Şekil 2.10 Boltzmann makine yapısı

Son zamanlarda Boltzmann Makineleri ve Derin İnanç Ağları (DİA) gibi derin öğrenme modellerinin, bazı zorlu bilgisayar vizyonu görevlerini çözmek için etkili araçlar olduğu kanıtlanmıştır. Büyük miktarda görüntü verisi göz önüne alındığında, derin öğrenme modelleri daha iyi öznitelikler öğrenebilir. Bu durum Hiper-spektral görüntüler gibi insan uzmanlığının kısıtlı olduğu veriler için önemlidir.

Termal kızılötesi görüntülerden yüz mimik tanıma son yıllarda giderek daha fazla dikkat çekmektedir. Bununla birlikte, He vd. (2013) çalışmalarında benimsedikleri öznitelikler, yüz bölgelerinden çıkarılan sıcaklık istatistik parametreleridir. He'e göre, 2013' e kadar termal kızılötesi görüntüler için özel olarak tanımlanmış görüntü öznitelikleri yoktu ve YMA için termal uzun dalga boyu kızılötesi görüntülerden termal öznitelikleri öğrenmesi için Boltzmann makineleri kullanan ilk kişi onlardı. Mevcut çalışmada ilk önce yüz, termal kızılötesi görüntülerden bulunur ve normalleştirilir. Daha sonra, iki katmandan oluşan bir Derin Boltzmann Makineleri (DBM) modeli önerilmiştir. DBMmodelinin parametreleri, öznitelik öğreniminin ön eğitiminden sonra yüz mimiğinin tanınması için daha da hassas ayarlanmıştır. NVIE veritabanında yapılan deney sonuçları karşılaştırıldığında, yaptıkları yaklaşımın sıcaklık istatistik öznitelikleri kullanılarak diğer yaklaşımlardan daha iyi performans gösterdiği kanıtlanmıştır. Ayrıca alın, ağız ve yanaktan öğrenilen öznitelikler, diğer yüz bölgelerine kıyasla tiksinti, korku ve mutluluğu ayırt etmek açısından daha güvenilirdir.

- **Derin inanç ağları**

Derin mimarilerin parametre alanını araştırmak zor bir iştir ancak son zamanlarda bu zor problemlerin üstesinden gelebilmek ve önemli başarılar elde edebilmek için Derin İnanç Ağları (DİA) gibi öğrenme algoritmaları önerilmiştir.

DİA, Hinton vd. (2006) tarafından açgözlü gözetimsiz öğrenme algoritması olarak tanımlanan stokastik, gizli değişken faktörleri ile birlikte çoklu katmanlarından oluşan olasılıksal üretken sinir ağı modelleridir. Gizli değişkenler tipik olarak ikili değerlere sahiptir ve genellikle gizli birimler veya öznitelik algılayıcıları olarak adlandırılırlar. En üstteki iki katman arasında yönsüz, simetrik bağlantılar vardır ve ilişkisel (associative)

bellek oluřtururlar. En alt katmanlar, yukarıdaki katmandan yukarıdan ařaęı doęru y nlendirilmiř baęlantılardır. En alt katmandaki birimlerin durumu bir veri vekt r n  temsil eder.

Bir katmandaki deęiřkenlerin yukarıdaki katmandaki deęiřkenlere nasıl baęlı olduęunu belirleyen yukarıdan ařaęıya,  retken aęırlıkları  ğrenmek i in verimli, katmanlı bir prosed r vardır (Hinton 2006).

 ğrenme sonrasında, her katmandaki gizli deęiřkenlerin deęerleri, alt katmanda g zlemlenen bir veri vekt r yle bařlayan ve ters y nde  retken aęırlıkları kullanan tekli bir ařaęıdan yukarıya ge iřle  ıkarılabilir.

D A, bir katmandaki gizli deęiřkenlerin deęerlerini, veriden  ıkarılırken bir sonraki katmanı eęitmek i in veriler olarak deęerlendirerek, bir kerede bir katmandan  ğrenilir. Bu etkili, a g z   ğrenme, t m aęın  retken veya fark edilebilir performansını geliřtirmek i in aęırlıkların hepsini hassas ayarlayandięer  ğrenme prosed rleri tarafından takip edilebilir veya bunlarla kombine edilebilir.

Fark edilebilir hassas ayarlama, hata t revlerinin geri yayılımı ve istenen  ıktıları temsil eden deęiřkenlerin son bir katmanını ekleyerek ger ekleřtirilebilir.  ok sayıda gizli katmana sahip aęlar, g r nt ler gibi y ksek seviyede yapılandırılmıř girdi verilerine uygulandięında, gizli katmanlardaki  znitelik detekt rleri, girdi verilerinde yapıyı modelleyen bir DBM  ğrenerek bařlatılırsa geri yayılım daha iyi  alıřır (Hinton ve Salakhutdinov 2006).

- **Kısıtlı boltzmann makineleri**

Kısıtlı Boltzmann Makine (KBM) modelini ‘Harmony’ adı altında ilk bahseden Smolensky (1986)  alıřmasıdır. KBM, bir D A’nın ayrı katmanlarıyla parametrizasyonu paylařtıęı i in DBN’nin yapı tařıdır ve verimli  ğrenme algoritmalarının eęitilmesi i in bulunmuřtur (Bengio 2009). KBM makinelerinin  ıkarımları kolaydır ve daha derin

modeller geliřtirmek iin yapı tařları olarak bařarıyla kullanılmaktadır (Roux ve Bengio 2008).

KBM, girdiler zerinde bir olasılık daėılımı ğrenebilen retken bir stokastik sinir aėıdır. KBM’ler gizli, grnr ve sapma (bias) katmanlarından oluřmaktadır. İleri beslemeli aėların aksine, grnr ve gizli katmanlar arasındaki baėlantılar ynlendirilmemektedir ve tam olarak baėlıdır (Vasilev 2017).

KBM’ler Hinton (2012)’e gre, birok farklı tip verinin retken modelleri olarak kullanılmaktadır. KBM’ler genellikle kontrastlı sapma ğrenme prosedr kullanılarak eėitilir. Bu, sayısal meta parametrelerinin deėerlerinin nasıl ayarlayacaėına belirli miktarda pratik yaparak karar vermek gerekmektedir.

- **Konvolsyonel sinir aėları**

Konvolsyonel sinir aėları probleme gre katman sayısı olan ve bu katmanlardan ıkan sonulara doėrusal olmayan fonksiyonların (tahn, ReLu vs.) uygulandıėı yapılardır. Klasik bir ileri beslemeli sinir aėı iinde, her bir ıktı nronu bir sonraki katmandaki girdi nronuna baėlanır. Buna tam baėlı katman (fully connected layer) veya afin katmanı (affine layer) denir. Fakat Konvolsyonel Sinir Aėlarında (KSA) bu durum sz konusu deėildir. Bunun yerine ıktıyı hesaplamak iin girdi katmanı zerinde konvolsyonlar kullanılmaktadır. Bu, girdinin her bir blgesinin ıktıdaki bir nrona baėlandıėı yerel baėlantılarla sonulanır. Her katman farklı filtreleri uygular ve sonunda bunların sonucu birleřtirilir (Britz 2015).

- **KSA ile yapılan alıřmalar**

KSA’nın ortaya ıkıřı 20 yıl ncelerine dayanmaktadır ve bu aė yapısı ile ilgili ilk alıřmalardan biri yz tanımadır (Lawrance vd. 1997). Son zamanlarda derin sinir aėlarının, grntlerden yze ait mimikleri tahmin etme grevinde rekabet gcnn

yüksek olduğu gösterilmiş ve KSA ile yüz mimik tanıma çalışmaları özellikle 2015 tarihinden başlayarak yoğunlaşmıştır.

Devries vd. (2014) çalışmalarında aynı anda yüzün önemli noktalarını ve yüz mimiklerini tahmin eden çok görevli bir KSA sunmuşlardır. ICML 2013 Yüz Mimik Tanıma veri setinde ve ayrıca Toronto Yüz veri setinde yapılan deneylerde, yüze ait önemli noktaların konumu hakkında mantık yürütmenin mimik tanımayı geliştirdiği gösterilmiştir.

Derin sinir ağı modellerden en iyi öznitelikler çıkarılmasına rağmen, biraz büyük bir gizlemede (occlusion) iyi bir performans göstermezler. Bu performansı iyileştirmek için Xu vd. (2015) çalışmalarında, derin konvolüsyonel ağlardan transfer öğrenmeye dayalı etkin bir yüz mimik tanıma modeli önermişlerdir. Yüze ait MSRA-CFW veritabanı üzerinde 1580 sınıf yüz kimliği KSA ile eğitilmiştir. Eğitilmiş KSA'lerden 120 boyutlu yüksek seviyeli öznitelikler çıkarılmıştır. Mevcut modeldeki KSA'lar 4 adet konvolüsyonel katmanlardan oluşmaktadır. Bu model aynı yapıda iki eğitilmiş derin KSA'dan transfer edilen yüksek seviyeli öznitelikleri birleştirilmiştir. En sonunda öznitelikler yedi adet mimik durumuna DVM ile sınıflandırılmıştır. MSRA-CFW, CK+ ve JAFFE veritabanlarının birleşimi ile oluşturdukları veritabanlarında modellerini test etmişler ve %81.50 oranında bir başarı elde etmişlerdir.

Derin Sinir Ağları (DSA) sadece tam bağlantılı katmanlardan oluşurken, KSA, ağırlıklılarını paylaşan yerel bağlantılı katmanlardan oluşmaktadır. Böylece, KSA daha çok görüntü tanıma problemlerinde kullanılırken, DSA ses tanıma gibi sinyal işleme problemlerinde kullanılmaktadır. Fakat Jung vd. (2015) çalışmalarında, yüz mimik tanıma için DSA'ını da kullanmışlardır. Çünkü girdi görüntüsü çok geniş değildir, bu yüzden DSA için tam bağlantılı katmanların eğitimi zor değildir. FER 2013 veritabanı kullanılarak DSA ve KSA eğitilmiş ve sonuçları karşılaştırılmıştır. Sonuç olarak KSA'nın, DSA'ından mimik tanıma oranı daha yüksek çıkmıştır.

Deng vd. (2015) çalışmalarında, daha gerçekçi şartlarda bir sistem geliştirmek için Real-world Affective Face Database (RAF-DB) sosyal ağlardan yaklaşık 30,000'e yakın

çeşitli yüze ait görüntüleri içeren veri tabanı kullanmışlardır. DeepEmo adını verdikleri, geliştirdikleri derin öğrenme mimarisinde, mimik sınıflandırması için klasik KSA'ına kıyasla iki yeni yapılandırma vardır. İlk olarak, DeepEmo'nun ilk katmanı alın, göz ve kaş, burun, yanak ve ağıza karşılık gelen yerel olarak dört bölgeye bölünmüş konvolüsyon tabakasıdır. Tabaka, farklı yüz bölgelerinde eylem birimlerini belirlemek için 4 kez belirli filtre sayısını öğrenmeyi amaçlamaktadır. İkincisi, 70 birimden oluşan bir ML-Softmax çıktı katmanı; olası tüm mimik etiket kombinasyonlarına karşılık gelen olasılıkları çıkartmaktadır. ML-Softmax'ın amacı, sadece altı temel mimiği sınıflamak yerine, mimiklerin hafif değişimini daha ayırt edilebilen öznelilikleri öğrenmektir. Kapsamlı deneysel sonuçlar, derin öğrenme yönteminin YMA için bu tür ek yapılarla geliştirilerek insan düzeyinde tanıma doğruluğuna erişebileceğini göstermektedir.

Shin vd. (2016), yüz mimik tanıma algoritmasını geliştirmek için temel bir KSA yapısı ve görüntü ön işleme yöntemini sunmuşlardır. En etkili ağ yapısını analiz etmek için, yüz mimik tanımda iyi performans gösteren dört ağ yapısı araştırılmıştır. Ayrıca girdi görüntüsü ön işleme metotlarının etkisi araştırılmıştır. Beş veri girdi tipi (ham, histogram eşitlemesi, izotropik yumuşatma (smoothing), difüzyona dayalı normalleştirme, Gauss farkı) test edilmiş ve doğruluk karşılaştırılmıştır. 20 farklı KSA modeli ($4 \text{ ağ} \times 5 \text{ veri girdi türü}$) eğitilmiş ve her bir ağın performansı beş farklı veritabanından test görüntüleri ile doğrulanmıştır. Önemli gözlemlere dayanan deney sonuçlarına göre, histogram eşitleme görüntü girdisi ile basit bir konvolüsyon katmanı ve maksimum havuz katmanından oluşan üç katmanlı bir yapının en etkili olduğu ortaya konmuştur.

Raghuvanshi ve Choksi (2016), YMA çalışmaları için KSA'ları kullanmışlardır. Çalışmalarında ön eğitilmiş VGG16 modelini kullanmışlardır. Yedi adet mimik kategorilerine sınıflandıran çalışma sonucunda % 48 civarında doğruluk elde etmişlerdir.

2.3 Sınıflandırma

Yüz bölgesinden çıkarılan öznitelikler kümesi sınıflandırma aşamasında kullanılır. Bu öznitelikler kümesi, mimikleri tanımlamak için kullanılır. Sınıflandırma, denetimli eğitim gerektirir, bu nedenle eğitim kümesi etiketli verilerden oluşmalıdır. Sınıflandırıcı eğitildikten sonra, girdi görüntüleri belirli bir sınıf etiketi atayarak tanıyabilir.

En sık kullanılan mimik sınıflandırması yaklaşımları prototipik mimik yaklaşımları ve eylem birimlerini kullanan yaklaşımlardır. İlki zihinsel aktivitelerle belirgin yüze ait örüntüleri doğrudan ilişkilendirirken, ikincisi, nihai yorumlama girişimleri öncesinde yüz mimiklerini kodlanmış bir biçimde yüze ait hareketlerle temsil etmektedir.

2.3.1 Prototipik mimikler

Ön cepheden çekilmiş tüm yüzü kullanmaya dayanan bir yaklaşımdır. Ayrıntılı yüz özniteliklerini tanımlamak yerine, yüzün tüm görüntüsü olduğu gibi kullanılır ve Ekman ve Friesen (1971) tarafından tanımlanmış yedi adet prototipik mimiği sınıflandırmak için işlenir. Ekman'ın teorisine göre, farklı ulusların ve kültürlerin insanları için evrensel olan yedi temel duygu ifadesi vardır. Bu temel duygular öfke, tiksinti, korku, mutlu, üzgün, şaşkın ve nötrdür.

Yacoob ve Davis (1996), Lien (2008), Ahmed (2014), Sajjad (2017) mimik analiz yaklaşımları, mimikleri Ekman ve Friesen'in ortaya koyduğu temel yedi adet duygu sınıflarından birine direkt olarak eşleştirmeyi amaçlar.

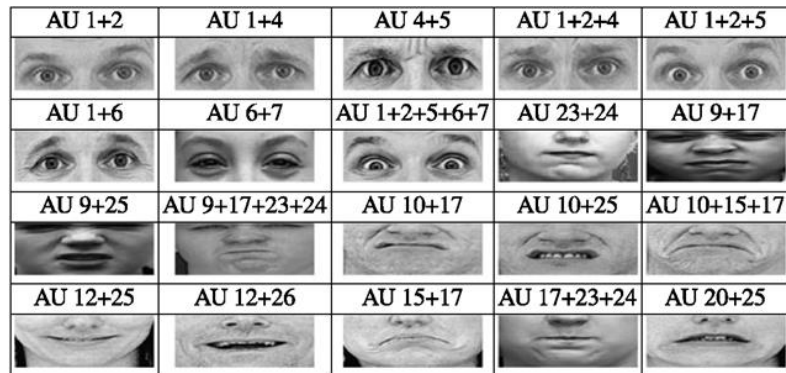
2.3.2 Eylem birimlerini kullanma

Bu yaklaşımda görüntü alt bölümlere bölünür ve daha sonra mimik analizi için kullanılır. Yüz hareketi kodlama sistemi (FACS) diye adlandırılan bu yaklaşım Ekman ve Friesen (1978) tarafından öne sürülmüştür.

Bazı mimiklere bağlı olarak ortaya çıkan kas kasılmaları bir Eylem birimi (AU) olarak işaretlenmiştir. Eylem birimleri, bir kasın veya kasların bir kombinasyonunun neden olduğu yüzdeki değişikliklerdir. FACS kullanımı ile mimik analizi görevi, gözlemlenen mimiğin eylem birimlerinin kümesine ayrılmasına dayanır. Toplam 44 eylem birimi vardır ve bunlardan 30'u belli kasların kasılması ile elde edilmiştir. Bu 30 eylem biriminden 12'si yüzün üst kısmı, geri kalan 18'i ise yüzün alt kısmındadır. Yüz mimikleri tanıma için etkili bir yöntemdir; görüntüyü objektiflik ve esneklikle uygulayarak yüzü hareket birimlerine böler. Bu yaklaşım, mimiklerdeki hassas seviyedeki değişikliklerin tanımlanması gereken uygulamalara uygulanır.

Huang (1999), Bartlett (2004), Whitehill ve Omlin (2006), Kotsia (2007) yüz mimik analiz çalışmalarında yüz mimiklerini kodlanmış bir biçimde yüze ait hareketlerle temsil etmektedirler.

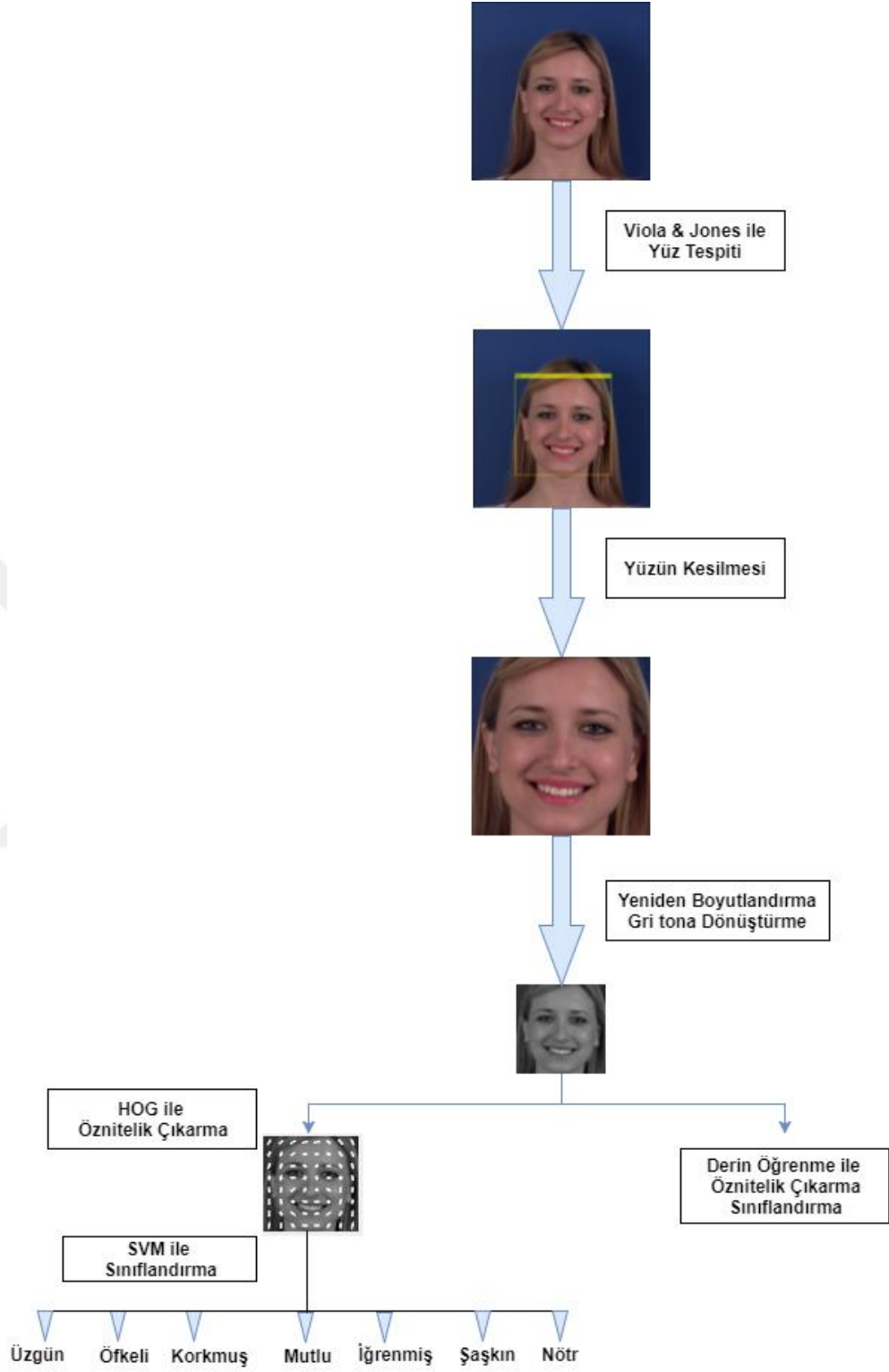
Khorrami vd. (2015) çalışmalarında, hem konvolüsyonel sinir ağlarının güçlü performans elde edebildiğini göstermişler hem de yüzün hangi kısımlarının KSA'nın tahminlerini etkilediğini tespit etmek için bir yaklaşım getirmişlerdir. İlk olarak, CK+ ve Toronto Face Dataset (TFD) veritabanlarından alınan yüz mimik verileri üzerinde sıfır eğimli bir KSA eğitilmiştir. Daha sonra, ağın konvolüsyonel katmanlardaki farklı filtreleri en çok uyaran uzaysal örüntüleri görselleştirilmiş, Yüz hareketi birimlerine (FAU) nasıl benzediğini göstermek suretiyle ağ analiz edilmiştir. Son olarak, filtre görsellemelerinde gözlemlenen FAU'ların kişinin yüz hareketleriyle hizalandığını doğrulamak için CK + veri setinde verilen FAU etiketleri kullanılmıştır.



Şekil 2.12 FACS eylem birimlerinin bazı örnek kombinasyonları

3. GELİŞTİRİLEN YÖNTEMLER

Bu tez çalışması kapsamında iki adet yüz mimik tanıma sistemi geliştirilmiştir. İlk çalışma HOG-SVM temelli Yüz Mimik Tanıma Sistemidir. İkinci çalışma ise Derin Öğrenme ile Yüz Mimik Tanıma Sistemidir. Yüz mimikleri tanıma problemi çok aşamalı bir işlemler dizisidir. Bu bölümde adım adım yüz mimikleri tanıma sistemi oluşturulmuştur. İlk adım, girdi olarak alınacak resimlerden sadece yüz kısmının alınıp diğer faktörlerin (saç, boyun, arka plan vs.) yok edilmesidir. Bu aşamaya yüz tespiti denmiştir. İkinci adım ön işlemler kısmı olup, bu ön işlemler; resimlerden elde edilen yüz görüntülerin hepsinin renk, boyut vs. olarak aynı standartlarda olmasını sağlamaktadır. Bu amaçla tüm renkli görüntüler gri tonlamalı görüntüye dönüştürülüp 64x64 aynı boyuta indirilmiştir. Böylelikle işlem kalitesi artacak ve işlem süresi önemli ölçüde düşecektir. Aynı şartlar sağlanıp işlem için hazır hale getirilen tüm görüntülerin ilk önce HOG-SVM bazlı sistemde yüz öznitelikleri çıkarılmış ve ‘üzgün’, ‘mutlu’, ‘şaşkın’, ‘iğrenmiş’, ‘sinirli’, ‘korkmuş’, ‘nötr’ olmak üzere yedi adet mimik sınıflarından birine sınıflandırılmıştır. Yine aynı şekilde yüz tespiti yapılan ve ön işlemlerden geçirilen tüm görüntülerin Derin öğrenme yapısının özelleştirilmiş bir yapısı olan konvolüsyonel sinir ağları ile öznitelikleri çıkarılmış ve yedi adet mimik sınıflarından birine sınıflandırılmıştır.



Şekil 3.1 Geliştirilen yaklaşımın genel bir akış şeması

3.1 Yüz Tespiti

Yüz tespiti, insanların kolaylıkla yapabileceği görsel görevlerden biridir. Bununla birlikte, bilgisayarda görme terimlerinde bu çok da kolay olmayan bir görevdir. Yüz tespiti probleminin genel amacı; üzerinde çalışılacak olan sabit veya video görüntüsünden istenmeyen tüm arka plan, saç, boyun gibi tüm faktörleri kaldırmak ve daha sonraki işlemler için yalnızca yüzü çıkarmaktır.



Şekil 3.2 Yüz tespiti yapılmış görüntülere birkaç örnek (Rowley vd. 1998)

Yüz tespiti konusundaki ilk çabalar, basit, sezgisel ve antropometrik teknikler kullanan Sakai vd. (1972) çalışmaları ile başlamıştır. Bu teknikler, sade arkaplan, ön cephe yüz, tipik bir pasaport fotoğrafı gibi kısıtlı durumlarda çalışmakta olup çeşitli varsayımlardan birinin (örn; görüntü şartlarındaki herhangi bir değişiklik) ufak bir değişimi ile herhangi bir başarı söz konusu olmamıştır. Bu sebeple 1990'lı yıllara kadar bu alana büyük bir ilgi duyulmamıştır. Chellappa vd. (1995) çalışmaları, yüz tespiti yöntemleriyle ilgili önemli ve açıklayıcı bir çalışma olmuştur. Bu yıllardan itibaren üzerinde çalışmalar yapılan yüz tespiti, son yıllarda oldukça popüler bir araştırma konusu olup; düşük kalitede gri tonlamalı görüntüler ve karmaşık arka planlar gibi ortamlarla uğraşan algoritmalarda büyük ilerlemeler katedilmiştir.

Yüz tespiti teknolojisi, yüz tanıma, cinsiyet tespiti, ırk tespiti, yaş tespiti vb. gibi konuların temelini oluşturmakta olup; biyometrik tanımlama, video konferans, görüntü

ve video veritabanlarının indekslenmesi, akıllı insan-bilgisayar arabirimleri gibi geniş bir uygulama yelpazesinde yaygın bir şekilde kullanılır olmuştur.

Yüz mimiklerinin tanınabilmesi için öncelikle resimlerdeki yüzlerin bulunması gerektiğinden yüz tespiti çalışmanın ilk adımını oluşturmaktadır. Yüz tespiti için çok sayıda yöntem mevcuttur, fakat yüksek algılama oranı ve gerçek zamanlı çalışabilme özelliği nedeniyle en popüler yöntemlerden biri Viola Jones algoritması bu çalışmada kullanılmıştır.

3.1.1 Viola ve Jones algoritması

Viola Jones algoritması, Viola ve Jones tarafından nesne tespiti için ortaya çıkarılmış, sıklıkla yüz tespitinde kullanılan yüksek başarı oranı sağlayan bir yöntemdir (Viola ve Jones 2001). En temel manada resim kayan pencerelerle taranarak tespit edilecek yüz bulunmaya çalışılır.

Bu algoritmanın dört önemli aşaması mevcuttur.

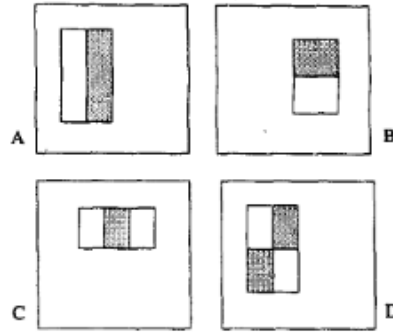
- Öznitelik çıkarma
- İntegral imge
- Öznitelikleri kullanarak öğrenme
- Sınıflandırıcıları kademelendirme

➤ Öznitelik çıkarma

Tespit edilmesi istenen yüzün tüm pikselleri üzerinde işlem yapmak hem zor hem de oldukça fazla zaman kaybına neden olacaktır. Bunun yerine görüntüden önemli öznitelikler çıkarılarak yüz ya da yüz değil diye problemi çözmek makul bir çözüm olacaktır. Bu algorithmada iki dikdörtgenli, üç dikdörtgenli, dört dikdörtgenli olmak üzere üç tip dikdörtgen, ‘haarlike öznitelikler’ adı verilen özniteliklerin çıkarılmasında kullanılacaktır. Bu öznitelikleri bulurken tek bir dikdörtgen ile değil, birden fazla ve

döndürülmüş dikdörtgenler kullanılır. Herhangi bir öz niteliği elde etmek için basit bir şekilde koyu renkli kısımdan açık renkli kısımlar çıkarılır (Şekil 3.3).

Bu şekilde bir dizi öz nitelik elde edilmiş olur.



Şekil 3.3 Öz nitelik çıkarmak için kullanılan dikdörtgen çeşitleri (Viola ve Jones 2001)

➤ **İntegral imge**

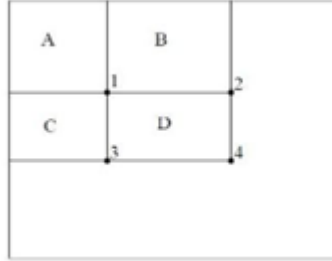
Dedektörün tüm resim üzerinde dolaştırılmasıyla tüm piksellerin toplanması çok büyük miktarlarda öz nitelik kümesi verecektir. Fakat integral imge mantığı ile bu hesaplamalar oldukça hesaplı ve hızlı yapılabilmektedir. Integral imgenin (x,y) konumundaki değeri, (x,y)'nin üstünde ve solunda kalan alanın piksel değerleri toplamıdır. Şu formülle ifade edilir:

$$ii(x,y) = \sum_{x^1 \leq x, y^1 \leq y} i(x^1, y^1) \quad (3.1)$$

Burada $ii(x,y)$ integral imgeyi, $i(x,y)$ ise orijinal imgeyi göstermektedir.

Aşağıda integral imgenin hesaplanması basitleştirilmiş bir şekilde şematize edilmiştir. Oluşturulan integral imgedeki her bir piksel, kendisinin sol ve yukarısındaki piksellerin değerleri toplamı olup dört referansla hesaplanmıştır. 1 konumundaki integral imgenin değeri, A dikdörtgeni içindeki piksel değerleri toplamıdır. 2 konumundaki değer A+B, 3

konumundaki değer $A+C$, 4 konumundaki değer $A+B+C+D$ 'dir. D içindeki toplam, $4+1-(2+3)$ olarak hesaplanır.



Şekil 3.4 İntegral görüntünün oluşturulma esası (Viola ve Jones 2001)

➤ Öznitelikleri kullanarak öğrenme

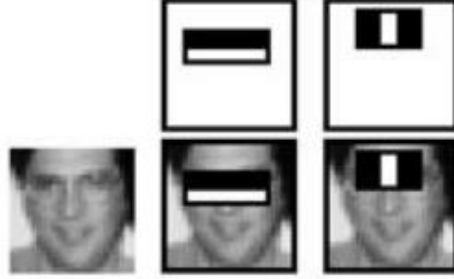
İntegral imge sayesinde işlemler biraz daha hızlı ve kolay olmuştur. Fakat yine de elde edilen dikdörtgensel öznitelikleri hesaplamak oldukça zahmetli olacaktır. Ancak bu öznitelikler etkili bir sınıflandırıcı ile kombine edilerek çok küçük bir sayıya dönüştürülebilecektir. Bu aşamada, hem bu özniteliklerin en iyilerini seçmek hem de sınıflandırıcıyı eğitmek için tasarlanan Adaboost adlı bir zayıf öğrenme algoritması kullanılacaktır.

$$h_j(x) = \begin{cases} \text{eğer } p_j f_j(x) < p_j \theta_j & \text{ise } 1 \\ \text{değilse} & 0 \end{cases} \quad (3.2)$$

Bu denklemde, $h_j(x)$ bir zayıf sınıflandırıcı fonksiyon, f bir öznitelik, x , 24×24 boyutlarında görüntünün bir alt penceresi, p_j benzerlik, θ_j eşik değeri temsil etmektedir. Bu zayıf sınıflandırıcı minimum sayıda yanlış sınıflandırılmış örnek veren eşik değerli bir fonksiyondur. Bu fonksiyonlar bir öznitelikmiş gibi düşünülüp pozitif ve negatif örnekleri en iyi şekilde ayıran öznitelikler aranmaktadır.

Şekil 3.5'te iki dikdörtgen bulunmaktadır. Bu dikdörtgenler alt sıradaki eğitim yüzeyi üzerine bindirilir. AdaBoost tarafından seçilen birinci öznitelik, göz bölgesi ile üst yanaklardaki bir bölge arasındaki yoğunluğun farkını ölçer. Bu öznitelik, göz bölgesinin

genellikle yanaklardan daha koyu olduđu gözlemini esas alır. Seçilen ikinci öznelik, göz bölgelerindeki yoğunluđu burun köprüsündeki yoğunluđa kıyaslar.



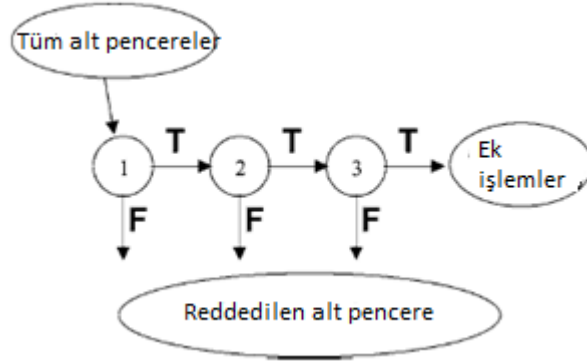
Şekil 3.5 AdaBoost tarafından seçilen öznelikler (Viola ve Jones 2001)

➤ Sınıflandırıcıları kademelendirme

Bu kısımda elde edilen özneliklerden anlamlar çıkarılması ve görüntüde yüzün var olup olmadığını anlamak amacıyla kademeli çalışan sınıflandırıcılar oluşturulmuştur. Güçlü bir sınıflandırıcı oluşturmak için zayıf sınıflandırıcılar art arda kullanılmıştır. Zayıf sınıflandırıcılar, karmaşıklıklarına göre sıralı biçimde birbirlerine eklenmişlerdir. Kademeli bir sınıflandırıcı yapısı oluşturulmasındaki amaç ise özneliklerden istenmeyenlerinin ön elemelerden geçirilerek boş yere işlem yapılmasını engellemektir. Bu sınıflandırıcılar istenilen özellikler ile eğitilerek, girdi olarak aldıkları görüntülerde istenilen alanların mevcut olup olmadıklarını kontrol ederler. Eğer incelenen alt pencere herhangi bir sınıflandırıcı tarafından kabul edilmezse bu alt pencere üzerinde daha fazla işlem yapılmaz. Böylece işlem ve zamandan tasarruf edilmiş olunur.

Algılama sürecinin genel biçimi, kademe denilen bir karar ağacıdır (Şekil 3.6). Birinci sınıflandırıcıdan gelen pozitif bir sonuç, çok yüksek algılama oranlarına ulaşmak için ayarlanan ikinci bir sınıflandırıcının değerlendirilmesini tetikler. İkinci sınıflandırıcıdan gelen olumlu bir sonuç, üçüncü bir sınıflandırıcıyı tetikler ve böyle devam eder. Herhangi bir noktada negatif bir sonuç, alt pencerenin acil reddedilmesine yol açar. Kademeli basamaklar AdaBoost kullanılarak sınıflandırıcıların eğitimi ve daha sonra yanlış negatiflerin en aza indirilmesi için eşik değerini ayarlayarak oluşturulur. Varsayılan AdaBoost eşik, eğitim verileri üzerinde düşük bir hata oranı elde edecek

şekilde tasarlanmıştır. Genel olarak düşük bir eşik, daha yüksek algılama oranları ve daha yüksek yanlış pozitif oranlar verecektir.



Şekil 3.6 Algılama kademesinin şematik tasviri (Viola ve Jones 2001)

Her alt pencereye bir dizi sınıflandırıcı uygulanır. İlk sınıflandırıcı çok az işlem ile çok sayıda negatif örnekleri ortadan kaldırır. Sonraki katmanlar ek negatifleri ortadan kaldırır, ancak ek hesaplama gerektirir. Birkaç işlem aşamasından sonra alt pencerelerin sayısı radikal olarak azaltılır. Daha sonraki işlem, kademenin ek aşamaları veya alternatif bir algılama sistemi gibi herhangi bir biçimi alabilir (Viola ve Jones 2001).

3.2 Ön İşlemler

Görüntü işleme, görüntünün dijital ortama aktarılarak sayısallaştırılması ve üzerinde çeşitli işlemler yapılabilir hale getirilmesidir. Günümüzde görüntü işleme, hemen hemen her alanda kullanılmaktadır. Özellikle, son yıllarda bilgisayarın her alanda kullanılmasıyla çok önemli bir konuma gelmiştir.

Geliştirilecek YMA çalışmasında da görüntü işlemeden faydalanılmıştır. Görüntü işleme tekniğinin ilk adımı olan ön işlemler yapılarak girdi görüntüsü önemli işlemler için uyumlu hale getirilmiştir. Görüntü, farklı donanımla farklı koşullar altında alınabilir. Bu yüzden elde edilen görüntüler standart bir formda olmayabilir. Girdi görüntüsü, gürültü, aydınlatma değişimi, arka planlar vb. gibi çeşitli karışıklıklardan da etkilenebilir. Ayrıca

görüntülerin boyutları değişebilir. Bu nedenle, YMA'nın ön aşamasında görüntüler normalleştirilir.

Bu aşamada yapılacak olan işlemler sıralanacak olursa;

- 1) Yüz tespiti yapılmış görüntüler, gri tonlamalı görüntüye dönüştürülecektir (aydınlatılacaktır).
- 2) Görüntüler belirli bir boyuta standardize edilecektir.

3.2.1 Gri tonlamalı görüntüye dönüştürme

Kanal, bir görüntünün bileşenlerini ifade etmek için kullanılan bir terimdir. Standart bir dijital kameradan alınan bir resim, kırmızı, yeşil ve mavi olmak üzere üç kanala sahip olacaktır. Bunlar her biri [0 255] aralığında piksel değerlerine sahip olan üç adet 2 boyutlu matristir.

Öte yandan, gri tonlamalı bir görüntünün yalnızca bir kanalı vardır. Bu çalışmada yalnızca gri tonlamalı görüntüler ele alınacak, bu nedenle bir görüntüyü temsil eden tek bir 2B matris olacaktır. Bu matris, yalnızca parlaklık bilgisi içermekte olup, renk bilgisi içermemektedir. Matristeki her gri tonu (parlaklığı) 0 ile 255 değerleri arasında değişmektedir.

Bu aşamada renkli olan görüntüler MATLAB'da rgb2gray fonksiyonu ile gri seviyeli görüntüye dönüştürülmüştür. Böylece görüntülere aydınlatma uygulanarak, görüntü üzerindeki parlaklıkların hepsi aynı seviyeye getirilmiş, resmin yoğunluk değerleri her pikselde aynı olmuştur.

3.2.2 Yeniden boyutlandırma

Yüz tespiti yapılmış görüntüler gri seviyeli görüntülere dönüştürüldükten sonra tüm görüntüler, 64 x 64 olacak şekilde aynı boyuta normalize edilmiştir.

3.3. HOG-SVM Tabanlı Yüz Mimik Tanıma Sistemi

Çalışmada kullanılacak veritabanındaki tüm resimlerden yüz tespiti yapıp kırılmıştır. Elde edilen bu yüz görüntüleri MATLAB'ın görüntü işlem adımlarından geçirilip sıradaki adımlara hazır hale getirilmiştir.

Öznitelik, temel olarak görüntüye ait ölçülebilir ya da gözlenebilir bilgi şeklinde ifade edilebilir. Örneğin, ses sinyali için sinyalin frekans bilgisi öznitelik olurken, çeşitli çalışmalara tabi olan bir yüzün öznitelikleri, kaşlar, gözler, burun ve ağzın belirgin özellikleri olacaktır. Bu öznitelikler, görüntünün sınıfını niteleyen gerçel sayılar veya vektörler kümesi şeklinde olabilir. Öznitelik çıkarımı ise, gereksiz bilgiyi eleyerek görüntüye ait karakteristik özelliklerin elde edilmesini sağlar. Temel anlamda bir çeşit boyut indirgeme işlemidir. Başarılı bir YMA için en kritik konu, önemli özniteliklerin seçilmesi ve sağlam, etkin bir yüz öznitelik tanımlayıcısı tasarlamaktır. Bu, öznitelik çıkarma aşamasından sonra kullanılan sınıflandırma algoritmaların hızlı çalışmasını, zaman kazancını ve belleğin gereksiz yere kullanılmamasını sağlayacaktır. Sağlam ve etkili öznitelikleri çıkarmak için birçok algoritma geliştirilmiştir. Bu YMA çalışmasının ilk yaklaşımında öznitelik çıkarmak için HOG algoritması kullanılmıştır.

3.3.1 HOG özniteliklerinin kullanımı

Bu çalışmada başarılı bir öznitelik çıkarma yöntemi olan Yönlü Gradyan Histogramları (HOG) kullanılarak eğitim ve test verileri için öznitelik vektörleri elde edilecektir. Daha sonra elde edilen bu vektörlerden sınıflandırma yapılarak insanın yüz mimik analizi gerçekleştirilecektir.

HOG, bir görüntüdeki yoğunluk değişimlerinin yönlerine göre hesaplanması ile oluşturulan histogramlardan meydana gelen gradyan tabanlı bir yöntemdir. HOG ilk olarak Dalal ve Triggs (2005) çalışmalarında önerilmiştir. İlk olarak nesne tespiti için geliştirilen bu yöntem çok geniş uygulama alanlarında kullanılmaktadır.

Bu öznitelik tanımlayıcı algoritması birkaç aşamadan oluşmaktadır.

➤ Gradyan ve yönelim histogramların hesaplanması

Bu aşamada görüntünün yatay ve dikeydeki gradyanları hesaplanır. Bunlar, aydınlatma varyasyonlarına daha fazla direnç sağlayan kontur, siluet ve bazı doku bilgilerini yakalarlar.

Ön işlemler aşamasında gri tonlamaya dönüştürülmüş görüntüler, Sobel filtresi $[-1, 0, 1]$ kullanılarak yatay gradyanı G_x ve dikey gradyanı G_y hesaplanır.

$$G_x(x, y) = I(x + 1, y) - I(x - 1, y) \quad (3.3)$$

$$G_y(x, y) = I(x, y + 1) - I(x, y - 1) \quad (3.4)$$

Daha sonra bulunan bu gradyan bileşenleri kullanılarak görüntünün gradyan büyüklüğü ve gradyan yönelimleri hesaplanır.

$$|G| = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \quad (3.5)$$

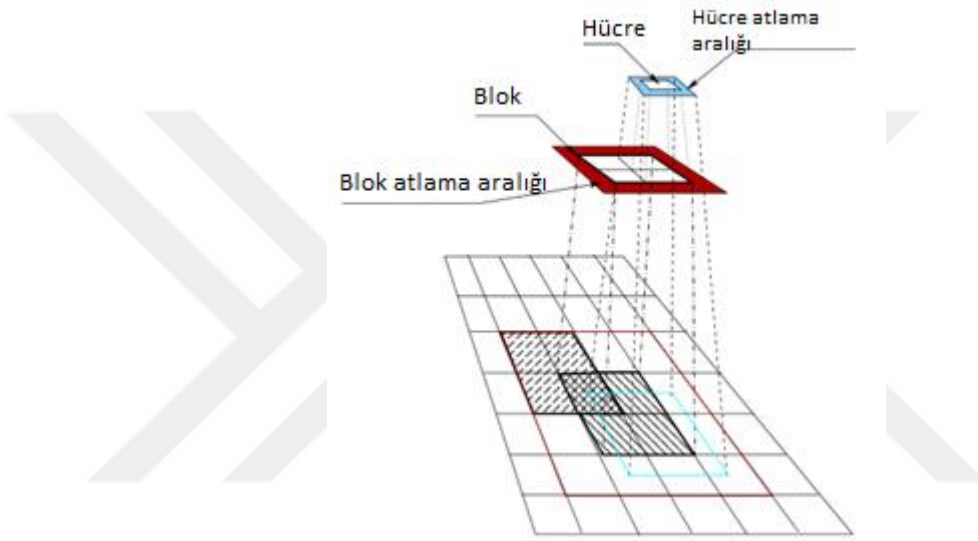
$$\varphi = \tan^{-1} \frac{G_y(x, y)}{G_x(x, y)} \quad (3.6)$$

➤ Hücre ve blok büyüklüğü

Gradyan büyüklüğü ve gradyan yönelimi hesaplandıktan sonra, gradyan görüntüsü 8×8 (8 satır, 8 sütun) büyüklüğünde hücrelere bölünmüştür. Bu hücreler iki boyutlu bir vektördür. Hücre boyutu artırılarak büyük ölçekli uzaysal bilgiler yakalanabilir. Fakat bu kez küçük ölçekli detaylar kaybedilebilir.

Hücredeki blok boyutları ise 2x2 büyüklüğünde oluşturulmuştur. Küçük blok boyutu, HOG özniteliklerinin aydınlatma değişikliklerini önlemeye yardımcı olur.

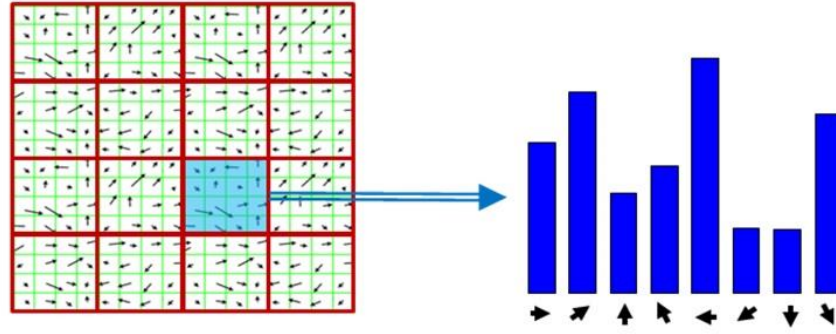
Yeterli kontrast normalizasyonunu sağlamak için, bitişik bloklar arasında çakışan (overlapping) hücrelerin sayısı blok büyüklüğünün en az yarısı kadar olmalıdır. Bu çalışmada blok büyüklüğü 2 olduğundan çakışan hücre büyüklüğü 1x1 şeklindedir. Böylece bir blok kullanıldığından çakışma meydana gelmez.



Şekil 3.7 HOG öznitelik seviyeleri

➤ Yönlendirme kutulama (Orientation binning)

Her bir hücrenin yönelim histogramını elde etmek için yönelim açıları, yön histogramlarının bin sayısı kadar farklı açı aralığına bölünür. Mantıksal skaler olarak belirtilen yönlendirme değerinin seçimi true olarak ayarlandığında, yönlendirme değerleri -180 ile 180 derece arasında eşit aralıklarla yerleştirilir. Bu özellik false olarak ayarlandığında yönelim histogramları, 0 ile 180 derece arasında eşit aralıklarla yerleştirilir. Bu çalışmada işaretsiz yönlü histogramlarının binleri false ayarlanmıştır ve binlerin sayısı dokuzdur. Bu bin sayısının değerini artırmak, öznitelik vektörünün boyutunu artırır, bu da işlem için daha fazla zaman gerektirir. Böylece yönelim histogramları [0 180] arası 9 eşit parçaya bölünmüştür.



Şekil 3.8 Yönelim histogramları (Gill 2013)

➤ Blok normalizasyonu

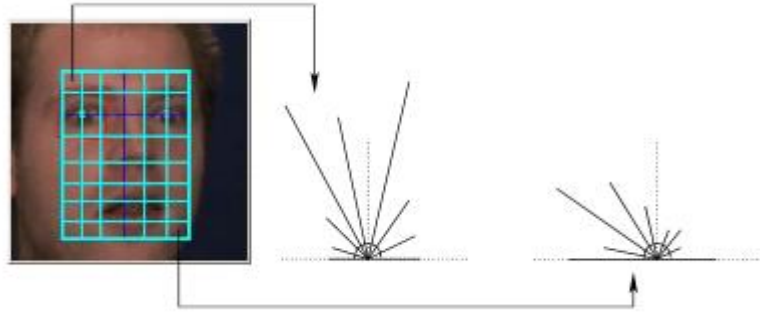
Şu ana kadar yapılan işlemler bir blok içerisinde oluşturulan histogramlardı. Tüm bu histogramların birleştirilmesiye büyük bir histogram elde edilir. Bu aşamaya blokların normalizasyonu denir. HOG için 4 farklı blok normalizasyon biçimi vardır. Çalışmada L2 norm blok normalizasyonu kullanılmıştır. Bu norm aşağıdaki eşitsizlikle mevcuttur.

$$f = \frac{v_k}{\sqrt{\|v_k\| + \epsilon^2}} \quad (3.7)$$

Bu eşitsizlikte $k=1,2$, ve ϵ sabit değerleri için v_k , bir bloktan elde edilmiş normalize edilmiş histogram vektörü, v normalize edilmemiş HOG öznitelik vektörüdür.

➤ Blok geometrisi

HOG algoritmasında kullanılan blokların bir geometrisi söz konusudur. Kare veya dikdörtgen parçalara bölünmüş blok geometrisine R-HOG, dairesel parçalara bölünen blok geometrisine C-HOG denmiştir. Bu çalışmada histogram bölmeleri, dikdörtgen hücreler (R-HOG) üzerinden hesaplanmıştır.



Şekil 3.9 Her bir hücreden çıkarılan yerel HOG'ları birleştiren global bir HOG (Dahmane 2011)

Bu çalışmada HOG özniteliklerinin çıkarılması için MATLAB programının Computer Vision System Toolbox'ında `extractHOGFeatures` fonksiyonu kullanılmıştır. Görüntünün her pikseli için yoğunluk değişiminin yönü ve büyüklüğü hesaplanmıştır. Gradyan büyüklüğü ve yönü hesaplanan görüntü ilk önce hücre olarak adlandırılan küçük bağlı bölgelere bölünmüştür. 8x8 büyüklüğünde her bir hücre için kenar yönelimlerinin histogramı/yönlere göre değişimleri gösteren histogramlar elde edilmiştir. Elde edilen histogramlar, gradyanların 'işaretsiz' olmasına bağlı olarak 0-180 derece arasında eşit olarak yayılmıştır. Bu yerel histogramlar toplanmış ve sonuçları bloğun tüm hücrelerinde normalize edilmiştir. Bu histogramların birleşimi en son HOG tanımlayıcıyı temsil etmiş ve öznitelik vektörleri çıkarılmıştır.

HOG'un iyi performans göstermesinin nedeni, özellikle yerel kenarlardaki yönlendirme gruplamasına ve buna bağlı olarak gradyan büyüklüğüne dayanan global özellikleridir.

3.3.2 DVM ile sınıflandırma

YMA'nin son aşaması, çıkarılan öznitelikleri istenen sonuca göre sınıflandırmaktır. Sınıflandırma işlemi için, hangi sınıfa ait olduğu önceden bilinen belirli sayıdaki öznitelik vektörünün oluşturduğu veri kümeleri bir eğitim sürecinden geçirilir. Bu eğitim sonucunda, bilinmeyen görüntüyü uygun sınıfa atamakta kullanılan bir karar mekanizması oluşturulur. Bu sınıflandırma yaklaşımı gözetimli öğrenme ve gözetimsiz öğrenme şeklindedir (Bkz. Bölüm 2.1).

Sınıflandırma aşamasının amacı, örüntüleri öznitelik uzaylarına göre kendilerine en yakın sınıflara minimum hata ile eşleştirmektir. Alt uzay sınıflandırıcıları, yüksek boyutlu girdi verilerini düşük boyutlu öznitelik alt uzayına dönüştürmektir. Yani bir diğer deyişle alt uzay sınıflandırıcıları, işlem alanını en aza indirgeyen öznitelikleri temsil eder. Binlerce öznitelik yığınınından düşük boyutlu bir öznitelik altuzayı seçmek, optimal sınıflandırma için çok kritiktir.

- **Destek vektör makineleri**

DVM, yapısal risk minimizasyonu prensibine göre çalışan etkin, hızlı ve başarılı bir makine öğrenmesi algoritmasıdır. DVM, örüntü tanıma ve sınıflandırma problemlerinin çözümü için Vapnik tarafından geliştirilmiştir (Cortes ve Vapnik 1995). DVM'nin temelleri istatistiksel öğrenme teorisine diğer bir ifadeyle Vapnik-Chervonenkis (VC) teorisine dayanmaktadır (Vapnik 1999).

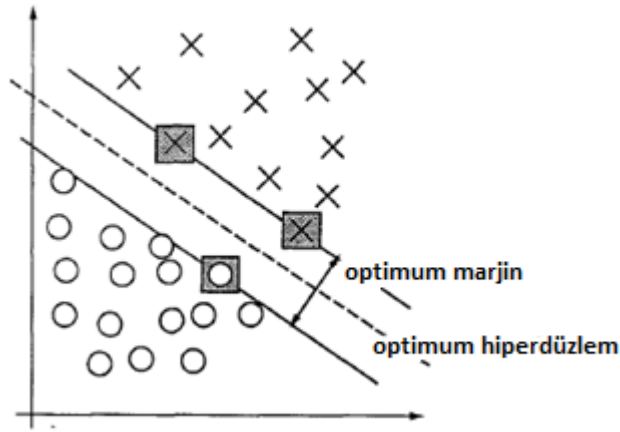
DVM, başlangıçta iki kategorili sınıflandırma için oluşturulan bir tasarımıdır. Daha sonra çok sınıflı problemler ve doğrusal olmayan verilerin sınıflandırılması için geliştirilmiştir.

➤ **Doğrusal olarak ayrılabilen DVM**

DVM mekanizması sınıflandırma gereksinimi için, küçük örnek temelli sınırlı bilgi koşulları altında, iki farklı sınıfı ayırt etmek için en iyi destek vektörü olan bir hiperdüzlem aramaktadır.

İki sınıflı problem şekil 3.10'da gösterildiği gibi iki boyutlu bir düzlem üzerinde gösterilmiştir. Bu özellik alanında lineer bir karar yüzeyi oluşturulmuştur. Karar yüzeyinin özel özellikleri, öğrenen makinenin yüksek genelleme yeteneğini sağlar. Bu sınıflandırma problemindeki amaç; bu iki sınıfı mevcut örneklerden uyarlanmış bir fonksiyon ile ayırmaktır. Şekilde verileri ayırabilecek birçok olası lineer sınıflandırıcı bulunduğu halde amaç, en iyi ayırıcı doğruyu bulabilmektir. Her iki sınıftan, aynı tarafta

aynı sınıftaki noktaların en büyük bölümünü ayıran destek vektörleri (Şekil 3.10’da gri kareler ile işaretlenmiş) arasındaki mesafeye marjın denir. En iyi ayırıcı doğru, bu marjini maksimize eden tek bir lineer doğru (sınıflandırıcıdır). Bu lineer sınıflandırıcı, optimal ayırıcı hiperdüzlem (OSH) olarak adlandırılır. Bu OSH sadece eğitim setindeki örnekleri değil aynı zamanda test setinin görünmeyen örneklerini de sınıflandırma riskini en aza indirir. Sınıflandırma problemi tamamen buna dayanmaktadır.



Şekil 3.10 İki boyutlu bir uzayda ayrılabilir problem örneği (Cortes ve Vapnik 1995)

İki sınıflı bu problemde bu iki sınıfın +1, -1 olduğu varsayalım. Bu veri setinde,

$$t_n = \begin{cases} -1, & \text{sınıf} = -1 \text{ ve} \\ +1, & \text{sınıf} = +1 \text{ ve} \end{cases} \quad (3.8)$$

$$\{(p_n, t_n) \mid p_n \in \mathbb{R}^d\}, \quad n = 1, \dots, N \quad (3.9)$$

Yukarıdaki formülde, N veri sayısına, d girdi öznitelik boyutuna, p_n n.ci örnek verinin öznitelik vektörüne, t_n ise hedef etikete karşılık gelmektedir.

DVM’nin amacı, doğrusal bir fonksiyonla bu iki sınıfı ayırabilecek d boyutlu bir optimal bir hiperdüzlem yaratmaktır. Bu hiperdüzlem aşağıdaki eşitsizlikle ifade edilebilir.

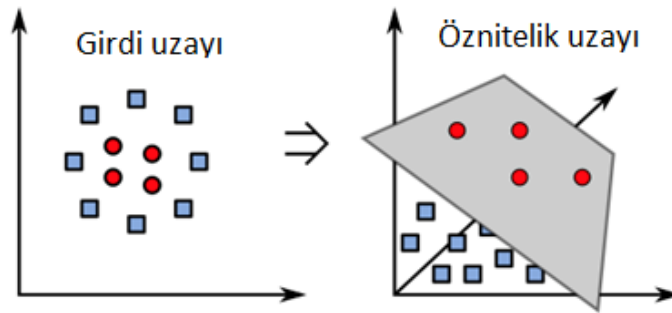
$$w^T p - b = 0 \quad (3.10)$$

w ağırlık değeri, b bias değeridir. Marjini maksimize etmek için $\|w\|$ değerinin minimum olması gerekir. Bu durumda en iyi hiperdüzlemin bulunması için aşağıdaki optimizasyon probleminin çözülmesi gerekir:

$$\min \frac{1}{2} \|w\|^2 \quad (3.11)$$

➤ Doğrusal olarak ayrılamayan DVM

Bu modelde veriler doğrusal olarak ayrılamadığında, DVM iki boyutlu girdi verilerini, orijinal girdi alanından, daha yüksek boyutlu bir öznitelik alanına doğrusal olmayan bir şekilde haritalar. Doğrusal olmayan bu haritalama işlevine çekirdek fonksiyonu denir. Öğrenme algoritması, çekirdek fonksiyonlarından faydalanmak için formülleştirilir, iç çarpımlarının doğrudan öznitelik alanında verimli bir şekilde hesaplanmasına izin verilir. Tipik olarak kullanılan çekirdek fonksiyonları lineer, polinom, RBF (Radial Basis Function) ve sigmoid fonksiyonlarıdır. DVM'nin uygulanmasında çekirdek fonksiyonunun seçimi kritik rol oynamaktadır. Probleme göre bu seçim değişebilmektedir.



Şekil 3.11 Çekirdek fonksiyonları ile verinin daha yüksek bir boyuta dönüştürülmesi (Dima 2016)

Doğrusal olarak ayrılamayan veriler için optimizasyon problemi aşağıdaki gibidir:

$$\min \left[\frac{||w||^2}{2} + C \cdot \sum_{i=1}^r \xi_i \right] \quad (3.12)$$

Burada doğrusal olarak ayrılabilen problemden farklı olarak C ve ξ parametreleri eklenmiştir.

C, hata maliyeti, ξ negatif olmayan slak değeridir.

Veri sınıflandırması için güçlü bir makine öğrenme tekniği olan DVM, verileri daha yüksek belki sonsuz boyutlu bir öznitelik alanına haritalar ve sonra daha yüksek boyutsal uzayda veriyi ayırmak için maksimum marjin ile doğrusal bir ayırma hiperdüzlemi bulur.

Verilen bir dizi etiketlenmiş eğitim örnekleri $\{(x_i, y_i), i = 1, 2, \dots, l\}$, ve $x_i \in \mathbb{R}^n$, $y_i \in \{1, -1\}$ olsun.

Aşağıdaki fonksiyonla yeni bir test örneği x sınıflandırılır.

$$f(x) = \text{sgn}(\sum_{i=1}^l \alpha_i y_i K(x_i, x) + b) \quad (3.13)$$

Burada orijinal girdi uzayı x, daha yüksek boyutlu $f(x)$ öznitelik uzayına (hilbert uzayı) dönüştürülmüştür.

K (çekirdek fonksiyonu) dönüştürülmüş uzayda iki vektörün iç çarpımı biçimindedir. Bu çekirdek fonksiyonu probleme göre seçilebilmektedir.

$$K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle \quad (3.14)$$

α_i ayırıcı hiperdüzlemi belirten dual optimizasyon probleminin Laplace çarpanı, b hiperdüzlemin eşik değeri parametresidir. Φ , x girdi vektörünün çok boyutlu uzaydaki

öznitelik vektörüdür. Bu eğitim örneği x_i , $\alpha_i > 0$ olunca destek vektörleri çağrılır ve DVM destek vektörleri ve hiperdüzlem arasındaki mesafeyi maksimize eden hiperdüzlemi bulur.

➤ Çok sınıflı DVM

Buraya kadar anlatılan kısım DVM'lerinin iki sınıflı problemlere uygulanmasıydı. Çok sınıflı problemler için geliştirilen yöntemler de vardır. Bunlar için de en yaygın olanları;

✓ Lagrange çarpma tekniği

Lagrange fonksiyonu çok sınıflı probleme uyarlanarak çözülebilir. Bu optimizasyon problemi eşitlik 3.15'deki gibidir.

$$\min_{w, \rho, b} \max_{\varphi, \gamma} \left\{ \frac{1}{2} \|w\|^2 + L e^T \rho \sum_{n=1}^N \varphi_n [t_n (w^T p_n - b) + \rho_n - 1] - \sum_{n=1}^N \rho_n \gamma_n \right\} \quad (3.15)$$

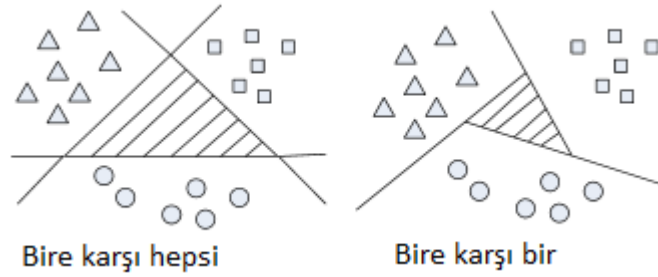
$\rho = (\rho_1, \dots, \rho_n, \dots, \rho_N)$ pozitif slak vektörleri, e N boyutlu vektörlerden biri, L hata cezasını (error penalty) temsil etmektedir.

Fakat sınıf sayısının artması ile L miktarı artacağı için çok başvurulan bir yöntem değildir.

✓ İkili sınıflar halinde sınıflandırma

Bu yöntemde çok sınıflı problem ikili sınıflara bölünerek sınıflandırılır, sonra birleştirilerek çok sınıflı bir sınıflandırma oluşturulur. Bunlar içinde en yaygın olan yaklaşımlar; bire karşı bir, bire karşı hepsi, ikili ağaç yaklaşımlarıdır (Duan ve Keerthi 2005, Nurhanna ve Othman 2017).

Bire karşı hepsi, bir veri kümesi bir sınıf, geriye kalanların tamamı bir sınıf olarak ayarlanır. Yani m sınıf varsa o zaman DVM'nin m adet sınıflandırıcısı üretilecektir. Bire karşı birde ise, DVM iki farklı sınıf seçer. Her bir çift kendi arasında sınıflandırılır. Yani m , $m \times (m - 1) / 2$ alt sınıflandırıcıları üretir.

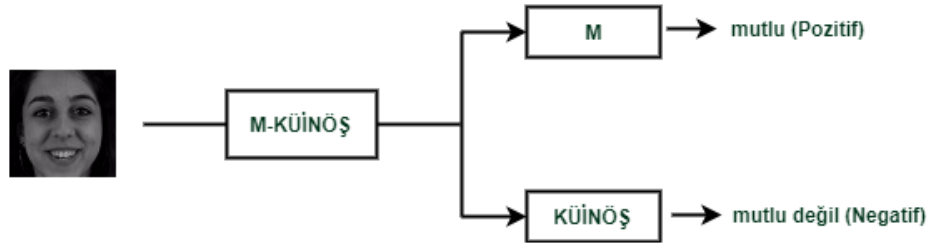


Şekil 3.12 Bire karşı hepsi ve bire karşı bir yaklaşımlarının sınıflandırılmayan bölgeleri (Liu vd. 2010)

Yukarıda bahsedildiği gibi DVM iki sınıflı problemler için geliştirilmiştir. Bu tez çalışmasında YMA problemi yedi sınıflı bir problemdir. Bu yüzden, yedi sınıflı bu görevi iki sınıflı görevlere bölecek bir tekniğe ihtiyaç duyulmaktadır. DVM, ikili kararlar üretir, bu nedenle buradaki çok sınıflı sınıflandırma, bir mimiği diğerlerinden ayırt etmek için ikili sınıflandırıcıları eğiten ve ikili sınıflandırmanın en büyük çıktısına sahip sınıfı çıkartan, bire karşı hepsi tekniği kullanarak gerçekleştirilecektir.

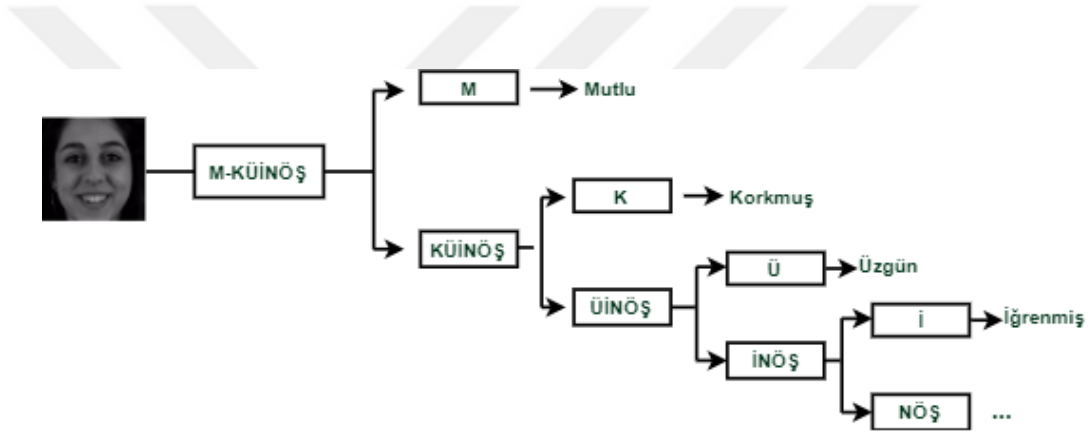
Bire karşı hepsi yöntemi

Bu çoklu sınıflandırma her sınıf için bir tane olmak üzere birçok ikili sınıflandırıcı birleşerek çoklu sınıflandırmayı oluşturmaktadır. Her bir ikili sınıflandırıcı bir sınıftan pozitif örnekler diğer sınıftan negatif örnekler alarak eğitilir. Bu çok sınıflı çıktı en çok verimi veren ikili sınıflandırıcının sınıfı için etkinleştirilir. Örneğin üzgün etiketi pozitif bir sınıf kalan diğer altı mimik üzgün değil diye negatif etiketlenmiştir. Her yedi mimik için ayrı ayrı bu işlem gerçekleştirilir. En yüksek sonucu veren mimik etiketi için çok sınıflı çıktı sonuç olmuştur.



Şekil 3.13 İki sınıflı sınıflandırıcı akış diyagramı

Şekil 3.13'teki ikili sınıflandırıcılar birleştirilerek aşağıdaki çoklu sınıflandırıcı oluşturulur. Sonuçta elde edilen çoklu sınıflandırıcı DVM, yedi adet ikili sınıflandırıcının birleşimi olur.



Şekil 3.14 Çok sınıflı sınıflandırıcı akış diyagramı

Bu problemin matematiksel olarak çözümü aşağıdaki gibidir:

YMA problemi yedisınıflı bir problemdir,

$$M = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7]$$

N adet eğitim örneğine sahip olalım: $\{x_1, y_1\}, \dots, \{x_N, y_N\}$

Burada $x_i \in \mathbb{R}^m$ m boyutlu bir öznitelik vektörü ve $y_i \in \{1, 2, \dots, M\}$ sınıf etiketine karşılık gelir.

Bu yaklaşım 7 adet ikili DVM sınıflandırıcılarından oluşur, her biri bir sınıf geriye kalan bir sınıf olarak ayrılır. i.ci DVM pozitif etiketle i.ci sınıfın tüm eğitim örnekleriyle eğitilir ve kalan tümü negatif etiketlenir. Matematik olarak i.ci DVM i.ci karar fonksiyonu ile aşağıdaki problemi çözer.

$$f_i(x) = w_i^T \Phi(x) + b_i \quad (3.16)$$

$$L(w_i, \xi_j^i) = \frac{1}{2} ||w_i||^2 + C \sum_{j=1}^N \xi_j^i \quad (3.17)$$

$$i^* = \underset{i=1,\dots,M}{\operatorname{argmax}} f_i(x) = \underset{i=1,\dots,M}{\operatorname{argmax}} (w_i^T \Phi(x) + b_i) \quad (3.18)$$

Bu sınıflandırma ifadesinde, f_i 'nin sağladığı en büyük değer i^* için x sınıflandırılmıştır. Sınıflandırıcının sonucu, bu argmax fonksiyonun çıktısı i^* olur ve yeni yüz mimiği görüntüsü sınıf i^* olarak tanımlanır.

3.4 Derin Öğrenme ile Yüz Mimikleri Durum Analizi

Derin öğrenme, derin mimariye dayanan, gizli katmanların sayısı artırılmış ve her katmanda probleme dair bir özneliliğin öğrenildiği yapay sinir ağlarından oluşan yöntemler bütünüdür. Derin öğrenme ile yüz mimikleri tanıma sistemi için derin öğrenmenin özelleşmiş bir yapısı olan KSA kullanılmıştır. KSA ile hem öznelilikler çıkarılmış hem de bu öznelilikler yedi adet mimik durumundan birine sınıflandırılmıştır.

3.4.1 Konvolüsyonel sinir ağları

KSA yapay sinir ağları gibi girdi, çıktı ve gizli katmanları olan derin bir ağ yapısıdır. KSA mimarisinin her katmanında probleme ait bir öznelilik öğrenilir ve bu öğrenilen nitelik sıradaki katmana bir girdi oluşturur. Böylelikle en baştaki katmandan en sondaki katmana doğru basit niteliklerden karmaşık niteliklerin öğrenildiği bir yapı kurulmuş

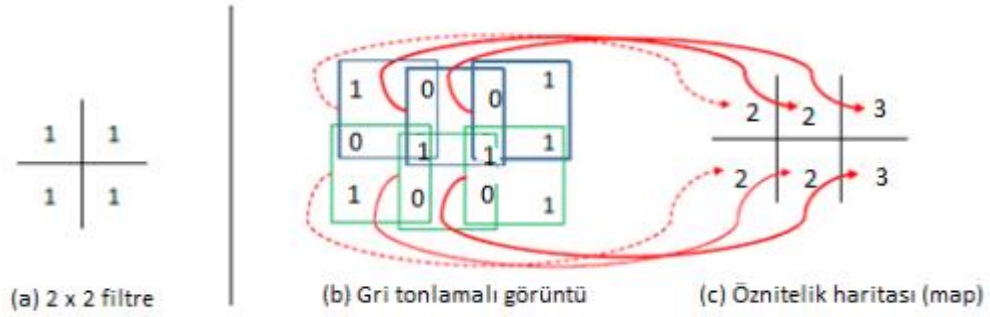
olur. Ayrıca bu özniteliklerin sayısı her bir katmanda adım adım azalır. Sonuç olarak yüksek seviyeli öznitelik katmanında en önemli özniteliklerin karışımı elde edilmiş olur. Son katmanda ise bu yüksek seviyeli özniteliklersınıflandırılır.

Bir işlem modülünde segmentasyon, öznitelik çıkarımı ve sınıflandırmayı birleştirmesi bakımından KSA yapısı eşsiz bir yöntemdir. Konvolüsyon katmanı gürültüyü kaldırır ve görüntünün çizgilerini, kenarlarını, köşelerini tespit eder. Alt örnekleme katmanında görüntü bozulmalarını önlemek için görüntünün çözünürlüğü azaltılır. Ayrıca diğer sinir ağlarından farklı olarak karmaşık veri tabanı uygulandığında sistem minimum düzeyde yeniden tasarım süreci geçirmelidir.

- **Konvolüsyon katmanı**

Bu katman KSA'nın temel yapı taşı aynı zamanda ilk katmandır. İsmi görüntü işlemede sık sık kullanılan 'konvolüsyon' işleminden almıştır. Öznitelik çıkarma işlemleri bu katman ile gerçekleştirilir. Girdi olarak ele alınacak görüntü piksel değerlerinden oluşan bir matristir. Bu katmanın parametreleri öğrenilebilir filtrelerden oluşur. Bu filtreler bu matrise uygulanır ve öznitelikler elde edilir.

Konvolüsyon: Bir matrise uygulanan kayan bir pencere fonksiyonudur. Bu kayan pencereye filtre veya çekirdek (kernel) denilir, bu filtre öznitelik algılayıcısı (feature detector) olarak işlev gerçekleştirmektedir. Şekil 3.15'te, (a) 2x2 boyutlu bir filtreyi, (b) görüntünün 3x4 boyutlu matrisini göstermektedir. 0 ve 1 ile gösterilenler sırasıyla siyah ve beyaz piksellere karşılık gelmektedir. Bu filtre matrisin başından başlayarak tüm resim gezilir ve filtre katsayıları ile matrisin üzerine geldiği kısmın değerleri eleman elemana çarpılıp toplamı alınır. Bu sonuçlar, öznitelik ya da aktivasyon haritası denilen yeni matriste filtrenin merkezinin geldiği konuma yazılır. Bu işleme konvolüsyon işlemi denmektedir. (c) Konvolüsyon işleminden sonra elde edilen öznitelik haritasını göstermektedir.



Şekil 3.15 Konvolüsyon işlemi ile elde edilen öz nitelikler

Geliştirilen bir KSA uygulamasında, KSA eğitim süresince bu filtrelerin değerlerini kendi başına öğrenir. Yapılması gereken eğitimden önce filtre sayısı, filtre büyüklüğü, ağın mimarisi gibi belirli parametreleri belirlemektir. Bu parametreler öz nitelik haritasının boyutunu kontrol etmektedir. Bu parametrelere konvolüsyon adımından önce karar verilmelidir.

Bir konvolüsyon katmanının parametreleri:

Filtre sayısı: Her bir konvolüsyon katmanında kullanılacak filtre sayısı değişmektedir. Filtre sayısı arttıkça, daha fazla öz nitelik çıkarılır ve görüntü detaylandırılır.

Derinlik: Öz nitelik haritasının derinliği, konvolüsyon işlemi için kullanılan filtrelerin sayısına karşılık gelir. Bu öz nitelik haritaları 2B matris olarak düşünülebilir.

Uzaysal boyut: Filtreler en ve boy olmak üzere 2B bir matristir. Uzaysal boyut denilen parametre filtre büyüklüğüdür.

Adım büyüklüğü: Filtrenin bir kez kaydırılması sırasındaki adım sayısıdır (stride). Adım 1 olduğunda, filtreler bir seferde bir piksel taşınır, adım 2 olduğunda bir seferde 2 piksel kaydırılır. Adım boyutu ne kadar artarsa, filtre daha az uygulanır ve daha küçük öz nitelik haritaları elde edilir.

Zero padding: Bir görüntüye filtre uygulandığında matrisin üstte ve solda komşu öğelere sahip olmayan elemanları bu filtreden yararlanamayacaktır. Bu durumda zero padding kullanılır. Matrisin dışına düşen tüm elemanlar sıfır alınır. Böylece, filtre girdi matrisinin her bir elemanına uygulanabilir. Zero padding değerine göre öznitelik haritasının büyüklüğü değişecektir. Zero padding değeri büyümesi daha büyük bir öznitelik haritası ile sonuçlanacaktır. Aksi durumda ise tam tersidir.

Konvolüsyon ağında çıktı olarak elde edilen öznitelik haritalarının boyutunun matematiksel olarak hesaplanması aşağıdaki gibidir:

Girdinin eni $\rightarrow E_g$, Girdinin boyu $\rightarrow B_g$, Filtre sayısı $\rightarrow k$, Uzaysal boyutu $\rightarrow F$, Adım büyüklüğü (stride) $\rightarrow s$, Zero padding $\rightarrow p$, Çıktının eni $\rightarrow E_\zeta$, Çıktının boyu $\rightarrow B_\zeta$, Derinliği $\rightarrow D$ ile ifade edilsin.

$$E_\zeta = (E_g - F + 2p)/s + 1 \quad (3.19)$$

$$B_\zeta = (B_g - F + 2p)/s + 1 \quad (3.20)$$

$$D = k \quad (3.21)$$

$$\text{Çıktı Büyüklüğü} = E_\zeta \times B_\zeta \times D \quad (3.22)$$

- **Parametre paylaşımı**

Eğer konvolüsyon işlemi söz konusu olmasaydı tek bir katman için parametre sayısı eşitlik 3.23'deki gibi olacaktı.

$$\text{parametre sayısı} = \text{nöron sayısı} \times \text{ağırlıklar(+bias)} \quad (3.23)$$

Bu şekilde oldukça fazla sayıda parametre ortaya çıkacaktı. Konvolüsyon işleminde kullanılan filtrelerin yerel bağlantısı sayesinde binlerce rakamı bulabilecek olan parametre sayısı azaltılmaktadır. Ağırlıkları ve biasları kullanmak için, her derinlik diliminde (filtre sayısı kadar) nöronlar sınırlanacaktır. Yani parametre paylaşımı denilen bu yöntemle her derinlik dilimindeki nöronların tümü aynı parametreleri kullanıyor olacaktır. Geri besleme sırasında, hacimdeki her nöron gradyanı ağırlıklar için hesaplar, ancak bu gradyanlar her derinlik dilimine eklenir ve yalnızca dilim başına tek bir ağırlık kümesi güncellenir. Eğer tek bir derinlik dilimindeki tüm nöronlar aynı ağırlık vektörünü kullanıyorsa, konvolüsyonel katmanının her derinlik diliminde ileri geçişi girdi ile nöron ağırlıklarının bir konvolüsyonu olarak hesaplanır. Bu sebeple, girdi ile konvolüsyona tabi tutulan ağırlık kümesine bir filtre olarak bakmak yaygın bir durumdur.

Burada önemli bir husus konum değiştirme (translation invariance) ortaya çıkmaktadır. Örneğin; yatay bir kenarın algılanması görüntünün bazı yerlerinde önemli ise, başka yerler de sinir ağı, tekrar bu yatay kenardır diyip atlayamaz, her seferinde yeniden öğrenmesi gerekir. Fakat katmanın çıktı hacmindeki $E_c \times B_c \times D$ farklı konumunun her birinde tekrar yatay bir kenarı algılamamanın gereği yoktur. Konvolüsyon işlemi sayesinde yatay kenar resmin neresinde olursa olsun bir kere de anlaşılacak resim üzerinde tekrar tekrar öğrenmesine gerek kalmayacaktır.

Konvolüsyon işleminin matematiksel ifadesi aşağıda tanımlandığı gibidir.

$$y^j = \max \left(0, \sum_i x^i * k^{i,j} + b^j \right) \quad (3.24)$$

$x^i \rightarrow$ i.ci input map'i, $y^j \rightarrow$ j.ci output map'i, $*$ $\rightarrow$ konvolüsyonu,

$k^{i,j} \rightarrow$ i.ci input map x^i , j.ci output map y^j bağlantılı filtre konvolüsyonu,

$b^j \rightarrow$ j.ci output map'in biasını ifade eder.

Farklı yerel öznitelikleri öğrenmek için, derin KSA mimarisinin her bir katmanındaki ağırlıklaryerel olarak paylaşılmıştır.

- **Rectified Linear Unit (ReLU)**

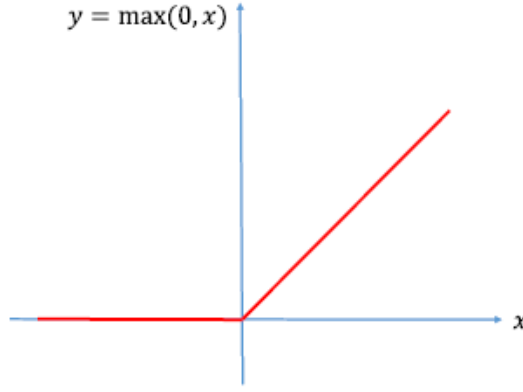
KSA yapısında katmanlar sigmoid, tahn, ReLU vb. gibi aktivasyon fonksiyonlarını kullanmaktadırlar. Bu fonksiyonları kullanmaktaki amaç, KSA'nın doğrusal olmayan fonksiyonları öğrenmesidir çünkü gerçek dünyadaki verilerin çoğu doğrusal değildir. Konvolüsyon işlemi de lineer bir işlemdir. Doğrusal olmaması için, bu çalışmada doğrusal olmayan ReLU fonksiyonu benimsenmiştir. Sigmoid'in yerine geçmek için ReLU'nun uygulanması, KSA mimarisinde önemli bir değişikliktir ve bu da KSA parametrelerini öğrenme zorluğunu önemli ölçüde azaltır ve doğruluğunu artırır (Zhang ve Woodland 2015).

Ayrıca gradyanın katmanlar boyunca düşmesi sebebiyle, ağı daha alt katmanlarında çok yavaş eğitim sorunu olan; Kayıp Gradyan Problemi (Vanishing Gradient Problem) nin hafifletilmesine yardımcı olur.

ReLU, sıfırda eşikleme $f(x) = \max(0, x)$ gibibir fonksiyonu kullanır. Bu fonksiyon ile öznitelik haritasındaki tüm negatif piksel değerleri sıfır ile değiştirilir. Bir diğer değişle ReLU fonksiyonu, katmandaki önemsenmeyen özniteliklerin eğimlerini 0 olarak ayarlar; aktif olan öznitelikler için ise, biaslar herhangi bir değişiklik yapılmaksızın geri yayılır ve Stokastik Gradyan Düşüşü (SGD) öğrenmesi için yararlıdır.

Bu fonksiyon girdinin boyutunda herhangi bir değişiklik yapmaz. Girdi ve çıktı aynı boyutu paylaşır. Ayrıca parametre ve hiperparametre içermez. Fonksiyonun çıktısı aşağıdaki formülde de ifade edildiği gibidir.

$$y_{i,j,d} = \max\{0, x_{i,j,d}^l\} \quad (3.25)$$

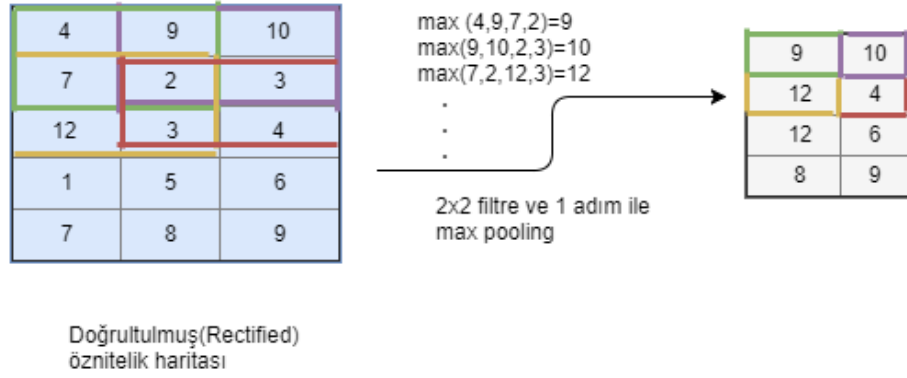


Şekil 3.16 ReLu fonksiyonu

- **Havuzlama/alt örnekleme katmanı**

Bir KSA mimarisinde art arda konvolüsyon katmanları arasına bir havuzlama katmanı eklemek yaygındır. Bu katmanın işlevi, ağıdaki parametre ve hesaplama miktarını azaltmak dolayısıyla aşırı uyumu kontrol etmektir.

Havuzlama katmanında her öznitelik haritasına duruma göre max, avg veya sum operatörleri uygulanabilir. En yaygın formu, max pooling operatörüdür. Max işlemiyle girdinin her derinlik diliminde bağımsız olarak çalışması ve böylece hacmin uzaysal olarak alt örneklenmesi yapılmaktadır. Bu yüzden uzamsal boyut derece derece azaltılarak yeniden boyutlandırılmış olur. Fakat derinlik boyutu değişmeden kalır.



Şekil 3.17 Havuzlama işlemi

Şekilde görüldüğü gibi her bölgeye 2x2 boyutlu filtre ile bir adım max pooling uygulanarak, filtrelerde oluşan değerlerin en yüksekisi alınmıştır. Böylece en önemli öznitelikler seçilerek doğrultulmuş öznitelik haritasının boyutu küçültülmüş olur.

max pooling matematiksel olarak aşağıdaki gibi ifade edilir.

$$y_{a,b}^j = \max_{0 \leq m, n \leq s} x_{a.s+n, b.s+m}^j \quad (3.26)$$

j .ci çıktı haritasında her bir değer için y^j , j .ci girdi haritası, x^j de $s \times s$ çakışmayan alanından fazlasını alt örnekler (bir merkezde toplar).

Max pooling katmanlarda önceki katmandaki bir bölge mevcut katmandaki bir birime bağlanarak, öznitelik haritalarının boyutsallığını azaltır. Ayrıca havuzlama; çevirme, döndürme ve ölçeklendirme gibi durumlar olduğunda ölçekte değişmezliği artırır. Bir bölgenin üzerinde havuz oluştururken görüntü birkaç piksel kaydırılıp döndürülse bile çıktı aynı kalır, çünkü max operatörler yerel bölgede hep aynı değeri yani en büyük değeri korur ve sıradaki katmana geçirir.

Ağın daha derin kısmında, boyutları küçülse de filtreler daha büyük alıcı alana sahip olmaya başlar; bu da daha büyük bir piksel alanına daha duyarlı olmalarıdır.

Havuzlama katmanları için zero padding kullanmak yaygın değildir.

- **Tam bağı katman**

Konvolüsyonel ve havuzlama katmanlarından sonra gelen katmanlardır. Bu iki katmandan farkı; önceki katmandaki her nöron bir sonraki katmandaki her bir nörona tam bağlıdır. Tam bağı katman (FC), çıktı katmanında softmax aktivasyon fonksiyonunu kullanan geleneksel çok katmanlı perseptron olarak düşünülebilir. Amacı, diğer katmanlardan gelen öznitelikleri eğitim veri kümesine dayalı çeşitli sınıflara sınıflandırmak için kullanmaktır. DVM gibi diğer sınıflandırıcılar da kullanılabilir.

Bu, tamamen genel amaçlı bir bağlantı modelidir ve verilerdeki öznitelikler hakkında hiçbir varsayım yapmaz. Bellek (ağırlık) ve hesaplama (bağlantılar) açısından da çok pahalıdır.

Konvolüsyon ve havuz katmanlarının çıktısı, girdi resminin üst düzey özniteliklerini temsil eden bir öznitelik alanıdır. Bu öznitelik alanı çıktısı, çıktı katmanına direkt bağlanabilirken, FC katmanın eklenmesi, o alanda bu özniteliklerin doğrusal olmayan kombinasyonlarını öğrenmeyi sağlayacaktır. Yani konvolüsyonel ve havuz katmanlarından çıkarılan özniteliklerin çoğu sınıflandırma görevi için iyi olabilir, ancak bu özniteliklerin kombinasyonları daha iyi olabilir (Karn 2016).

Bu katmandaki tüm nöronlar, normal sinir ağlarında olduğu gibi, önceki katmandaki her nörona tüm aktivasyonlarla bağlanır ve her bağlantı kendi ağırlığına sahiptir. Dolayısıyla, aktivasyonları bir bias ofset tarafından izlenen bir matris çarpımı ile hesaplanabilir.

$$y^j = \sum_i x^i \times k^{i,j} + b^j \quad (3.27)$$

$x^i \rightarrow$ FC katmanının i.ci çıktı değerine, $y^j \rightarrow$ yüksek seviyeli özniteliklerde j.ci özniteliğe, $\times \rightarrow$ normal çarpıma, $k^{i,j} \rightarrow$ i.ci çıktı değeri ve j.ci öznitelik arasındaki ağırlığa, $b^j \rightarrow$ j.ci öznitelik bias değerine karşılık gelmektedir.

- **Softmax çıktı katmanı**

Son olarak, yüz mimikleri üzerinde eğitilmiş bir softmax katmanı, yedi mimiğin her biri için ayrı bir olasılık üretecek ve olasılıkların toplamı 1 olacaktır.

Lojistik regresyon iki sınıf arasındaki sınıflandırma için kullanılmaktadır. Bu lojistik fonksiyon, çok sınıflı kategorik olasılık dağılımı çıkarmak üzere, softmax çıktı katmanı ile genelleştirilebilir. Bu softmax fonksiyonu ζ , girdi olarak C boyutlu vektör z 'yi alır ve 0 ve 1 arasında gerçek değerlerinde C boyutlu vektör y 'yi çıktı verir. Çıktı olasılıklarının toplamı 1 olacaktır. Bu fonksiyon normalize edilmiş bir üstel olup aşağıdaki şekilde tanımlanmıştır:

$$c = 1 \dots C \text{ için} \quad y_c = \zeta(z)_c = \frac{e^{z_c}}{\sum_{d=1}^C e^{z_d}} \quad (3.28)$$

Sinir ağının çıktı katmanı olarak; softmax fonksiyonu, C nöronlu bir katman olarak grafiksel olarak gösterilebilir. Girdi (z), $c = 1 \dots C$ için $t = c$ sınıfları için olasılıklar eşitlik 3.29'daki gibi yazılır. Burada $P(t = C|z)$ z girdisi için c sınıfının olasılığıdır.

$$\begin{bmatrix} P(t = 1|z) \\ P(t = C|z) \end{bmatrix} = \begin{bmatrix} \zeta(z)_1 \\ \zeta(z)_C \end{bmatrix} = \frac{1}{\sum_{d=1}^C e^{z_d}} \begin{bmatrix} e^{z_1} \\ e^{z_C} \end{bmatrix} \quad (3.29)$$

Sinir ağlarındaki softmax fonksiyonunu kullanmak için, bu fonksiyonun girdi ve çıktısının türevine ihtiyaç vardır. Eğer $c = 1 \dots C$ ve $\sum_C = \sum_{d=1}^C e^{z_d}$ için y_c hesaplanmışsa, softmax fonksiyonun girdisi z ile çıktısı y 'nin türevi; $\frac{\partial y_i}{\partial z_j}$ hesaplanır.

Bu olasılıkların maksimizasyonu şu şekilde yazılabilir:

$$\underset{\theta}{\operatorname{argmax}} L(\theta|t, z) \quad (3.30)$$

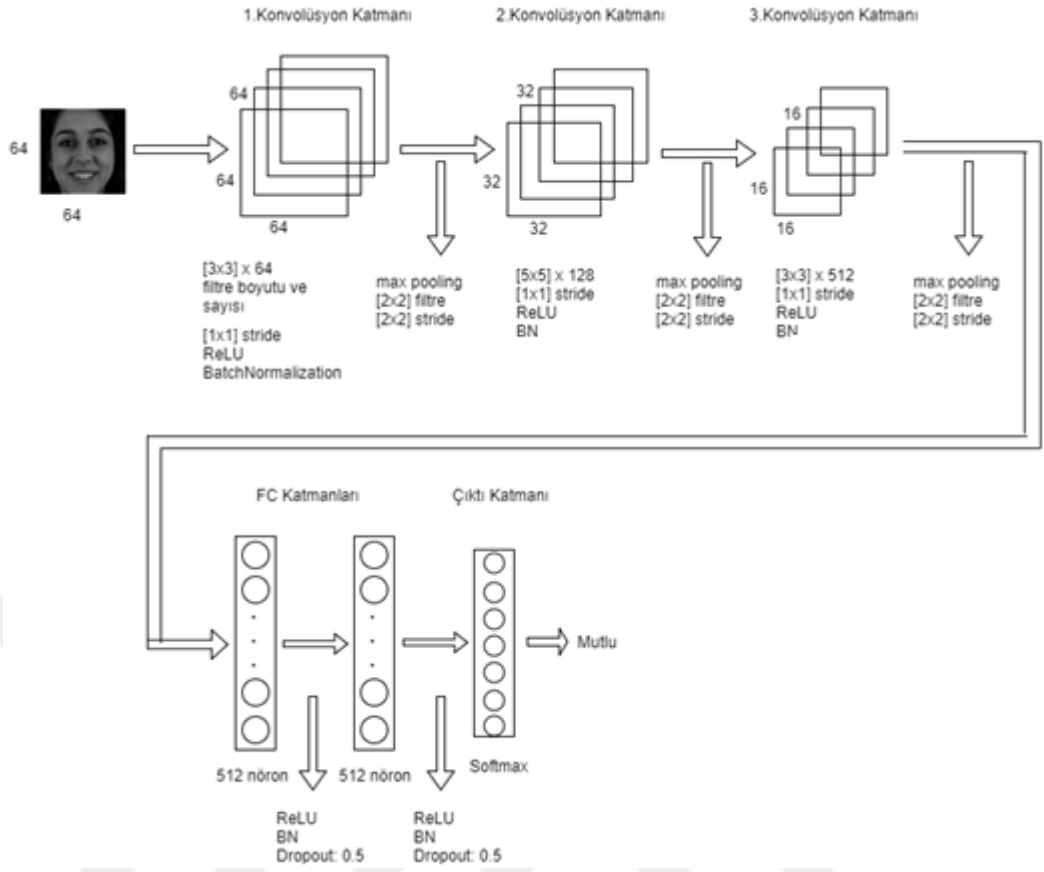
θ , her bir girdi (z) için doğru sınıfın (t) değerini tahmin edebilen modelin parametre kümesidir.

3.4.2 Geliştirilen KSA mimarisi

Bu tez kapsamında bir KSA mimarisi geliştirilmiş ve bu mimarinin eğitimi gerçekleştirilmiştir.

Tez kapsamında gerçekleştirilen YMA çalışmasında çizelge 3.1’de yer alan Jung vd. (2014), Lopes vd. (2015), Khorrami vd. (2015), AlexNet mimarileri uygulanıp test edilmiştir. Ancak en iyi başarı oranı şekil 3.18’de geliştirilen mimariden alınmıştır.

Bu KSA mimarisi DeepLearning4J kütüphanesi ile geliştirilmiştir. Bu kütüphane yüksek seviyeli bir derin öğrenme modeli oluşturulmasına izin vermiştir. Ayrıca bu kütüphane diğer birçok çerçeveden (framework) farklı olarak, optimizasyon algoritmasını güncelleyici algoritmadan ayırmaya karar vermiştir. Bu da veriler ve problem için en iyi sonucu veren bir kombinasyon bulmaya çalışırken esnek olunmasını sağlamaktadır.



Şekil 3.18 Geliştirilen KSA mimarisi

Bu mimaride öncelikle 64x64 yeniden boyutlandırılmış gri tonlamalı görüntü girdi olarak alınır. Yapılandırılan KSA mimarisinin ilk katmanı konvolüsyonel katmandır. Bu katmanda 64 adet 3x3 boyutunda filtreler kullanılarak bir adımla (stride=1) konvolüsyon işlemleri yapılır. Burada en yaygın kullanılan max pooling operatörü ile 2x2 filtrelerle iki adımla $\frac{1}{2}$ alt örnekleme faktörüne tabi tutulur. Böylece elde edilen yeni görüntünün boyutu 32x32 olur. Bu katmanın çıktısı, sıradaki konvolüsyon katmanının girdisi artık yeni boyutlu 32x32 görüntü olmuştur. Bu katmanda da 128 adet 5x5 filtreler kullanılarak konvolüsyon işlemi yapılır. Sıradaki havuzlama katmanda max pooling operatörü ile 2x2 filtrelerle iki adımla alt örnekleme işlemleri yapılır. Elde edilen 16x16 görüntülere 512 adet 3x3 filtrelerle konvolüsyon yapılır. 3.havuzlama katmanındaki işlemlerden sonra 8x8 boyutlu bir çıktı elde edilir. Artık konvolüsyon ve katmanlarının işlemleri bitmiş, öznitelikler elde edilmiştir. 5x5 boyutunda çıktı yani öznitelik haritası tam bağlı katmanın girdisi olmuştur. Bu sefer diğerlerinden farklı olarak bu girdinin tüm nöronları FC katmanının 512 adet nöronuna tam bağlıdır.

Sıradaki katman yine 512 nöronlu FC katmanıdır. Bu FC katmanı en son yedi mimik sınıfına sınıflandıracak çıktı katmanına tam bağlanmıştır. Bu çıktı katmanı softmax aktivasyon fonksiyonunu kullanmaktadır.

Geliştirilen KSA mimarisinde her konvolüsyon işleminden ve FC katmanlarından sonra ReLU aktivasyon işlemi gerçekleştirilmiştir.

Çizelge 3.1 Test edilen KSA mimarileri

Katman		Lopes vd.(2015)	Jung	Khorrami	AlexNet
1. katman	Katman	Konv	Konv	Konv	Konv
	Filtre	5x5 (1,0)	5x5 (1,0)	5x5 (1,0)	11x11(4,0)
	Harita	28@32	48@32	96@64	227@96
2. katman	Katman	Maxp	Maxp	Maxp	Maxp
	Filtre	2x2 (1,0)	3x3 (2,0)	2x2 (1,0)	3x3 (2,0)
	Harita	14@32	24@32	48@64	
3. katman	Katman	Konv	Konv	Konv	Konv
	Filtre	7x7 (1,0)	5x5 (1,0)	5x5 (1,0)	5x5(1,2)
	Harita	8@64	12@32	48@128	48@256
4. katman	Katman	Maxp	Maxp	Maxp	Maxp
	Filtre	2x2 (1,0)	3x3 (2,0)	2x2 (1,0)	3x3(2,0)
	Harita	4@64	6@32	24@128	
5. katman	Katman	FC	Konv	Konv	Konv
	Filtre	-	5x5 (1,0)	5x5 (1,0)	3x3(1,1)
	Harita	1@256	6@64	24@256	256@384
6. katman	Katman	Cıktı	Maxp	Quadp	Konv
	Filtre	1@6	3x3 (2,0)	2x2 (1,0)	3x3(1,1)
	Harita		3@32	12@256	192@384
7. katman	Katman		FC	FC	Konv
	Filtre				3x3(1,1)
	Harita		1@100	1@300	192@256
8. katman	Katman		FC	Cıktı	Maxp
	Filtre				3x3(2,0)
	Harita		1@100	1@7	
9. katman	Katman		Output		FC
	Filtre				
	Harita		1@7		1@4096
10. katman	Katman				FC
	Filtre				
	Harita				1@4096
11. katman	Katman				Output
	Filtre				
	Harita				1@7
12. katman	Katman				
	Filtre				
	Harita				
13. katman	Katman				
	Filtre				
	Harita				

Model tasarlandıktan sonra hiperparametreler ile modelin eğitilmesi gerekecektir. Hiperparametreler, sinir ağının nasıl öğrenildiğini belirleyen değişkenlerdir. Modelin ağırlıklarının kaç kez güncelleneceğini, bu ağırlıkların nasıl başlatılacağı, hangi aktivasyon fonksiyonların düğümlere ekleneceği, hangi optimizasyon algoritmasının kullanılacağı ve modelin ne kadar hızlı öğreneceğini içerirler.

Bu tez çalışmasında her üç veri seti için parametreler seçilmiştir. Çünkü parametre seçimi oldukça önemlidir. Ağın hiperparametreleri kötü seçilmişse, ağ çok yavaş öğrenecek belki de hiç öğrenemeyecektir.

Çizelge 3.2 Her üç veritabanı için kullanılan hiperparametreler

Parametre	JAFFE Veritabanı	KDEF Veritabanı	MUG Veritabanı
Epok	138	91	23
Düzenleştirme	0.01	0.01	0.01
Yığın büyüklüğü	16	57	128
Öğrenme oranı			
0-500 iterasyon	0.01	0.01	0.01
500-1000 iterasyon	0.005	0.005	0.005
1000+	0.003	0.003	0.003
Momentum	0.9	0.9	0.9
Bırakma (dropout)	0.5	0.5	0.5
Ağırlık başlatma	Distribution	Distribution	Distribution
Optimizasyon alg.	SGD	SGD	SGD
Kayıp fonksiyon	Cross entropy loss (NLL)	Cross entropy loss (NLL)	Cross entropy loss (NLL)

Epok ve iterasyon: Tüm veri kümesindeki veriler üzerinden tam bir geçişe ‘epok’ denmektedir. Model uygun uyumu yakalayana kadar, epok sayısı kadar, eğitilir. Az epok, ağıın parametreleri öğrenmesi için yeterli zaman tanımaz; fazla epok ise ağıın aşırı uyumuna neden olabilir. Epok sayısını seçmenin bir yolu erken durdurmayı kullanmaktır. Erken durdurma, sinir ağıının aşırı uyumu önlemesine yardımcı olabilir. Diğer bir deyişle, ağıın görünmeyen verilere daha iyi genelleşmesine yardımcı olabilir.

İterasyon, her bir yığın için bir satırdaki parametre güncellemelerinin sayısı olarak tanımlanır.

Düzenleştirme (Regularization): Düzenleme yöntemleri, eğitim sırasında aşırı uyumu önlemeye yardımcı olabilir. Ağ, eğitim setini çok iyi tahmin ettiği zaman aşırı uyum meydana gelir, ancak ağıın hiç görmediği veriler üzerinde kötü öngörülerde bulunur. Yani aşırı uyum, ağıın eğitim verilerini öğrenmek yerine ezberlemesidir.

Düzenleştirmenin yaygın türleri arasında L1 ve L2 bulunur. Bu düzenleştirmeler büyük ağ ağırlıklarını cezalandırır ve ağırlıkların çok büyük olmasını önler. Uygulamalarda L2 düzenleştirmesi yaygın olarak kullanılır. Bununla birlikte, L1 veya L2 düzenlilik katsayıları çok yüksek olursa, ağ aşırı derecede cezalandırılabilir ve öğrenmesi durdurulabilir. L2 düzenlenmesi için ortak değerler $1e-3$ ile $1e-6$ ’dır.

Dropout normalizasyonu: Eğitim sırasında optimumu bir yerel noktada takılıp kalmış bir aktivasyon grubunu pasifize eder. Böylece eğitimin sürekli aynı ağırlıklarda kalmamasını sağlamış olur. Bu yüzden bu çalışmada ağı düzenleştirmek ve aşırı uyumla mücadele etmek için dropout kullanılmıştır. Ayrıca dropout yalnızca eğitim sırasında kullanılıp, test sırasında kullanılmamaktadır.

Çalışmada her üç veri seti için dropout’un değeri 0.5’tir.

Yığın (Batch) normalizasyonu: Bir yığın, gradyanlar ve parametre güncellemeleri hesaplanırken bir seferde kullanılan örnek sayısına ‘yığın büyüklüğü’ denmektedir.

Yığın ağı verilmiş gibi normalize edilir. Böylece öğrenmeyi hızlandırır, yüksek öğrenme oranına izin verir çünkü öğrenmeyi düzenleştirir.

Uygulamada veri setleri birkaç yığın aralığına bölünmüştür. JAFFE veri kümesi için 187 resim 16 yığın büyüklüğüne bölünmüştür. 16'şar 16'şar yaklaşık 12 adet grup etmektedir. Tek bir grup içindeki verilerin güncellenme sayısına iterasyon, tüm veriler üzerinden bir kez geçmeye epok dendiğini hatırlayınız. Çalışmada her bir yığındaki verilere 1 iterasyon uygulanmıştır. İlk gruptan başlanarak sırasıyla tüm 12 grup ayrı ayrı eğitilmektedir.

KDEF veri kümesi 490 resim 57 yığın büyüklüğüne bölünmüştür. Bu da yaklaşık 9 adet grup etmektedir. Burada her bir yığındaki verilere 1 iterasyon uygulanmıştır.

MUG veri kümesi 9649 resim 128 yığın büyüklüğüne bölünmüştür. 128'er 128'er yaklaşık 75 adet grup etmektedir. Burada her bir yığındaki verilere 1 iterasyon uygulanmıştır.

Veri setindeki veriler arttıkça orantılı olmasa da yığın büyüklüğü de büyümektedir. Örneğin bir veri kümesinde 100 adet diğerinde 1000 adet veri olsun. İkisinde de yığın büyüklüğü 10 olursa birinde 10 adet, diğerinde 100 adet grup olacaktır. Bu da diğer parametrelerin daha orantısız şekilde dağılmasına sebep olacaktır. Bu yüzden veri artışına bağlı olarak daha makul yığın büyüklükleri seçilmelidir.

Öğrenme Hızı: Öğrenme hızı, en önemli hiperparametrelerden biridir. Ağırlıkların değişim miktarlarıdır. Eğer büyük seçilirse yerel çözümler arasında takılır kalır. Küçük değerler seçilirse öğrenme zamanı artar. Öğrenme hızı için tipik değerler 0.1 ile $1e-6$ arasındadır, ancak optimal öğrenme oranı genellikle veri ve ağ mimarisi için spesifikdir. Uygulamada veri seti için 0-500 arası iterasyon için öğrenme oranı 0.01, 500-1000 arasında iterasyon için 0.005, 1000'den fazla iterasyonda 0.003 olarak ayarlanmıştır.

Ağırlık başlatma (Weight initialization): Ağ eğitime başlanmadan önce ilk olarak parametrelere ilk değer ataması yapmak için bir işlem tanımlanır. DeepLearning4j, weightInit parametresiyle birkaç farklı ağırlık başlatmayı desteklemektedir. Xavier, zero, uniform, ReLu, Distribution, sigmoid_uniform gibi çeşitleri vardır. (<https://deeplearning4j.org> 2017). Bu çalışmada distribution çeşidi tercih edilmiştir. Distribution çeşidinde, başlangıçta parametrelerin ve ağırlıkların değerleri normal (gauss) dağılımına göre atanır.

Güncelleyici ve optimizasyon algoritması: Güncelleyici terimi, momentum, adagrad, rmsprop gibi eğitim mekanizmalarını ifade eder. Bu mekanizmalardan birini kullanmak, çok daha hızlı ağ eğitimi sağlayabilmektedir.

Optimizasyon algoritması, gradyana bakıldığında, güncellemelerin nasıl yapıldığıdır. En basit ve en sık kullanılan yöntem, Stokastik Gradyan Düşüş (SGD) yöntemidir; ancak DL4J ayrıca SGD'ye satır araması, eşlenik gradyan ve LBFGS optimizasyon algoritmaları sunar. Bu son algoritmalar, SGD'ye kıyasla daha güçlüdür, ancak bir satır arama bileşeni nedeniyle parametre güncelleme başına oldukça maliyetlidir ve pratikte fazla kullanılmaz. Sonuç olarak herhangi bir güncelleyici herhangi bir optimizasyon algoritması ile birleştirilebilir.

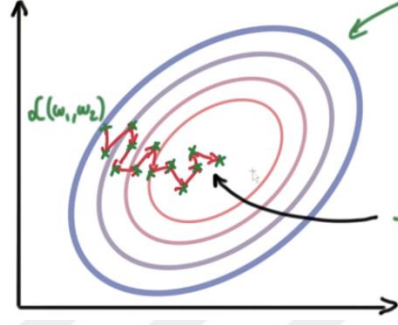
Bu mimaride SGD optimizasyon algoritması, güncelleştiricilerden momentumu kullanmıştır.

Stokastik gradyan düşüşü:

$$\theta_{t+1} = \theta_t - \alpha \delta L(\theta_t)$$

Teta (ağırlık), her teta için kaybın eğimin düşüşüne göre değiştirilir. Alfa öğrenme oranıdır. Alfa çok küçükse, hata minimum yakınsaması yavaş olacaktır. Çok büyükse, model hata minimumundan uzaklaşacak ve öğrenme sona erecektir.

Eğim düşüşü (L), her iterasyon işleminden sonra, eğitim örnekleri arasındaki varyansa bağlı olarak hızla değişir. Şekil 3.19'daki yakınsama yolunda güncelleyici küçük adımlar atmakta, ancak bu adımlar hata minimumuna doğru giderken öne ve arkaya zikzaklar yapmaktadır.



Şekil 3.19 Stokastik gradyan düşüşü

Momentum: Zig zag işlemini durdurmak için momentum kullanılmaktadır. Yani ağıın öğrenme esnasında yerel bir optimum noktaya takılıp kalmaması için ağırlık değişim değerinin belirli bir oranda bir sonraki değişime eklenmesini sağlar. Momentum, bilgiyi önceki adımlardan güncelleyicinin nereye gitmesi gerektiği yere uygular. Bunu temsil etmek için μ veya "mu" şeklinde yeni bir hiperparametre kullanılır.

$$v_{t+1} = \mu v_t - \alpha \delta L(\theta_t) \quad (3.28)$$

$$\theta_{t+1} = \theta_t + v_{t+1} \quad (3.29)$$

Kayıp (loss) fonksiyonu: Veri setleri üzerinde sınıflandırma fonksiyonunun ne kadar iyi ya da kötü olduğunu ölçen hiperparametredir. Mevcut yığın üzerindeki iterasyonlar arttıkça kayıp fonksiyonu düşmüştür.

Sinir ağında ağırlıkları daha iyi öğrenmek için ya ön eğitimde, ya da sınıflandırma için çıktı katmanında kayıp fonksiyonları kullanılabilirler.

Ağın amacı, kullanılan kayıp fonksiyonunu belirleyecektir yani uygulamaya özeldir. Ön eğitim için, yeniden yapılandırma entropisi seçilir. Sınıflandırma için, çok sınıflı çapraz entropi kullanılır. Sınıflandırma problemleri için genellikle softmax aktivasyon fonksiyonu, negatif log olasılığı / çok sınıflı çapraz entropi (MCXENT) ile birlikte kullanılmak istenir. Softmax aktivasyon fonksiyonu, çıktıların toplamı 1 olacak şekilde sınıflar üzerinde olasılık dağılımı sağlar. Bu yüzden bu uygulamada negatif logaritma olasılığı kullanılmıştır.

Sezgisel olarak eğitim verilerinin sınıflandırılması zayıfsa kayıp yüksek olacaktır, iyi sınıflandırma olmuşsa düşük kayıp çıkacaktır.

Veri normalizasyonu: Yapay sinir ağları en iyi kullandıkları veriler -1 ile 1 arasında bir aralıkta tutulduğunda çalışırlar. Bunun nedeni, gradyan düşüşü kullanılarak eğitilmeleridir ve bunların aktivasyon fonksiyonlarının genellikle -1 ile 1 arasında bir etkin aralığı vardır. Hızlı bir şekilde doyurulmayan bir aktivasyon fonksiyonu kullanılsa bile değerleri bu aralıkta sınırlamak genellikle performansı geliştirir.

Verileri normalize etmek için bu çalışmada, [0 255] aralığında olan gri tonlamalı görüntüler [0 1] aralığına normalize edilmiştir.

Veri dağılımı: Sürekli değerler; -1 ile 1, 0 ile 1 aralığında olmasını veya ortalama 0 ve standart sapma 1 ile normal olarak dağıtılması istenir. Bunun tam olması gerekmez, ancak girdilerin yaklaşık olarak bu aralıkta olması eğitim sırasında yardımcı olacaktır.

Ayrık sınıflar; çıktı ve sınıflandırma problemleri için genellikle bir sıcak gösterim kullanılır. Yani, 3 sınıf varsa, veriler her bir 3 sınıf için sırasıyla [1,0,0], [0,1,0] veya [0,0,1] olarak gösterilecektir.

Hem eğitim verileri hem de test verileri için aynı normalleştirme yöntemini kullanmak çok önemlidir.

4. ARAŞTIRMA BULGULARI ve DENEYSEL SONUÇLAR

4.1 Uygulama Platformu

Tez çalışması kapsamında önerilen yüz mimikleri analiz yaklaşımlarını uygulamak için kullanılan kişisel bilgisayarın konfigürasyonu ve yazılım platformu aşağıda belirtilmiştir.

- Intel Core i7-4790,
- 3.60 GHz CPU,
- 16 GB RAM,
- 256 GB SSD,
- Matlab R2016a,
- Eclipse Java IDE.

Çizelge 4.1 Geliştirilen yaklaşımların yaklaşık eğitim süreleri

	JAFFE	KDEF	MUG
HOG+SVM	5 dakika	15 dakika	60 dakika
Derin Öğrenme	30 dakika	70 dakika	390 dakika

Ayrıca aşağıdaki kütüphanelerden yararlanılmıştır.

- **Computer Vision System Toolbox**

Bu kütüphane, bilgisayar görme sistemlerini tasarlamak ve simule etmek için algoritmalar, işlevler ve uygulamalar sağlar. MATLAB'ın bu araç kitinden yüz tespiti, öznitelik çıkarımı ve sınıflandırma aşamasında yararlanılmıştır.

- **DeepLearning4J**

Derin öğrenme uygulamaların geliştirildiği birçok farklı kütüphane bulunmaktadır (Tenog 2015). Uygulamanın geliştirildiği platforma, yazılım diline, bunun gibi birkaç faktöre bağlı olarak kütüphane seçimi yapılabilir. Bu çalışmada JAVA için kullanılabilen, açık kaynak kodlu DeepLearning4j (DL4j) kütüphanesi ile YMA geliştirilmiştir. DL4J, çok katmanlı derin sinir ağlarını yapılandıran, eğiten ve dağıtan, alana özel java tabanlı bir araç setidir (toolkit). DL4j, çok katmanlı sinir ağlarını yapılandırmak ve esnek hesaplama grafikleri oluşturmak için araçlar sağlar.

4.2 Veri Kümeleri

Bu tez kapsamında yöntemler geliştirilirken JAFFE, KDEF ve MUG veritabanlarından yararlanılmıştır. Bu veri setlerinden aşağıda kısaca bahsedilmiştir.

4.2.1 JAFFE veri seti

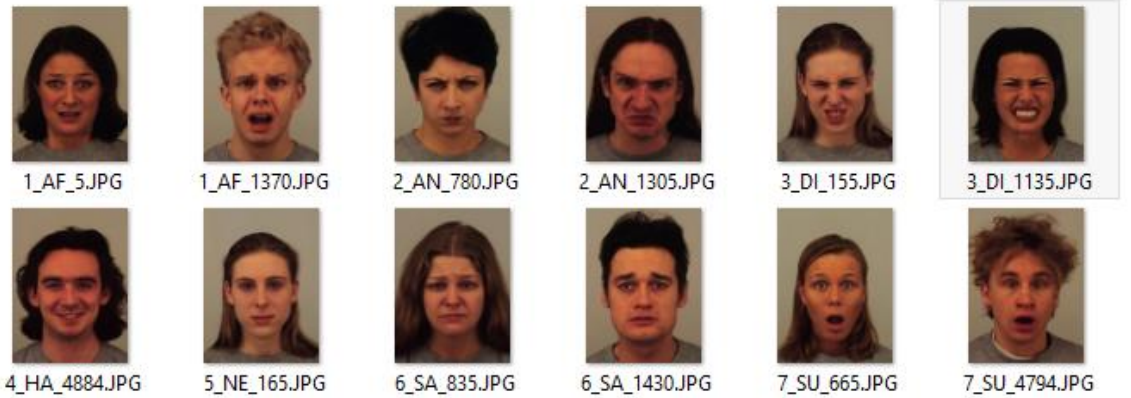
Japon Kadın Yüz Mimik Veritabanı (JAFFE), 10 kadın denek tarafından verilen yedi farklı mimikle birlikte 213 görüntüden oluşmaktadır (Lyons 1998). Fotoğraflar Kyushu Üniversitesi Psikoloji Bölümü'nde çekilmiştir. Veritabanı Michael Lyons, Miyuki Kamachi ve Jiro Gyoba tarafından planlanıp birleştirilmiştir. Bu veritabanına Ekman'ın altı evrensel duygusuyla birlikte ek olarak nötr kategorisi de dahil edilmiştir. Veritabanındaki her resim, ifade edilen mimik ile tutarlılık ve ilişki sağlamak için 60 farklı Japon sınıf öğrencisi tarafından derecelendirilmiştir. Şekil 4.1, mevcut yüzlerin bir örneğini göstermektedir.



Şekil 4.1 JAFFE veritabanından örnek resimler

4.2.2 KDEF veri seti

Karolinska Directed Emotional Faces (KDEF), yaşları 20 ile 30 arasında değişen, 35 kadın 35 erkek olmak üzere 70 kişinin 1500 adet mimik resmini içermektedir (Goeleven vd. 2008). Bu çalışmada 490 adeti kullanılmaktadır. Bu veri seti, Lundqwist, Flykt ve Ohman tarafından 1998 yılında tasarlanmıştır. Resimler 5 farklı açıdan çekilmiştir. Bireylerde sakal, bıyık, gözlük ve makyaj bulunmamaktadır. Şekil 4.2’de örnek yüz mimik resimleri yer almaktadır.



Şekil 4.2 KDEF veritabanından örnek resimler

4.2.3 MUG veri seti

MUG veritabanı yüz mimik tanıma sistemlerinin geliştirilmesi ve değerlendirilmesi için Multimedia Understanding Group tarafından geliştirilmiştir (Aifanti vd. 2010). Yüksek çözünürlük, tekdüze aydınlatma vb. gibi çok sayıda kısıtlamaların üstesinden gelmek için oluşturulmuştur. Seksen altı denek FACS el kılavuzunda tanımlanan mimik prototiplerine göre temel mimikleri gerçekleştirmişlerdir. Deney için bu veri setinden 9649 adet resim seçilmiştir.



Şekil 4.3 MUG veritabanından örnek resimler

4.3 Geliştirilen İki Yaklaşımın Karşılaştırılması

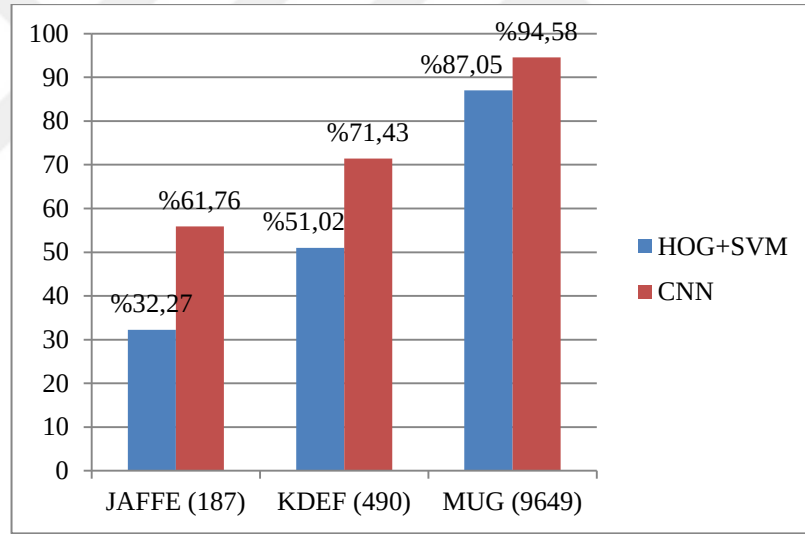
Çalışmada HOG-SVM temelli ve derin öğrenme temelli KSA ile yüz mimik analiz yaklaşımları geliştirilmiştir. Bu yaklaşımlar JAFFE, KDEF ve MUG veritabanları üzerinde geliştirilmiş ve test edilmiştir. JAFFE veritabanı ön cephe, sabit duruşlu resimler içermektedirken, KDEF ve MUG dinamik duruş (düz, tam yandan, 30° yandan) ve değişken ışık sistemi içermektedir. Bu veritabanları % 90'lık kısımlara bölünmüş ve bu kısım eğitim için kullanılmıştır. Geriye kalan % 10 kısmı ise test için kullanılmıştır. Son olarak test resimlerinin doğru tahmin sonuçları sayılmış ve tanıma oranları hesaplanmıştır.

İlk öncelikle HOG-SVM bazlı yüz mimik analizi yaklaşımı bu veritabanlarının test için ayrılan resimleri üzerinde test edilmiştir. YMA probleminde veri sayısı sıkıntısı

olduğundan az veritabanında düşük bir başarı oranı çıkmıştır. Fakat veri kümesindeki eğitilen veri sayısı arttıkça başarı da artmıştır.

İkinci olarak derin öğrenme tabanlı yüz mimik analizi test edilmiştir. Bu yaklaşımda da HOG-SVM tabanlı yaklaşımda olduğu gibi veri sayısı arttıkça başarı artmıştır.

Şekil 4.4'te YMA için geliştirilen iki yaklaşımın karşılaştırılması görülmektedir. Test edilen üç veri setindeki sonuçlara göre, Derin öğrenme temelli KSA yapısı, HOG-SVM yapısından daha başarılı çıkmıştır. Çünkü KSA öz niteliklerin elle çıkarıldığı HOG-SVM tabanlı geleneksel makine öğrenme yaklaşımının aksine, gradyan eğimi gibi yinelemeli algoritmaları kullanarak, eğitim veri setinden doğrudan öz nitelikleri çıkarmayı öğrenmektedir.



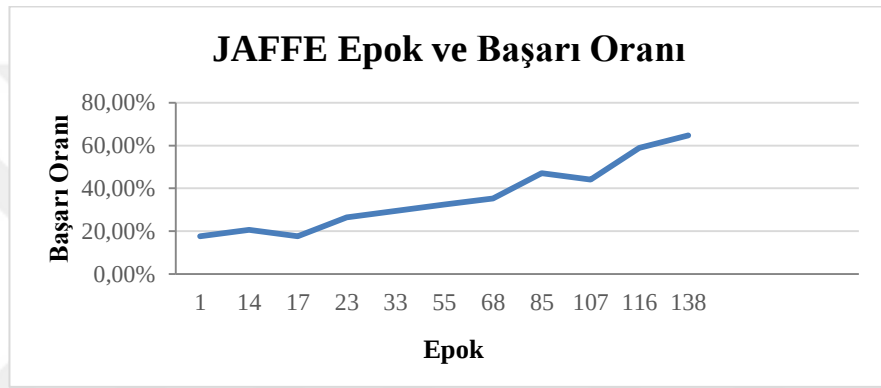
Şekil 4.4 HOG-SVM ve Derin öğrenme tabanlı KSA yaklaşımlarının karşılaştırılması

4.4 KSA Değişen Parametre Değerlerine Göre Elde Edilen Grafikler

KSA yapısının başarılı sonuç vermesi için parametre seçimi oldukça önemlidir. Bu yüzden bu parametrelerin neye göre değiştiği, parametrenin hangi durumunda başarının arttığını test etmek için bu değerler sürekli denenmiştir. En uygun konfigürasyona ulaşmak için gözlemlenen bazı hiperparametrelerin etkileri bu kısımda incelenmiştir.

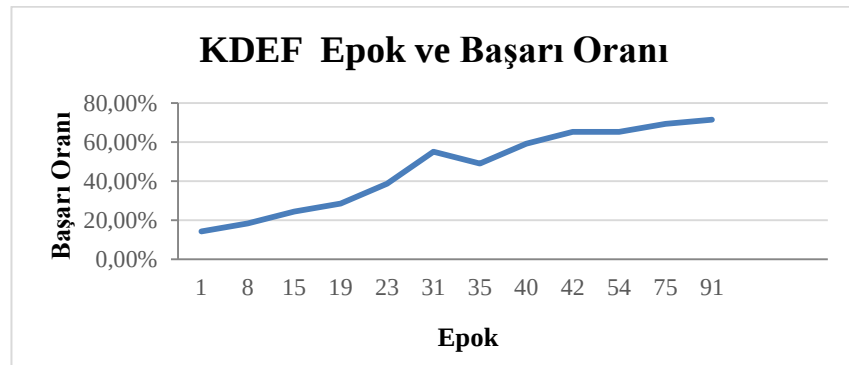
4.4.1 Epok sayısı ve test verilerinin başarı oranı

Derin öğrenme tabanlı geliştirilen sistemde epok sayısının YMA'nin başarısına etkisini belirlemek için birçok epok sayısı test edilmiştir. Düşük epok sayısında sistemin başarı oranı düşük çıkarken, fazla epok sayısında ağın görünmeyen verileri ezberlenmeye yani aşırı uyuma neden olmuştur. JAFFE veritabanı için yapılan deneyler sonucunda epok sayısı 138 olarak belirlenmiştir. Şekil 4.5 denenen epok değerlerinin grafiğini göstermektedir.



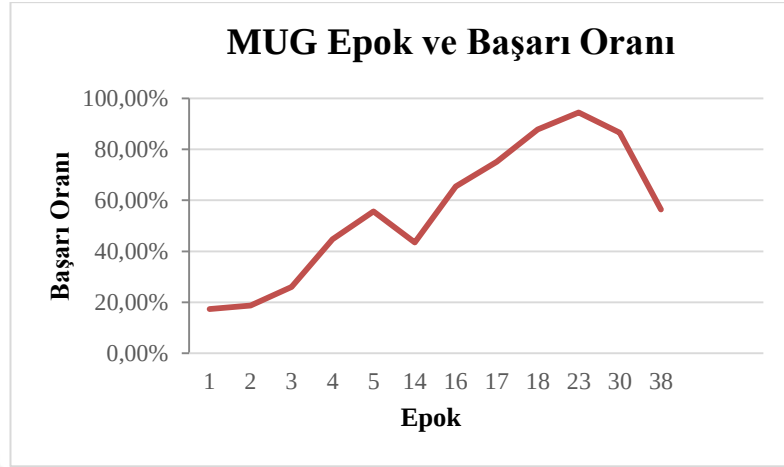
Şekil 4.5 JAFFE veri seti için epok ve başarı oranı grafiği

Şekil 4.6'da KDEF veri seti için sistemde birçok epok sayısı test edilmiş ve en iyi epok sayısı 91 epok olarak belirlenmiştir.



Şekil 4.6 KDEF veri seti için epok ve başarı oranı grafiği

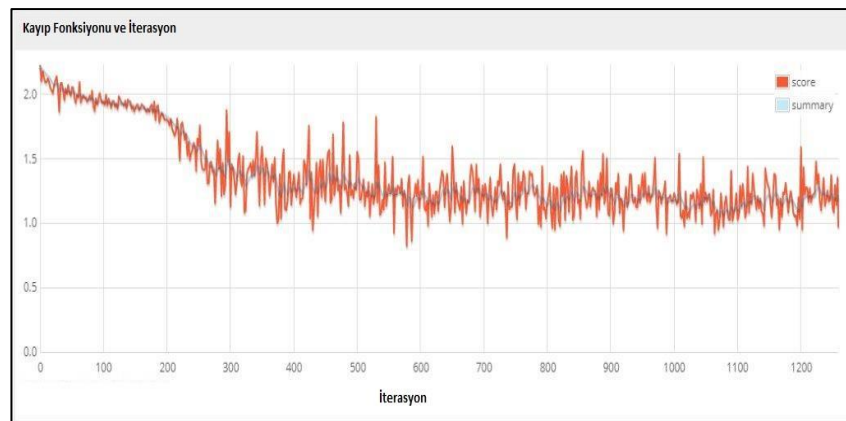
Şekil 4.7’de MUG veri seti için en iyi başarı 23 epok belirlenerek alınmıştır.



Şekil 4.7 MUG veri seti için epok ve başarı oranı grafiği

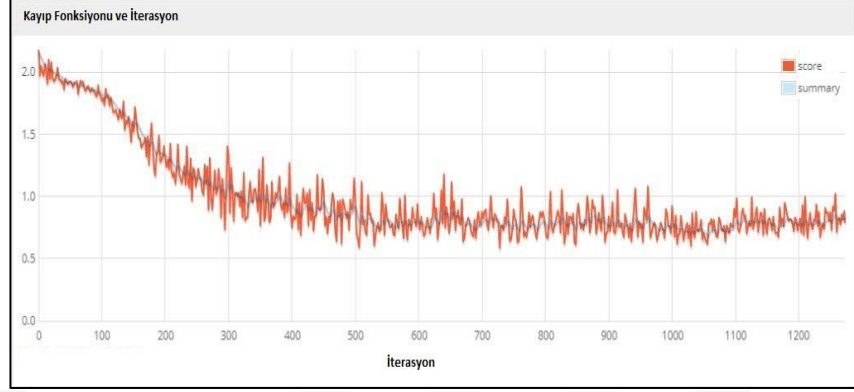
4.4.2 KSA kayıp fonksiyonu ve iterasyon arasındaki ilişki

Bu kısımda kayıp fonksiyonun geliştirilen modeli ve öğrenme dinamiklerini nasıl etkilediği araştırılmıştır. Bunun için her veri seti test edilmiştir. Şekil 4.8’de iterasyon arttıkça kayıp fonksiyonun değeri düşmüş, belli bir değerden sonra ise sabitlenmiştir. Daha başarılı eğitim için kayıp fonksiyonun düşmesi istenen bir faktördür.



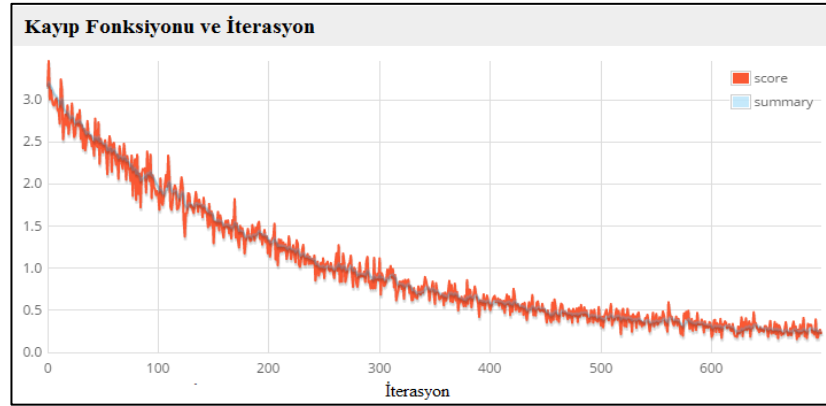
Şekil 4.8 JAFFE veri setinin kayıp fonksiyon-iterasyon grafiği

Yine KDEF veritabanında yapılan deneylere göre mevcut yığın üzerindeki iterasyonlar arttıkça kayıp fonksiyonun değeri düşmüştür.



Şekil 4.9 KDEF veri setinin kayıp fonksiyon-iterasyongrafiği

Gerçekleşmesi en çok istenilen kayıp fonksiyon-iterasyon grafiği MUG veritabanından elde edilmiştir. Zaten en çok başarının bu veritabanından alınmış olması da bunu göstermektedir.



Şekil 4.10 MUG veri setinin kayıp fonksiyon-iterasyongrafiği

4.5 KSA Hata Matrisleri

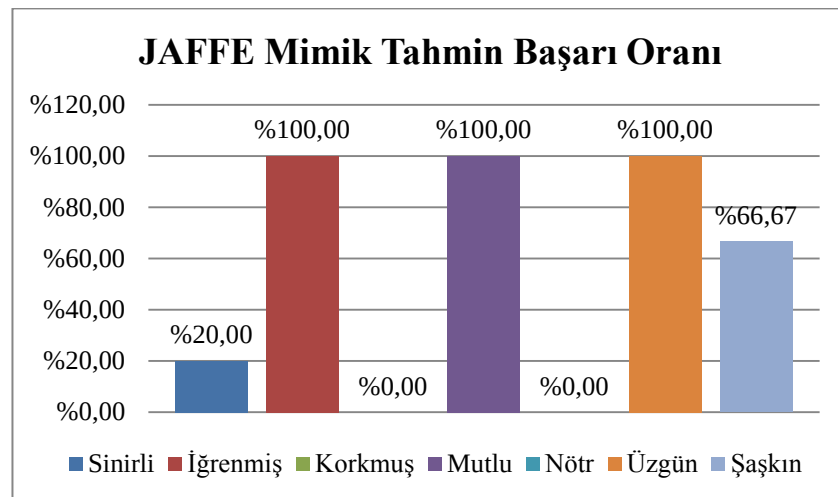
Softmax çıktı katmanı test edilen her bir görüntü için yedi mimik sınıftan hangi mimik etiketine ait olma olasılığını çıkarmaktadır. Örneğin çizelge 4.4'te 118 adet sinirli olarak

etiketlenmiş test görüntülerinden sistem 108'ini sinirli, 2'sini nötr, 8'ini ise üzgün olarak tahmin etmiştir. Böylece sinirli etiketi için % 91,53 sınıflandırma doğruluğu hesaplanmıştır. Test görüntüleri yedi mimik durumu için de ayrı ayrı hesaplanmış ve hata matrisleri aşağıdaki gibi oluşturulmuştur.

Çizelge 4.2 JAFFE veritabanının hata matrisi

	Tahmin							
		Sinirli	İğrenmiş	Korkmuş	Mutlu	Nötr	Üzgün	Şaşkın
Gerçek	Sinirli	%20,00	%40,00	%0,00	%20,00	%0,00	%20,00	%0,00
	İğrenmiş	%0,00	%100,00	%0,00	%0,00	%0,00	%0,00	%0,00
	Korkmuş	%0,00	%100,00	%0,00	%0,00	%0,00	%0,00	%0,00
	Mutlu	%0,00	%0,00	%0,00	%100,00	%0,00	%0,00	%0,00
	Nötr	%0,00	%66,67	%0,00	%0,00	%0,00	%33,33	%0,00
	Üzgün	%0,00	%0,00	%0,00	%0,00	%0,00	%100,00	%0,00
	Şaşkın	%0,00	%16,67	%0,00	%0,00	%0,00	%16,67	%66,67

JAFFE veritabanında hata matrisine bakılacak olursa mutlu, iğrenmiş ve üzgün durumunu hatasız tanımıştır. Ancak aynı başarıyı korkmuş durumunda gösterememiş, iğrenmiş etiketiyle karıştırmıştır. Şekil 4.11 her mimik için başarı oranlarının grafiğini göstermektedir.

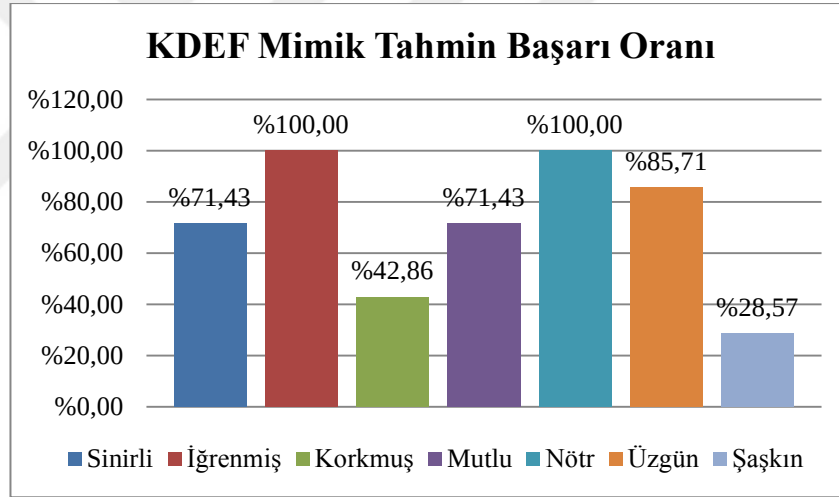


Şekil 4.11 JAFFE mimik tahmin başarı oranı

Çizelge 4.3 KDEF veritabanının hata matrisi

Gerçek	Tahmin							
		Sinirli	İğrenmiş	Korkmuş	Mutlu	Nötr	Üzgün	Şaşkın
	Sinirli	%71,43	%28,57	%0,00	%0,00	%0,00	%0,00	%0,00
	İğrenmiş	%0,00	%100,00	%0,00	%0,00	%0,00	%0,00	%0,00
	Korkmuş	%0,00	%0,00	%42,86	%0,00	%0,00	%28,57	%28,57
	Mutlu	%0,00	%14,29	%0,00	%71,43	%0,00	%14,29	%0,00
	Nötr	%0,00	%0,00	%0,00	%0,00	%100,00	%0,00	%0,00
	Üzgün	%0,00	%0,00	%0,00	%0,00	%14,29	%85,71	%0,00
	Şaşkın	%0,00	%0,00	%28,57	%0,00	%42,86	%0,00	%28,57

KDEF veritabanının sonuçlarına göre sistem, iğrenmiş ve nötr durumunu hatasız tanımış, en düşük başarıyı şaşkın etiketinden almıştır.



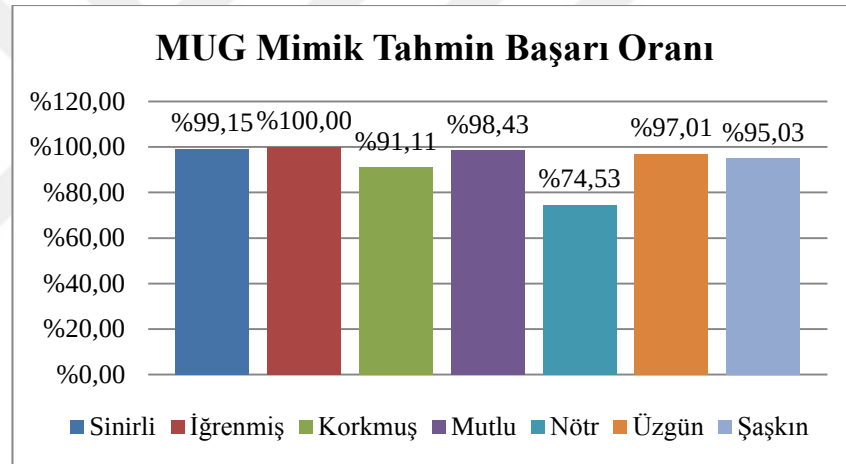
Şekil 4.12 KDEF mimik tahmin başarı oranı

Veri sayının artmış olması mimiklerin daha iyi eğitilmesini sağlamış haliyle her mimik için ayrı ayrı başarı oranları artmıştır.

Çizelge 4.4 MUG veritabanının hata matrisi

Gerçek	Tahmin							
		Sinirli	İğrenmiş	Korkmuş	Mutlu	Nötr	Üzgün	Şaşkın
	Sinirli	%99,15	%0,85	%0,00	%0,00	%0,00	%0,00	%0,00
	İğrenmiş	%0,00	%100,00	%0,00	%0,00	%0,00	%0,00	%0,00
	Korkmuş	%0,00	%0,00	%91,11	%1,11	%0,00	%3,33	%4,44
	Mutlu	%0,00	%1,57	%0,00	%98,43	%0,00	%0,00	%0,00
	Nötr	%7,55	%1,89	%1,89	%10,38	%74,53	%0,94	%2,83
	Üzgün	%0,00	%1,20	%0,00	%1,80	%0,00	%97,01	%0,00
	Şaşkın	%0,55	%0,00	%1,66	%1,10	%1,10	%0,55	%95,03

MUG veritabanının sonuçlarına göre en başarısız mimik, % 77,90 doğrulukla şaşkın mimiği olmuştur.



Şekil 4.13 MUG veritabanının mimik tahmin başarı oranı

5. SONUÇ

Bu tez çalışmasında, alınan bir fotoğraftan ‘mutlu’, ‘üzgün’, ‘korkmuş’, ‘sinirli’, ‘şaşkın’, ‘iğrenmiş’, ‘nötr’ olmak üzere yedi farklı evrensel mimik sınıfının durum analizi için iki yaklaşım geliştirilmiştir. Bu yaklaşımlar JAFFE, KDEF ve MUG veri setleri üzerinde ayrı ayrı geliştirilip, test edilmiştir. İlk öncelikle bu üç veri setinde mevcut her bir resim için Viola ve Jones algoritması ile resimden yüz tespiti yapılmış, yüz dışındaki tüm faktörler elenmiştir. Daha sonra bu yüz görüntüleri gri tonlamalı görüntüye dönüştürülmüş, 64x64 boyutuna indirilmiştir. Böylelikle tüm resimler artık önemli işlevlerin gerçekleştirilmesi için hazır hale getirilmiştir. İlk yaklaşım olarak HOG-SVM tabanlı yüz mimikleri durum analizi yaklaşımı geliştirilmiştir. Ön işlemlerden geçirilmiş olan bu yüz görüntülerinde mimiklerin HOG ile öznitelikleri çıkarılmış, SVM ile bu öznitelikler sınıflandırılarak yedi mimik sınıfından birine tahmin edilmiştir. İkinci yaklaşım olarak Derin Öğrenme ile yüz mimikleri durum analizi yaklaşımı geliştirilmiştir. Yine ön işlemlerden geçirilmiş yüz görüntülerinin Derin sinir ağlarının özelleşmiş bir biçimi olan Konvolüsyonel Sinir Ağları mimarisi önce yüksek seviyeli öznitelikleri çıkarmış ve daha sonra görüntünün her bir mimik grubuna ait olma olasılığını hesaplamıştır. Sınıflandırıcı olasılık değerlerinin interpolasyonu en son tahmini mimik durumunu üretmiştir. En son tartışma ve bulgular kısmında geliştirilen bu iki yaklaşımın başarı oranları karşılaştırılmıştır.

5.1 Öneriler

Büyük veri setlerinin kullanılabilirliği ve GPU tarafından sağlanan ucuz hesaplama gücü derin öğrenmenin popülerliğini artırmıştır. Ancak veri kümelerinin sınırlı olduğu yüz mimik analiz probleminde durum böyle değildir. Sonraki çalışmalarda daha büyük ve doğal koşullarda hazırlanmış bir veri kümesi hazırlanıp test edilebilir. Ayrıca 3B görüntüler için yüz mimikleri analizi gerçekleştirmek için konvolüsyonel sinir ağı modelleri geliştirilebilir.

KAYNAKLAR

- Ackley, D., Hinton, G. and Sejnowski, T. 1985. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(1), 147-169.
- Ahmed, F., Bari, H. and Hossain, E. 2014. Person-Independent Facial Expression Recognition Based on Compound Local Binary Pattern (CLBP). *The International Arab Journal of Information Technology*, 11(2).
- Aifanti, N., Papachristou, C. and Delopoulos, A. 2010. The MUG facial expression database. In *Image Analysis for Multimedia Interactive Services*, 11th International Workshop on IEEE, pp. 1-4.
- Barlett, M.S., Littlewort, G., Lainscsek, C. and Fasel, I. 2004. Machine Learning Methods for Fully Automatic Recognition of Facial Expressions and Facial Actions. *IEEE Conference: Systems, Men and Cybernetics*, pp. 592-597.
- Bengio, Y. 2009. Learning Deep Architectures for AI. *Foundations and Trends in Machine Learning*, 2(1), 1-127.
- Britz, D. 2015. Web Sitesi: <http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/>. Erişim Tarihi: 28.02.2017.
- Chellappa, R., Wilson, C.L. and Sirohey, S. 1995. Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, 83(5), 705-740.
- Cortes, C. and Vapnik, V. 1995. Support-vector networks. *Machine learning*, 20(3), 273-297.
- Dahmane, M. and Meunier, J. 2011. Emotion Recognition using Dynamic Grid-based HoG Features. *Automatic Face & Gesture Recognition and Workshops, IEEE International Conf on*, pp. 884-888.
- Dai, Y., Shibata, Y., Hashimoto, K., Ishii, T., Osuo, A., Katamachi, K., Nokuchi, K., Kakizaki, N. and Cai, D. 2000. Facial Expression Recognition of Person without Language Ability Based on the Optical Flow Histogram. In *Proceedings of the 5th Int. Conf. on Signal Processing, Beijing*, vol. 2, pp. 1209-1212.
- Dalal, N. and Triggs, B. 2005. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conf on*, pp. 886-893.
- Darwin, C. 1872. *The Expression of the Emotions in Man and Animals*. J. Murray, London.
- Deng, W., Hu, J., Zhang, S. and Guo, J. 2015. DeepEmo: Real-world Facial Expression Analysis via Deep Learning. *Visual Communications and Image Processing*.

- Deniz, O., Bueno, G., Salido, J. and Torre, F.D.L. 2011. Face recognition using Histograms of Oriented Gradients. *Pattern Recognition Letters* - Elsevier, 32 (12), 1598-1603.
- Devries, T., Biswaranjan, K. and Taylor, G.W. 2014. Multi-task Learning of Facial Landmarks and Expression. *Computer and Robot Vision, Canadian Conf on*.
- Dima, C. 2016. Web Sitesi: <http://www.cristiandima.com/basics-of-support-vector-machines/>. Erişim Tarihi: 02.04.2017.
- Donato, G., Bartlett, M.S., Hager, J.C., Ekman, P. and Sejnowski, T.J. 1999. Classifying facial actions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, 974-989.
- Duan, K.B. and Keerthi, S.S. 2015. Which Is the Best Multiclass SVM Method? An Empirical Study. *International Workshop on Multiple Classifier Systems*, vol. 3541, 278-285.
- Ekman, P. and Friesen, W.V. 1971. Constants across cultures in the face and emotion, *J. Personality Social Psychol*, 17(2), 124–129.
- Ekman, P. and Friesen, W.V. 1978. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press.
- Farley, B.G., and Clark, W.A. 1954. Simulation of Self-Organizing Systems by Digital Computer. *IRE Transactions on Information Theory*, 4(4), 76-84.
- Ghimire, D. and Lee, J. 2013. Geometric Feature-Based Facial Expression Recognition in Image Sequences Using Multi-Class AdaBoost and Support Vector Machines. *Sensors*, 13(6), 7714-7734.
- Ghimire, D., Jeong, S., Lee, J. and Park, S.H. 2017. Facial expression recognition based on local region specific features and support vector machines. *Multimedia Tools and Applications*, 76(6), 7803–7821.
- Gill, E. 2013. Web Sitesi: <https://gilscvblog.com/2013/08/18/a-short-introduction-to-descriptors/#more-3>. Erişim Tarihi: 05.06.2017.
- Gizatdinova, Y. and Surakka, V. 2006. Feature-Based Detection of Facial Landmarks from Neutral and Expressive Facial Images. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 28(1), 135-139.
- Goeleven, E., Raedth, R. D., Leyman, L. and Verschuere, B. The Karolinska Directed Emotional Faces: A validation study. *Cognition & Emotion*, 22(6), 1094-1118.
- Gokturk, S.B., Bouguet, J.Y., Tomasi, C. and Girod, B. 2002. Model-based face tracking for view-independent facial expression recognition. in *Proceedings of the 5th IEEE International Conf. on Automatic Face and Gesture Recognition*, pp. 287-293.

- Golovan, A. 2000. Neurobionic Algorithms of Low-Level Image Processing. Proc. Second All-Russia Scientific Conf. Neuroinformatics, vol. 1, pp. 166-173.
- Happy, S.L. and Routray, A. 2015. Robust facial expression classification using shape and appearance features. Advances in Pattern Recognition, 8th Intern. Conf on.
- He, S., Wang, S., Lan, W., Fu, H. and Ji, Q. 2013. Facial Expression Recognition Using Deep Boltzmann Machine from Thermal Infrared Images. Affective Computing and Intelligent Interaction, Humaine Association Conf, 239-244.
- Hinton, G.E. 2009. Deep belief Networks. Scholarpedia, 4(5), 5947.
- Hinton, G.E. 2012. A Practical Guide to Training Restricted Boltzmann Machines. Lecture Notes in Computer Science, 7700, 599-619.
- Hinton, G.E. and Salakhutdinov, R.R. 2006. Reducing the dimensionality of data with neural Networks. Science, 313 (5786), 504–507.
- Hinton, G.E., Osindero, S. and Teh, Y.W. 2006. A Fast Learning Algorithm for Deep Belief Nets. Neural Computation, 18(7), 1527-1554.
- Huang, C.L. and Huang, Y.M. 1997. Facial expression recognition using model-based feature extraction and action parameters classification. Journal of Visual Communication and Image Representation, vol. 8, 278-290.
- Jain, A.K., Mao, J. and Mohiuddin, K.M. 1996. Artificial neural Networks: a tutorial. Computer March, pp. 31-44.
- Jung, H., Lee, S., Park, S., Kim, B., Kim, J., Lee, I. and Ahn, C. 2015. Development of deep learning-based facial expression recognition system. Frontiers of Computer Vision 21st Korea-Japan Joint Workshop on.
- Karn, U. 2016. Web Sitesi: <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>. Erişim tarihi: 05.05.2017.
- Khatri, N.N., Shah, Z.H. and Patel, S.A. 2014. Facial Expression Recognition: A Survey. International Journal of Computer Science and Information Technologies, 5(1), 149-152.
- Khorrami, P., Paine, T. and Huang, T. 2015 Do deep neural networks learn facial action units when doing expression recognition?. In Proceedings of the IEEE International Conf on Computer Vision Workshops, pp. 19-27.
- Kotsia, I. and Pitas, I. 2007. Facial Expression Recognition in Image Sequence Using Geometric Deformation Features and Support Vector Machines. IEEE Trans. Image Process, 16(1), 172-187.
- Kumari, J., Rajesh, R. and Pooja, K.M. 2015. Facial expression recognition: A survey. Second International Symposium on Computer Vision and the Internet, Procedia Computer Science, no. 58, 486 – 491.

- Lawrence, S., Giles, C. L., Tsoi, A. C. and Back, A. D. Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8(1), 98-113.
- LeCun, Y., Bengio, Y. and Hinton, G. 2015. Deep Learning. *Nature*, 521(7553), 436-444.
- LeCun, Y., Bengio, Y., Hinton, G. 2015. Deep learning. *Nature*, 521(7553), 436-444.
- Lektemur, E. 2017. Web Sitesi: <http://www.kesit.info/tag/denetimli-ogrenme/>, Erişim Tarihi: 13.06.2017.
- Lien, C.C., Lin, L.Y. and Tu, C.M. 2008. A New Appearance-Based Facial Expression Recognition System with Expression Transition Matrices. In *Proceedings of the 3rd International Conf on Automatic Face and Gesture Recognition*.
- Liu, C., Yang, Y. and Tang, C. 2010. An Improved Method for Multi-class Support Vector Machines. *Measuring Technology and Mechatronics Automation Int. Conf on*, pp. 504-508.
- Lyons, M.J., Akamatsu, S., Kamachi, M. and Gyoba, J. 1998. Coding Facial Expressions with GaborWavelets. *Proceedings, Third IEEE International Conference on Automatic Face and Gesture Recognition*, IEEE Computer Society, pp. 200-205.
- Mase, K. and Pentland, A. 1991. Recognition of Facial Expression from Optical Flow. *IEICE Transactions*, 74(10), 3474-3483.
- McCulloch, W.S. and Pitts W. 1943. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4), 115–133.
- Mehrabian, A. 1968. Communication without words[J]. *Psychology Today*, 2(4), 53-56.
- Minsky, M. and Papert, S. 1969. *Perceptrons*. M.I.T. Press *Perceptrons*.
- Nabiyev, V.V. 2010. Yapay Zeka: İnsan – Bilgisayar Etkileşimi (3.baskı). Seçkin, 752, Ankara.
- Nurhanna, A.A. and Othman, M.F. 2017. Multi-class Support Vector Machine Application in the Field of Agriculture and Poultry: A Review. *Malaysian Journal of Mathematical Sciences*, vol. 11, 35-52.
- Pantic, M. and Rothkrantz, L.J.M. 2003. Toward an affect-sensitive multimodal human-computer interaction. *Proceedings of the IEEE*, 91(9), 1370–1390.
- Qin, W. and Zeng-fu, W. 2005. Recognition of Human Facial Expression Based on 3D Data[J]. *Computer Simulation*, 22(7), 99-103.
- Raghuvanshi, A. and Choksi, V. 2016. Facial Expression Recognition with Convolutional Neural Networks. *CS231n Course Projects*.

- Rosenblatt, F. 1958. The Perceptron: A Probabilistic Model For Information Storage and Organization in the Brain. *Psychological Review*, 65(6), 386-408.
- Roux, N.L. and Bengio, Y. 2008. Representational Power of Restricted Boltzmann Machines and Deep Belief Networks. *Neural Computation*. 20(6), 1631-1649.
- Rowley, H. A., Baluja, S. and Kanade, T. 1998. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1), 23-38.
- Rumelhart, D.E., Hinton, G.E. and Williams R.J. 1986. in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT, Foundations, vol. 1, 318-362.
- Sajjad, M., Shah, A., Jan, Z., Shah, S.I. and Baik, S.W. 2017. Facial appearance and texture feature-based robust facial expression recognition framework for sentiment knowledge discovery. *Cluster Computing*, pp. 1–19.
- Sakai, T., Nagao, M. and Kanade, T. 1972. Computer analysis and classification of photographs of human faces, in *Proc. Japan Computer Conference*, pp. 2-7.
- Samuel, A.L. 1959. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3), 210-229.
- Shin, M., Kim, M. and Kwon, D.S. 2016. Baseline CNN structure analysis for facial expression recognition, pp. 724-729.
- Shinohara, Y. and Otsu, N. 2004. Facial Expression Recognition Using Fisher Weight Maps. *Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition*.
- Smolensky, P. 1986. *Information Processing in Dynamical Systems: Foundations of Harmony Theory*, 195-281.
- Suwa, M., Sugie, N. and Fujinmora, K. 1978. A Preliminary Note on Pattern Recognition of Human Emotional Expression. In *proceedings of 4th International Conference on Pattern Recognition*, pp. 408-410.
- Teglor. 2015. Web sitesi: <http://www.datasciencecentral.com/profiles/blogs/here-are-15-libraries-in-various-languages-to-help-implement-your>. Erişim tarihi: 09.05.2017.
- Turing, A.M. 1950. Computing Machinery and Intelligence. *Mind Association*, 59(236), 433-460.
- Vasilev, I. 2017. Web Sitesi: <https://www.toptal.com/machine-learning/an-introduction-to-deep-learning-from-perceptrons-to-deep-networks>. Erişim Tarihi: 12.06.2017.

- Viola, P. and Jones, M. 2001. Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition. IEEE Computer Society Conference on, vol. 1, 511-518.
- Vretos N., Nikolaidis N. and Pitas I. 2009. A Model-Based Facial Expression Recognition Algorithm using Principal Components Analysis. In Proceedings of the 16th IEEE International Conference on image Processing, pp. 3301-3304.
- Vukadinovic, D. and Pantic, M. 2005. Fully automatic facial feature point detection using Gabor feature based boosted classifiers. Systems, Man and Cybernetics, IEEE International Conference on.
- Web Sitesi: <http://www.uplifiers.com/evrensel-yuz-ifadeleri-charles-darwine-ovgu>. Erişim Tarihi: 03.06.2017.
- Web Sitesi: <https://deeplearning4j.org/documentation>. Erişim Tarihi: 08.05.2017.
- Whitehill, J., Omlin, C.W. 2006. Haar Features for FACS AU Recognition. IEEE Conference on Automatic Face and Gesture Recognition.
- Xu, M., Cheng, W., Zhao, Q., Ma, L. and Xu, F. 2015. Facial expression recognition based on transfer learning from deep convolutional Networks. 11th Intern. Conf. on Natural Computation, pp. 702-708.
- Yacoob, Y. and Davis, L.S. 1996. Recognizing human facial expressions from long image sequences using optical flow. IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(6), 636-642.
- Zeng, Z., Pantic, M., Roisman, G.I. and Huang, T.S. 2009. A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 31, 39- 58.
- Zhang, C., and Woodland, P.C. 2015. Parameterised sigmoid and ReLU hidden activation functions for DNN acoustic modelling. In 16th Annual Conf of the International Speech Communication Association, pp. 3224-3228.
- Zhang, Z., Lyons, M., Schuster, M. and Akamatsu, S. 1998. Comparison Between Geometry-based and Gabor-Wavelet- based Facial Expression Recognition Using Multi-layer Perceptron, Proc. IEEE Int'l Conf. Face and Gesture Recognition, pp. 454-459.

ÖZGEÇMİŞ

Adı Soyadı : Büşra EMEK SOYLU

Doğum Yeri : Erzurum

Doğum Tarihi : 01.12.1990

Medeni Hali : Evli

Yabancı Dili : İngilizce

Eğitim Durumu

Lise : Nevzat Karabağ Anadolu Öğretmen Lisesi (2008)

Lisans : Ankara Üniversitesi Bilgisayar Mühendisliği (2014)

Yüksek Lisans : Ankara Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Bölümü (Eylül 2015–Temmuz 2017)

Çalıştığı Kurumlar

Bilgisayar Mühendisi, PTT Genel Müdürlüğü, 2015 – 2015

Araştırma Görevlisi, Erzurum Teknik Üniversitesi Mühendislik ve Mimarlık Fakültesi,
Bilgisayar Mühendisliği, 2016–Devam ediyor