

**T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



**YÜZ İFADESİ TANIMA BAŞARIMI İÇİN İLGİLİ BÖLGE
BELİRLEME VE ÇEKİŞMELİ ÜRETİCİ AĞLAR
KULLANILARAK ÖRNEK ARTIRIM YÖNTEMLERİNİN
GELİŞTİRİLMESİ**

Ömer Faruk SÖYLEMEZ

Doktora Tezi

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

HAZİRAN 2021

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Bilgisayar Mühendisliği Anabilim Dalı

Doktora Tezi

**YÜZ İFADESİ TANIMA BAŞARIMI İÇİN İLGİLİ BÖLGE
BELİRLEME VE ÇEKİŞMELİ ÜRETİCİ AĞLAR KULLANILARAK
ÖRNEK ARTIRIM YÖNTEMLERİNİN GELİŞTİRİLMESİ**

Tez Yazarı

Ömer Faruk SÖYLEMEZ

Danışman

Prof. Dr. Burhan ERGEN

HAZİRAN 2021

ELAZIĞ

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Bilgisayar Mühendisliği Anabilim Dalı

Doktora Tezi

Başlığı:	Yüz İfadesi Tanıma Başarımı İçin İlgili Bölge Belirleme ve Çekişmeli Üretici Ağlar Kullanılarak Örnek Artırım Yöntemlerinin Geliştirilmesi
Yazarı:	Ömer Faruk SÖYLEMEZ
İlk Teslim Tarihi:	06.06.2021
Savunma Tarihi:	28.06.2021

TEZ ONAYI

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına göre hazırlanan bu tez aşağıda imzaları bulunan jüri üyeleri tarafından değerlendirilmiş ve akademik dinleyicilere açık yapılan savunma sonucunda OYBİRLİĞİ ile kabul edilmiştir.

Danışman:	Prof. Dr. Burhan ERGEN Fırat Üniversitesi, Mühendislik Fakültesi	<i>İmza</i> Onayladım
Başkan:	Prof. Dr. Mehmet KAYA Fırat Üniversitesi, Mühendislik Fakültesi	Onayladım
Üye:	Prof. Dr. İbrahim TÜRKOĞLU Fırat Üniversitesi, Mühendislik Fakültesi	Onayladım
Üye:	Prof. Dr. Mehmet Sıraç ÖZERDEM Dicle Üniversitesi Mühendislik Fakültesi	Onayladım
Üye:	Doç. Dr. Muhammet Fatih TALU İnönü Üniversitesi Mühendislik Fakültesi	Onayladım

Bu tez, Enstitü Yönetim Kurulunun/...../20..... tarihli toplantısında tescillenmiştir.

İmza
Doç. Dr. Kürşat Esat ALYAMAÇ
Enstitü Müdürü

BEYAN

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım “Yüz İfadesi Tanıma Başarımı İçin İlgili Bölge Belirleme ve Çekişmeli Üretici Ağlar Kullanılarak Örnek Artırım Yöntemlerinin Geliştirilmesi” Başlıklı Doktora Tezimin içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

28.06.2021

Ömer Faruk SÖYLEMEZ



ÖNSÖZ

Gerçekleştirilen doktora tez çalışmasında, yüz ifadesi tanıma başarımının artırılması için iki farklı çalışma önerilmiştir. İlk çalışma kapsamında, yüz üzerindeki yüz ifadesi ile ilgili farklı alanların kırılması için bir yöntem sunulmuştur. Ardından bu imgeler, farklı mimarilere sahip olan evrişimsel sinir ağları üzerinde eğitilmiştir. Bu çalışma sonucunda kırılan yüz bölgesi ve ağ seçimi hakkında önemli bilgiler edinilmiştir. İkinci yöntem kümesi kapsamında, sınırlı sayıda örnek içeren bir yüz ifade veri tabanında örnek artırımı için çekişmeli üretici ağlar temelli bir örnek artırımı önerilmiştir. Çekişmeli üretici ağlar ile artırılan örnekler ile gerçekleştirilen eğitim işlemi sonucunda, literatürdeki diğer çalışmalara oranla daha yüksek başarımlar elde edilmiştir.

İnsanoğlu, doğumundan ölümüne kadar bir eğitim süreci içerisindeydir. Anne ve baba ile aile içinde başlayan bu süreç, eğitim kurumları ve çevre ile dışarıda devam eder. Bu sürecin büyük bir bölümünün öznesi olan eğitimcinin rolü ise sürecin ehemmiyetinden ötürü oldukça büyüktür. Farklı uzmanlar tarafından eğitimcilerin sahip olması gereken birçok nitelik belirtilmiştir. Bu görüşlere katılmakla birlikte, naçizane fikrimce bunların en önemlisi sabırdır. Ebeveynlerin evlatlarına gösterdiği sabır ve şefkati, talebelerine gösteren eğitimci zannınca en mahir eğitimcidir. İniş ve çıkışlarla dolu olan yüksek lisans ve tez süreçlerimde, tökezlediğim, boş verdiğim, çıkmaza girdiğim ve bunaldığım her anda benden ümidini kesmeyerek sabır ile gayrete sevk eden hocam Prof. Dr. Burhan ERGEN'e yürekten teşekkürlerimi sunarım. Göstermiş olduğunuz bu sabırlı ve kollayıcı tavrı, öğrencilerime karşı bir sorumluluk olarak şiar edineceğime ant içerim.

Hayatım boyunca doğru veya yanlış verdiğim kararların arkasında bir dağ gibi duran annem ve babama, iyi kalpli ve fedakâr eşime ve bana baba olma duygusunu yaşatan dünyalar güzeli kızım Erva Duru'ya teşekkürlerimi sunarım.

7 YAŞINA GİREN KIZIM ERVA DURU'YA İTHAFEN...

Ömer Faruk SÖYLEMEZ

ELAZIĞ, 2021

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	iv
İÇİNDEKİLER	v
ÖZET	vii
ABSTRACT	viii
ŞEKİLLER LİSTESİ	ix
TABLolar LİSTESİ	x
SİMGELER VE KISALTMALAR	xi
1. GİRİŞ	1
1.1. Tezin Amacı ve Kapsamı	2
1.2. Tezde Gerçekleştirilen Çalışmalar ve Literatüre Katkısı	3
1.3. Tezin Organizasyonu	4
2. YÜZ TESPİTİ VE YÜZ İFADESİ TANIMA	5
2.1. Yüz tespiti.....	5
2.1.1. Viola-Jones Nesne Tespit Çatısı.....	7
2.1.2. Yönlü Gradyanlar Histogramı	11
2.2. Yüz İfadesi Tanıma	13
2.2.1. Yüz Hareketi Kodlama Sistemi	15
2.2.2. Bağdeğer – Uyarım Uzayı.....	15
2.2.3. Yüz İfadelerinin Kategorilendirilmesi.....	16
2.2.4. Yüz Anahtar Noktaları	17
3. DERİN ÖĞRENME	18
3.1. Derin Öğrenme	18
3.2. Evrişimsel Sinir Ağları	19
3.2.1. Katmanlar	19
3.2.2. Kayıp Fonksiyonları	25
3.2.3. Eğitim Sürecinde Kullanılan Parametreler	26
3.3. Evrişimsel Sinir Ağları Tabanlı Derin Öğrenme Modelleri.....	28
3.3.1. AlexNET	28
3.3.2. VGGNet mimarisi	29
3.3.3. Inception mimarisi.....	32
3.3.4. ResNet mimarisi	34
3.3.5. Kullanılan ESA Modelleri ve Seçim Nedenleri	35
3.4. Çekişmeli Üretici Ağlar.....	36
3.4.1. FCGAN	38
3.4.2. ProGAN.....	38
3.4.3. StyleGAN	40
3.4.4. StyleGAN2	42
4. EVRİŞİMSSEL SİNİR AĞLARI VE ÇEKİŞMELİ ÜRETİCİ AĞLAR KULLANILARAK GERÇEKLEŞTİRİLEN YÜZ İFADESİ TANIMAYA YÖNELİK ÇALIŞMALAR	44
4.1. Kullanılan Veri Tabanı	44
4.2. Evrişimsel Sinir Ağları Kullanılarak Gerçekleştirilen Yüz İfadesi Tanıma	44

4.2.1. Önerilen Yüz Kırpma Metodolojisi.....	45
4.2.2. Ağların Hazırlanması ve Eğitimi.....	47
4.3. Çekişmeli Üretici Ağlar ile Örnek Artırımı	50
4.3.1. Gerçekleştirilen Ön İşlem Aşamaları	53
4.3.2. Çekişmeli Üretici Ağ Yardımıyla Gerçekleştirilen Örnek Artırımı	54
5. BULGULAR	60
5.1. Evrişimsel Sinir Ağları Kullanılarak Gerçekleştirilen Yüz İfade Tespitine Dair Bulgular	60
5.2. Çekişmeli Üretici Ağlar Yardımıyla Örnek Artırımı Sonrasında Gerçekleştirilen Yüz İfade Tespitine İlişkin Elde Edilen Bulgular	62
5.3. Sonuçların Literatürdeki Diğer Çalışmalar ile Karşılaştırılması.....	63
6. SONUÇLAR VE DEĞERLENDİRME	65
KAYNAKLAR.....	66
ÖZGEÇMİŞ	



ÖZET

Yüz İfadesi Tanıma Başarımı İçin İlgili Bölge Belirleme ve Çekişmeli Üretici Ağlar Kullanılarak Örnek Artırım Yöntemlerinin Geliştirilmesi

Ömer Faruk SÖYLEMEZ

Doktora Tezi

FIRAT ÜNİVERSİTESİ

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Haziran 2021, Sayfa: xi + 72

İnsanlar arası iletişimde en önemli araçlardan biri olan yüz ifadesi tanıma, teknolojinin geldiği noktanın da yardımıyla geçtiğimiz son on yılın en önemli araştırma alanlarından birisi haline gelmiştir. Yüz ifadelerinin otomatik olarak sınıflandırılması, başta insan bilgisayar etkileşimi olmak üzere klinik psikoloji, nöroloji, güvenlik sistemleri, akıllı ortamlar, müşteri memnuniyeti ve reklam sektörü gibi birçok alanda kullanılmaktadır. Taşınabilir cihazlar ile görüntüleme ve kaydetme işlemlerinin bireyler seviyesinde gerçekleştirildiği günümüzde, yüz ifadelerini hızlı ve gürbüz bir şekilde tanıyabilecek uygulama ve yöntemlere ihtiyaç vardır.

Bu tez kapsamında, yüz ifadesi tanımanın başarımını artırmak üzere iki farklı çalışma sunulmuştur. Yapılan ilk çalışmada, yüz ifadesini temsil eden yüz üzerindeki ilgili alanın seçimi için bir kırpma metodolojisi önerilmiştir. Ardından bu metodoloji yardımıyla elde edilen imgeler, farklı mimarilere sahip olan üç evrişimsel sinir ağı üzerinde eğitilmiştir. Gerçekleştirilen çalışmanın sonucunda, yüz ifadesi içeren alan ve ağ seçiminin, yüz ifadesi tanıma üzerindeki etkisi hakkında önemli bulgular elde edilmiştir. Tez kapsamında sunulan ikinci çalışmada, kısıtlı sayıda örnek içeren bir yüz ifadesi veri tabanının örnek sayısını artırmak için çekişmeli üretici ağlar temelli bir veri artırımı yaklaşımı sunulmuştur. Çekişmeli üretici ağlar yardımıyla artırılan imgeler kullanılarak, yüz ifadesi tanıma için bir evrişimsel sinir ağı eğitilmiştir. Verinin dağılımını doğru bir şekilde yansıtan çekişmeli üretici ağlar yardımıyla artırılan örnekler ile eğitilen ağı, yüz ifadesi tanıma işlemi için literatürdeki çalışmalara kıyasla daha yüksek başarımler sergilediği gözlemlenmiştir.

Anahtar Kelimeler: Yüz İfadesi Tanıma, Derin Öğrenme, Çekişmeli Üretici Ağlar

ABSTRACT

Development of a Region of Interest Detection and a Generative Adversarial Network Based Image Augmentation Approach for Improving Facial Expression Recognition Performance

Ömer Faruk SÖYLEMEZ

Ph.D. Thesis

FIRAT UNIVERSITY
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

June 2021, Pages: xi + 72

Computer-aided analysis of facial expressions, which is one of the most important tools in interpersonal communication, has become one of the most important research areas of the last decade with the help of technology. The automatic classification of facial expressions is used in many areas such as human-computer interaction, clinical psychology, neurology, security systems, smart environments, customer satisfaction and the advertising industry. In today's world, where imaging and recording operations are carried out at the level of individuals with portable devices, there is a need for applications and methods that can recognize facial expressions quickly and robustly.

Within scope of this paper, two different studies were carried out to improve the performance of facial expression recognition. Within the first study, a cropping methodology is proposed for detection of the region of interest on the face that represents the facial expression. Images obtained with this methodology are trained on three convolutional neural networks with each having a different architecture. As a result, important findings were obtained on region of interest and network selection for facial expression recognition. Within the second study, a data augmentation approach based on generative adversarial networks is presented in order to augment a facial expression database that contains a limited number of samples. Augmented images were used to train a convolutional neural network. According to the test phase results of the mentioned network, it was found out that, training with augmented samples that accurately reflects the distribution of the data yields a better performance than the rest of the literature.

Keywords: Facial Expression Recognition, Deep Learning, Generative Adversarial Networks

ŞEKİLLER LİSTESİ

	Sayfa
Şekil 2.1. İntegral imge ve Haar benzeri dikdörtgen özelliklerin gösterimi.	8
Şekil 2.2. AdaBoost algoritması için temel bir pseudo kod.....	9
Şekil 2.3. Dikkat mekanizmasına sahip kademeli yapının şeması.....	10
Şekil 2.4. YGH için akış şeması.	11
Şekil 2.5. Bağdeğer – uyarım uzayı.....	16
Şekil 2.6. 68 adet yüz anahtar noktasına sahip olan bir şablon	17
Şekil 3.1. 4x4'lük bir girdi imgesi üzerinde, 2x2'lik bir filtre ve 1x1'lik kaydırma ile gerçekleştirilen evrişim işlemi.....	21
Şekil 3.2. Farklı şekillerde uygulanan havuzlama işlemi sonucunda elde edilen özellik haritaları	23
Şekil 3.3. Sigmoid, Tanh, ReLU ve Leaky ReLU fonksiyonlarının grafik üzerinde gösterimi.....	25
Şekil 3.4. 5x5 boyutundaki bir özellik haritası üzerinde gerçekleştirilen parçalı evrişimin gösterimi.	31
Şekil 3.5. Inception-v1 sürümünde önerilen inception modülleri.....	33
Şekil 3.6. Inception modülünün faktörizasyon işlemi.	34
Şekil 3.7. ResNet mimarisinde kullanılan residual modüller.	35
Şekil 3.8. Geleneksel bir Çekişmeli Üretici Ağ mimarisinin gösterimi.	37
Şekil 3.9. FCGAN ile Toronto Face Dataset üzerinde gerçekleştirilen eğitim sonucu üretilen yüzler.....	38
Şekil 3.10. ProGAN'a ait eğitim süreci ve oluşturulan görüntüler.....	40
Şekil 3.11. ProGAN ve StyleGAN üreticilerinin karşılaştırılması.	41
Şekil 3.12. AdaIN normalizasyonu sonucu imgelerde oluşan su damlacıkları kalıntıları.	42
Şekil 3.13. Kademeli büyümenin neden olduğu geçiş kalıntıları.	43
Şekil 4.1. Veri seti için seçilen örnek görüntüler. a. nötr b. mutlu c. iğrenme d. şaşırma e. korku	44
Şekil 4.2. YGH + DVM ile gerçekleştirilen yüz tespit işlemi.	45
Şekil 4.3. AGM kullanılarak uydurulan 68 adet yüz özellik noktası.....	46
Şekil 4.4. Uygulama kapsamında tasarlanan üç adet ilgi bölgesi.	47
Şekil 4.5. Tez kapsamında gerçekleştirilen ikinci çalışmaya ait akış şeması.	52
Şekil 4.6. Gerçekleştirilen ön işlem adımları.....	53
Şekil 4.7. CK+ veri tabanından alınan imgeler ve bu imgelerin StyleGAN2 latent uzayındaki karşılıkları. 55	
Şekil 4.8. Farklı sınıflar için latent vektör aritmetiği ile üretilen örnekler.	56
Şekil 4.9. Üretilen imgeler ve art işlem sonucunda elde edilen yüz ifadesi ile ilgili alanlar.	58
Şekil 5.1. Bütün ölçeklerde gerçekleştirilen ESA eğitimlerinin, eğitim ve test başarımları.....	60

TABLÖLER LİSTESİ

	Sayfa
Tablo 3.1. AlexNET modelinin katmanları, boyutları ve filtre parametreleri.....	30
Tablo 3.2. VGGNet mimarisi modellerinin konfigürasyonları.	32
Tablo 4.1. Önerilen 9 katmanlı modelin mimari yapısı.	48
Tablo 4.2. CK+ veri tabanındaki her bir sınıf için baz örnek ve artırılmış örnek sayıları.....	57
Tablo 4.3. Kullanılan Inceptionv3 mimarisi için eğitim parametreleri	59
Tablo 5.1. Kullanılan ağların her bir ölçek için ortalama başarımları.....	61
Tablo 5.2. Farklı ölçeklerde bir döngü boyunca ağ eğitimi için gereken saniye cinsinden ortalama süre ...	61
Tablo 5.3. En yüksek başarımları sağlayan ölçek 3 için 6 ifadeyi içeren hata matrisi.	62
Tablo 5.4. Her bir grup için elde edilen en yüksek başarımlar ve genel ortalama başarımlar	63
Tablo 5.5. Gerçekleştirilen çalışmaların literatürdeki diğer çalışmalar ile karşılaştırılması	64

SİMGELER VE KISALTMALAR

Kısaltmalar

AGM	Aktif Görünüm Modeli
BoW	Bag of Words
ÇÜA	Çekişmeli Üretici Ağ
DCGAN	Deep Convolutional Generative Adversarial Network
DSA	Derin Sinir Ağı
DÜM	Derin Üretici Modeller
DVM	Destek Vektör Makineleri
EEG	Elektroenselelografi
EKG	Elektrokardiyogram
EMG	Elektromiyografi
ESA	Evrişimsel Sinir Ağı
FCGAN	Fully Connected Generative Adversarial Network
GPGPU	General Purpose Graphics Processing Unit
HB	Hareket Birimi
KYM	Kısıtlı Yerel Model
LRN	Local Response Normalization
NMF	Nonnegative Matrix Factorization
ProGAN	Progressive Growing Generative Adversarial Network
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SIFT	Scale Invariant Feature Transform
SURF	Speeded Up Robust Features
TBA	Temel Bileşen Analizi
VOK	Varyasyonel Otomatik Kodlayıcı
YGH	Yönlü Gradyanlar Histogramı
YHKS	Yüz Hareketi Kodlama Sistemi
YİT	Yüz İfadesi Tanıma
YİÖ	Yerel İkili Örüntüler
YSA	Yapay Sinir Ağları

1. GİRİŞ

Teknolojinin hızla geliştiği günümüzde, teknolojik cihazlar hayatımızın önemli bir parçası haline gelmiştir. Özellikle geçtiğimiz son on yılda yaşanan gelişmeler sayesinde görüntü yakalamayı, depolamayı ve işlemeyi kolaylaştıran akıllı telefon, bilgisayar ve tablet gibi cihazlar, toplumun bütün bireyleri tarafından yaygın olarak kullanılmaktadır. Elde edilen bu görüntüler, bilgisayarla görü yardımıyla birçok uygulama tarafından işlenmekte ve amaca yönelik sonuçlar elde edilmektedir. Bu uygulamalardan birisi de Yüz İfadesi Tanımadır (YİT). YİT, başta insan bilgisayar etkileşimi olmak üzere, kişisel güvenlik uygulamaları, sürücü yorgunluk tespit sistemleri, market araştırmaları, sosyalleşebilir robotlar, kişisel avatarlar ve hasta duygu durum tespiti gibi birçok yenilikçi uygulama alanında kullanılmaktadır [1–3].

Yüz ifadeleri, ses ve vücut hareketleri ile birlikte, bireyler arası duygu aktarımında kullanılan temel paylaşım araçlarından biridir. Aktarımın şiddeti, keskinliği ve içtenliği gibi parametreler göz önüne alındığında, yüz ifadelerinin ses ve vücut hareketlerinden daha etkin bir duygu aktarımı aracı olduğu söylenebilir. Bu nedenle yüz ifadeleri, geçmişten bu yana insanoğlunun duygularını aktarmak için kullandığı sözel olmayan iletişim türlerinin en önemlisi olarak kabul edilmektedir. Zira yüz, doğası gereği bireyin duygularının önlenemez bir biçimde ortaya çıktığı ilk aktarım organıdır. Ayrıca bireylerin dış dünyayı algılamada en etkin olarak kullandıkları görsel etkileşimin de merkez noktasıdır. Diğer aktarım araçlarından daha fazla sayıda duygu sergileme yeteneğine sahip olan yüz ifadeleri, bahsedilen sebeplerden ötürü bireyler arası duygu aktarımında kullanılan en etkin iletişim biçimi olarak konumlandırılabilir.

İnsanoğlu yüz ifadelerini ayırt etme konusunda yeteneklidir. Araştırmalar henüz iki ve üç aylık bebeklerin mutluluk, üzüntü, kızgınlık, şaşkınlık ve korku gibi yüz ifadelerini doğru bir şekilde ayırt edebildiklerini göstermektedir [4,5]. Ayrıca yüz ifadelerinin bireyler ve kültürlere özgü olmayıp bütün kültür ve alt-kültürlerde aynı anlama geldiğini kanıtlayan çalışmalar da mevcuttur [6].

İnsan için her ne kadar günlük sıradan bir aktivite olsa da makine öğrenmesi yöntemleriyle gerçekleştirilen YİT, probleme özgü birçok zorluk içermektedir. Yetersiz ve dengesiz ışıklandırma, yüzün bir kısmının kapalı olması, farklı bakış açıları ve düşük çözünürlük başlıca bu zorluklara örnek olarak verilebilir. Bilgisayar destekli YİT yeni bir uğraş değildir. Söz konusu alan, başta bilgisayar bilimleri ve psikolojinin ana konusu olmakla birlikte, mevcut konuma son 30 yıl boyunca bu alanda gerçekleştirilen çalışmalar sayesinde ulaşılmıştır. Bu süreç boyunca makine öğrenmesi alanında geliştirilen sınıflandırma ve özellik çıkarımı yöntemleri, bilgisayar destekli YİT'in başarımı ve yaygınlaşmasına katkıda bulunmuştur.

1.1. Tezin Amacı ve Kapsamı

Bu tez çalışmasının amacı, kontrollü ortamlarda gerçekleştirilen YİT başarımını, derin öğrenme yaklaşımları yardımıyla iyileştirmektir. YİT, birçok farklı uygulama başlığı altında kullanım alanına sahiptir. Bu nedenle YİT'in doğru bir şekilde gerçekleştirilmesi, söz konusu uygulamaların hedeflenen şekilde çalışması için oldukça önemlidir. Geleneksel yöntemlerle gerçekleştirilen YİT başarımı, ışıklandırma, yüzün kameraya bakış açısı, yüzün belirli kısımlarının örtülü olması, yaş, cinsiyet, ten rengi ve ırk gibi etkenlere bağlıdır. Bu nedenle geliştirilen yöntemlerin, farklı şartlar altında uygun sonuçlar verememeleri mümkündür. Geleneksek yöntemlerin aksine son yıllarda oldukça popülerleşen Evrişimsel Sinir Ağları (ESA) tabanlı yaklaşımlar, doğru parametreler ve yeterli eğitim örneği içeren veri setleri ile eğitildikleri takdirde, YİT konusunda geleneksel yöntemlere oranla çok daha yüksek başarımlar sergileyebilmektedirler.

ESA'lar çok sayıda katman ve parametreden oluşan mimarilerdir. Bu tarz derin mimarilerin eğitimi esnasında, eğitim verisinin ezberlenerek aşırı öğrenme gerçekleştirilmemesi için çok sayıda eğitim verisine ihtiyaç vardır. Kontrollü ortamlarda oluşturulan yüz ifadesi veri tabanları, örnek sayısı bakımından oldukça kısıtlıdır. Ayrıca süjeler tarafından sergilenen yüz ifadelerinin, ilgili ifadeyi yeterince iyi yansıtmadığı durumlar da söz konusudur. Bu ise zaten sınırlı sayıda olan örnek sayısını daha da aşağılara çekmektedir. Yetersiz sayıda örnek ile ESA'lar üzerinde gerçekleştirilen eğitim işlemi sonucunda, ağlarda sıklıkla genelleştirebilme sorunu gözlemlenmektedir. Bu da ilgili ESA'nın test aşamasında daha düşük bir başarımla çıkarım yapmasına yol açmaktadır. Aşırı öğrenmenin üstesinden gelmek için L1 ve L2 normalizasyonu gibi regülerizasyon teknikleri, seyreltme, yığın normalleştirme ve öğrenme transferi gibi yöntemler, araştırmacılar tarafından sıklıkla kullanılmaktadır.

Veri artırım yaklaşımları, yukarıda bahsedilen tekniklerden farklı olarak, aşırı öğrenmenin üstesinden gelmek için veri setini doğrudan artırarak sorunu kaynağından çözmeyi hedeflemektedirler. İlave eğitim verisi bulmanın mümkün olmadığı durumlarda veri artırım yaklaşımlarından sıklıkla faydalanılmaktadır. Veri artırım yaklaşımları, genel olarak uzaysal alanda çalışan temel imge manipülasyonları (yakınlaşma, kırpma, kaydırma, beyazlatma, renk uzayı manipülasyonu, rastgele silme vb.), çekirdek filtreleri (bulanıklaştırma, keskinleştirme vb.) ve veri çarpıtma yöntemleri (elastik deformasyon, optik sapma vb.) yardımıyla gerçekleştirilir. Genel olarak farklı veri setleri üzerinde bu tarz yaklaşımlar ile gerçekleştirilen veri artırma işlemi iyi sonuçlar verse de söz konusu YİT olduğu zaman bu geçerli değildir. Yüz ifadeleri, doğaları gereği küçük temsil değişiklikleri ile farklı ifadelere dönüşebilmektedirler. Bu nedenle uzamsal düzlemde gerçekleştirilen manipülasyonlara ve dönüşümlere karşı oldukça hassastırlar. Örneğin; bir yüze kesme, yakınlaşma veya kaydırma işlemi uygulandığı durumlarda, elde edilen yüzün ilgili ifadeyi temsil edecek bileşenlerden yoksun olması ihtimal dahilindedir. Buna benzer bir şekilde elastik deformasyonlar, ağzın, gözlerin ve kaşların yapılarını değiştirerek, bir yüz ifadesini tamamen farklı

bir yüz ifadesine dönüştürebilirler. Bunların yanı sıra uzamsal manipölasyon, verinin özniteliksel temsilini dikkate almayan bir yaklaşımdır. Bu şekilde oluşturulan örneklerin, veri setinin özniteliksel dağılımı ile aynı dağılıma sahip olmama durumu da söz konusudur. Bu da eğitim sürecini olumsuz etkileyebilecek bir faktör olarak öne çıkabilmektedir.

1.2. Tezde Gerçekleştirilen Çalışmalar ve Literatüre Katkısı

Bu tez çalışmasında, derin öğrenme yöntemleriyle yüz imgeleri üzerinde YİT'e yönelik deneysel çalışmalar gerçekleştirilmiştir. Gerçekleştirilen çalışmalar kapsamında YİT başarımının artırılması hedeflenmiştir.

İlk deneysel çalışmada, kontrollü ortamda elde edilen bir veri tabanı üzerinde, yüz ifadesini temsil eden yüz üzerindeki ilgili alanın seçiminin ve farklı ESA mimarilerinin, YİT üzerindeki etkisi incelenmiştir. Öncelikle ilgili alanların seçimi için bir kırpma metodolojisi önerilmiştir. Orijinal ve kırpılan imgeler kullanılarak, farklı tasarımlara sahip olan ağlar üzerinde eğitim süreci gerçekleştirilmiştir. Gerçekleştirilen bu çalışma neticesinde, farklı ölçeklere sahip olan ilgili alanlar ve farklı ağ tasarımlarının, YİT problemi üzerindeki başarımlarına ilişkin çıkarımlarda bulunulmuştur.

Yapılan ikinci çalışma kapsamında, ESA eğitimi için kısıtlı sayıda örnek içeren bir veri tabanının örneklerini artırmak için Çekişmeli Üretici Ağlar (ÇÜA) tabanlı bir yaklaşım sunulmuştur. Öncelikle veri tabanındaki imgeler bir takım önışlemden geçirilmiştir. Ardından bu imgelerin, sentetik yüz oluşturmada an itibariyle en başarılı ÇÜA olan StyleGAN2 [7] ağındaki temsilleri bulunmuştur. Aynı ifadeye sahip her bir ikili yüz imgesinin temsillerinin ortalamalarının alınmasıyla yeni yüz ifadeleri oluşturulmuştur. Bu yöntem yardımıyla veri tabanı, aynı öznitelik uzayında dağılımına sahip olacak bir şekilde genişletilmiştir. Çoğaltılan bu örnekler kullanılarak yapılan eğitim ve test işlemleri sonucunda yüksek başarımlar elde edilmiştir.

Yüz ifadesi tanıma üzerine gerçekleştirilen çalışmalar, örnek artırımı için sıklıkla uzamsal düzlemdeki manipölasyonlara başvurmaktadırlar. Lopes vd., Gauss gürültüsü ile yüz üzerindeki ifadeyi temsil eden özelliklerin konumlarını değiştirerek örnek artırımına gitmişlerdir [8]. Lin vd., 2 boyutlu yüz imgelerini 3 boyutlu uzaya haritalandırdıktan sonra farklı açılardan elde ettikleri 2 boyutlu yüz imgeleri ile örnek artırımını gerçekleştirmişlerdir [9]. Pitaloka vd., imgeleri normalleştirerek ve ardından gürültü ekleyerek bir örnek artırım yaklaşımı sunmuşlardır [10]. Bütün bu yaklaşımlar, uzamsal düzlemdeki manipölasyonlar olmakla birlikte, veri temsiliinde sıkıntılara yol açabilmektedirler.

Tez çalışması kapsamında, örnek artırımı için ÇÜA tabanlı bir yaklaşım kullanılmıştır. Literatürdeki uzamsal düzlem üzerinde gerçekleştirilen manipölasyonların aksine, ÇÜA ile gerçekleştirilen örnek artırımı sayesinde aynı öznitelik uzayında konumlanan yüz ifadeleri elde

edilmiştir. Benzer öznitelik dağılımına sahip olan imgeler yardımıyla gerçekleştirilen sınıflandırma sonucunda, literatürdeki benzer yöntemlere oranla daha yüksek başarımlar elde edilmiştir.

Bu tez çalışması kapsamında literatüre şu katkılar sağlanmıştır:

- Yüz ifadelerinin, yüz imgelerinden kırılması için bir metodoloji sunulmuştur.
- Tasarımsal olarak farklı mimarilere sahip olan ESA'ların, YİT problemi üzerindeki başarımları incelenmiştir.
- Yüz ifadesini temsil eden yüz üzerindeki ilgili alanın seçiminin YİT problemi üzerindeki etkisi incelenmiştir.
- Önceden eğitilmiş bir ÇÜA yardımıyla, kısıtlı sayıda örnek içeren bir veri tabanının örnek sayısını artırmak için ortalama yüzler tabanlı bir yaklaşım sunulmuştur.

1.3. Tezin Organizasyonu

Bu tez çalışmasının diğer kısımları şu şekildedir. İkinci bölümde, yüz tespiti ve YİT hakkında genel bilgiler verilmiş ve bu alanda yapılan bilimsel çalışmalara değinilmiştir. Üçüncü bölümde, makine öğrenmesi alanındaki gidişatı değiştiren derin öğrenmeye değinilmiştir. Bu kapsamda derin öğrenmenin kısa bir tarihi gelişimine, görüntü işleme alanında sıklıkla kullanılan ESA'lara, ESA'lara ilişkin işlem birimlerine ve tez kapsamındaki uygulamalarda kullanılan farklı ESA tasarımlarına yer verilmiştir. Son olarak, çalışmaların bir kısmında sentetik örnek oluşturmak için kullanılan ÇÜA'lara da ayrıca yer verilmiştir. Dördüncü bölümde, tez kapsamında gerçekleştirilen uygulamalara yer verilmiştir. Beşinci bölümde bu uygulamalara ilişkin bulgular verilmiş ve sonrasında sonuçlar ve değerlendirme bölümü ile tez çalışması sonlandırılmıştır.

2. YÜZ TESPİTİ VE YÜZ İFADESİ TANIMA

2.1. Yüz tespiti

Yüz tespiti, bilgisayarla görü literatüründe en çok çalışılan konulardan birisidir. Sadece bir nesne olarak yüzün zorlu doğasının ötesinde, aynı zamanda ilk adım olarak yüz tespitine ihtiyaç duyan sayısız uygulama nedeniyle yüz tespiti, araştırmacıların sıklıkla çalıştıkları bir alan olmuştur [11]. Yüz bölgesi, yüz imgeleri üzerinde çalışan bilgisayar destekli uygulamalar için birincil girdidir. Bu uygulamalara yüz tanıma ve doğrulama [12], gözetleme için yapılan yüz takibi [13], yüz ifadesi tespiti [14,15], cinsiyet ve yaş belirleme [16], güzellik değerlendirme [17], yüz rekonstrüksiyonu [18], imge ve video erişim sistemleri [19] ve dijital fotoğraf albümlerinin organizasyonu ve sunumu [20] gibi birçok örnek verilebilir. Başta Facebook olmak üzere birçok sosyal ağ, görüntü ve kişi etiketlemek için yüz tespiti modüllerini kullanmaktadırlar. Ayrıca birçok cep telefonu ve dijital kamera üreticisi, fotoğraf çekimi esnasında odaklanmaya yardımcı olmak üzere yüz tespitinden faydalanmaktadır. Sektörel olarak bakıldığında ise yüz tespiti, askeri, finansal ve kamu güvenliği alanında birçok uygulamaya sahiptir. Bu nedenle yüz bölgesinin doğru bir şekilde tespit edilmesi, bahsedilen bu alanlardaki çalışmalar için oldukça büyük önem arz etmektedir.

Bilgisayarla görünün ilk uygulamalarından olan yüz ve yüz özelliklerinin otomatik olarak tespitinin geçmişi 45-50 yıl öncesine dayanmaktadır [21–24]. İlk çalışmalar 1970'lerin başlarında, basit sezgisel ve antropometrik teknikler kullanılarak gerçekleştirilmiştir [23]. Bu teknikler, düz arka plan, ön cepheden çekilmiş yüz gibi tipik bir vesikalık fotoğraf senaryosuna benzer çeşitli varsayımlardan dolayı büyük ölçüde esnek değildir. Bu sebeple görüntü koşullarındaki herhangi bir değişiklik, yüz tespiti için yeniden ayar yapmayı gerektirmektedir. 1990'lara kadar bu alan üzerinde gerçekleştirilen çalışmalarda durgunluk, pratik yüz tanıma ve video kodlama sistemlerinin ve uygulamalarının gerçeğe dönüşmesiyle birlikte atlatılmış ve hızlı bir ivme kazanmıştır [25–27].

Geçtiğimiz son 20 yıl içerisinde ise yüz tanımaya özgü birçok önemli problemin çözümü için çok sayıda araştırmalar gerçekleştirilmiştir. Hareket, renk ve genelleştirilmiş bilgileri kullanan ve eskilerine oranla çok daha gülbüz çalışan segmentasyon şablonları sunulmuştur. İstatistik bilimi ve yapay sinir ağlarının kullanımı sayesinde aynı imge içerisinde farklı mesafe ve derinliklerde bulunan yüzlerin tespiti sağlanmıştır. Bunlara ek olarak, deforme edilebilen şablonlar ve yüz özelliklerini doğru bir şekilde bulabilen ve izleyebilen aktif konturlar gibi özellik çıkarıcıların tasarımında oldukça iyi ilerlemeler kaydedilmiştir. Fakat bütün bu çalışmalar sadece kısıtlı koşullarda yüksek başarımla sergileyebilmektedirler. Söz konusu herhangi bir kısıtlamanın olmadığı gerçek dünya koşullarında gerçekleştirilen yüz tespiti olduğunda, bahsedilen bu yöntemler arzulanan başarımla sağlayamamaktadır.

2001 yılında Viola-Jones tarafından sunulan nesne tespit çatısı, yüz ve yüz elemanlarının tespiti konusunda tam anlamıyla bir çığır açmıştır [28]. Yükseltme tabanlı kaskad sınıflandırıcı yapısı kullanan yöntem, yüz tespitini gerçek dünya şartlarında ve gerçek zamanlı olarak uygulanabilir kılmıştır. Bu aşamadan itibaren gerçek dünya koşullarında yüz tespiti üzerine birçok algoritma geliştirilmiştir. Modern bilgisayarların hızla gelişen işlemci gücü ve depolama kapasitelerinin yanı sıra, modern yüz tespit algoritmaları birçok gelişmeden faydalanmıştır. SIFT [29], Yönlü Gradyanlar Histogramı (YGH) [30], Yerel İkili Örüntüler (YİÖ) ve varyasyonları [31–33] ve bunun daha hızlı muadilleri olan SURF [34] ve DAISY [35] ve yukarıda bahsedilen özellikleri integral imgeler ile birleştiren integral kanal özellikleri [36,37] gibi gürbüz özellik çıkarım yöntemleri, yüz görünümünü tanımlamak için sıklıkla kullanılmaktadırlar. PASCAL [38,39], LFW [40], FDDB [41] gibi araştırma topluluklarının, gerçek dünyada nesne tespiti üzerine oluşturdukları veri tabanlarının yanı sıra internetin yaygınlaşmasıyla artan görsel verilerin (çoğunlukla veri paylaşımı hizmetleri ve arama motorları aracılığıyla) artışı, yüz tespiti üzerinde çalışan algoritmaları olumlu olarak etkilemiştir. Bunların yanı sıra yükseltme [42], Destek Vektör Makineleri (DVM) [43] ile birlikte bunların son yapısal uzantıları [44] ve Derin Sinir Ağı (DSA) mimarileri [45] gibi ayrıştırıcı özelliği çok yüksek olan sınıflandırma yöntemlerinin geliştirilmesi ile yukarıda bahsedilen büyük ölçekli veri setlerindeki veriler kullanabilmekte ve karmaşık deforme edilebilir objeler ve yüz modelleri eğitebilmektedir [46]. Son olarak OpenCV [47], deforme edilebilir parça tabanlı modeller [48] ve ESA eğitimi için altyapılar [49] gibi yüksek kaliteli kamuya açık kod depolarının geliştirilmesi ile YİT uygulamaları, herkes tarafından erişilebilir olmuştur.

Hjelmas ve Low [27], yüz tespit algoritmaları üzerinde gerçekleştirilen yöntemleri dört sınıfa ayırmıştır: Bilgi tabanlı yöntemler, şablon eşleştirme yöntemleri, özellikten bağımsız yaklaşımlar ve görünüm tabanlı yöntemler. Bilgi tabanlı yöntemler, yüz tespiti için insan bilgisine dayalı önceden tanımlanmış kuralları kullanmaktadırlar. Şablon eşleştirme yöntemleri, verilen görüntüde insan yüzünün tasvir edildiği yerleri belirlemek için önceden kaydedilmiş yüz şablonlarını kullanmaktadırlar. Özellikten bağımsız yaklaşımlar, poz açısı ve aydınlatmanın varyasyonlarına karşı gürbüz olan güçlü yapısal yüz özelliklerini bulmayı amaçlamaktadırlar. Son olarak görünüm tabanlı yöntemler, yüz tespiti için kullanılan temsili yüz eğitim imgelerinin yardımıyla yüz modellerini öğrenmeye çalışmaktadırlar.

Hjelmas ve Low [27] tarafından önerilen sınıflandırma, Viola-Jones tarafından sunulan nesne tespit çatısından sonra gerçekleştirilen çalışmaları sınıflandırmak için uygun değildir. Bu mihenk taşı çalışma sonrası gerçekleştirilen çalışmaları rijit-şablonlar tabanlı algoritmalar kümesi ve deforme edilebilir parça tabanlı modelleri kullanan yöntemler kümesi olarak iki kategoriye ayırmak çok daha doğru bir sınıflandırma şeması sunmaktadır [11]. Rijit-şablonlar tabanlı algoritma kümesi altında; yükseltme varyasyonları olarak Viola-Jones yüz tespit algoritması ve onun türevleri [28,50], çok sınıflı nesne tanımda olağanüstü başarımlar gösteren ve yüz tespiti çalışmalarında çok

geniş bir kullanım alanı bulan ESA ve Derin ESA tabanlı yöntemler [51,52] ve genelleştirilmiş Hough dönüşümünü temel alan yöntemler[53,54] bulunmaktadır. Deforme edilebilir parça tabanlı modelleri kullanan yöntemler kümesi altında; yüz kısımları arasında potansiyel deformasyonları modellemek üzere, Deforme edilebilir Parça Tabanlı Modelleri kullanan yöntemler vardır [24,46,55]. Bu yöntemler, aynı zamanda yüz tespiti ve yüz parçası lokalizasyonunu birlikte gerçekleştirebilirler [56]. Bu yöntem kümesi, genel olarak nesne tespit metodolojisinde kullanılan yöntemlerin varyasyonlarından ve uzantılarından oluşur [46,55]. Parça tabanlı yaklaşımları kullanan başka yöntemler de mevcuttur [57–59].

Çalışma kapsamında Viola-Jones nesne tespit çatısı ve YGH yöntemleriyle yüz tanıma gerçekleştirilmiştir. 2000’li yıllara damga vuran bu yöntemler, günümüzde sıklıkla yüz tespiti için kullanılmakta ve yüksek başarımlar sergilemektedirler.

2.1.1. Viola-Jones Nesne Tespit Çatısı

Viola-Jones nesne tespit çatısı, 2000’li yıllarda yüz tanıma için en çok ses getiren çalışmalardan biridir. Bu nesne tespit çatısı, yüz tanımının haricinde yüzün bileşenleri olan göz ve ağız ile birlikte, yaya tespiti için de sıklıkla kullanılmıştır. Gerçek zamanda çalışabilen Viola-Jones yüz dedektörü, için üç ana kısımdan oluşmaktadır: İntegral imge, AdaBoost ile öğrenilen sınıflandırıcı ve dikkat mekanizmasına sahip kademeli yapı.

İntegral İmge

Toplam alan tablosu olarak da bilinen integral imge, bir ızgaranın dikdörtgen bir alt kümesindeki değerlerin toplamını, hızlı ve verimli bir şekilde hesaplamak için kullanılan bir algoritmadır. Bu yaklaşım ilk olarak mipmaps’larda kullanılmak için sunulmuştur [60]. Viola-Jones, Haar-benzeri özelliklerin hızlı bir şekilde hesaplanabilmesi için integral imgeyi kullanmıştır. $ii(x, y)$, (x, y) konumundaki integral imge, $i(x', y')$ ise orjinal imge olmak üzere, integral imge, Denklem 2.1’deki gibi oluşturulur:

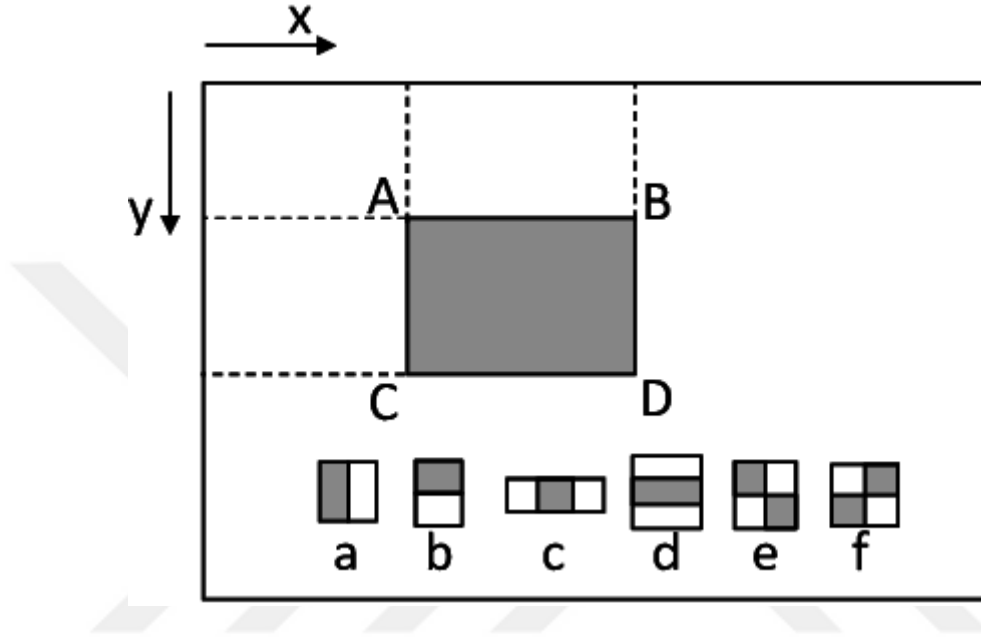
$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (2.1)$$

İntegral imgeyi kullanarak dikdörtgen bir alt alanın toplamını hesaplamak oldukça basittir. Şekil 2.1’de verilen ABCD dikdörtgen bölgesinin integral imgesi, sadece 4 adet dizi referansına ihtiyaç duyacak biçimde, Denklem 2.2.’de gösterildiği şekilde hesaplanır:

$$\sum_{(x,y) \in ABCD} i(x, y) = ii(D) + ii(A) - ii(B) - ii(C) \quad (2.2)$$

İntegral imge, aynı zamanda Şekil 2.1’de verilen Haar-benzeri dikdörtgen özelliklerin de hesaplanmasında kullanılır. Bu özellikler, farklı dikdörtgen bölgelerin ağırlıklı yoğunluk

farklarının birbirlerinden çıkarılması ile hesaplanır. Örneğin, Şekil 2.1’de özellik değeri, gri ve beyaz dikdörtgenler altında kalan piksel değerlerinin ortalama farkı bulunarak elde edilir. Dikdörtgenler köşeleri paylaştıkları için, iki dikdörtgen (a ve b) özelliğın hesaplanması için altı referans, üç dikdörtgen (c ve d) özelliğın hesaplanması için sekiz referans ve dört dikdörtgen (e ve f) özelliğın hesaplanması için dokuz referans yeterlidir.



Şekil 2.1. İntegral imge ve Haar benzeri dikdörtgen özelliklerin gösterimi.

AdaBoost Öğrenme

Yükseltme, her biri makul doğrulukta olan birçok "zayıf" hipotezi birleştirerek yüksek derecede doğruluğa sahip bir hipotez elde etmeye yarayan bir yöntemdir. Bu kapsamda AdaBoost (Adaptive Boosting) kullanılarak, düşük doğrulukta bulunan birçok özellik birleştirilerek yüksek doğrulukta sınıflandırıcılar elde edilmiştir.

AdaBoost algoritması için temel bir pseudo kod, Şekil 2.2’de verilmiştir. AdaBoost algoritması, genellikle daha pratik olan yükseltme algoritmalarına doğru atılan ilk adım olarak kabul edilir. Bu kısımda, AdaBoost’un genelleştirilmiş sürümü olan RealBoost hakkında bilgi verilecektir. RealBoost’un, AdaBoost’dan daha iyi başarımlar sergilediğini ortaya koyan çalışmalar mevcuttur [61,62].

x_i ’nin X uzayına ve y_i ’nin sınırlı Z etiket uzayına ait olduğu $S = \{(x_i, z_i), i = 1, \dots, N\}$ olarak tanımlanmış bir eğitim kümesi olsun. İkili sınıflandırma problemlerinde, pozitif örnekler için $z_i = 1$ ve negatif örnekler için $z_i = -1$ olmak üzere etiket uzayı, $Z = \{1, -1\}$ olarak belirlenir. Adaboost, girdi olarak verilen bir x örneğinin etiketini tespit etmek için $F^T(x) = \sum_{t=1}^T f_t(x)$ şeklinde bir artırmalı model kullanır. Burada $F^T(x)$, $F^T: X \rightarrow \mathbb{R}$ formunda gerçek değerli bir

fonksiyondur. Tahmin edilen etiket, $\text{sign}(\cdot)$ işaret fonksiyonu olmak üzere, $\hat{z}_i = \text{sign}(F^T(x_i))$ 'dir. İstatistiki bir açıdan bakıldığında AdaBoost algoritması, denklem 2.3'de verilen ve hedeflenen üssel kriterleri minimize etmek için, adaptif Newton güncellemelerini kullanarak eklemeli bir lojistik regresyon modelini uydurmaktadır:

$$L^T = \sum_{i=1}^N \exp\{-z_i F^T(x_i)\} \quad (2.3)$$

Girdi

- Eğitim örnekleri $S = \{(x_i, z_i), i = 1, \dots, N\}$.
- T , eğitilmesi gereken zayıf sınıflandırıcıların toplam sayısıdır.

Başla

- N_+ ve N_- eğitim kümesindeki pozitif ve negatif örneklerin sayısı olmak üzere ,
 $F^0(x_i) = \frac{1}{2} \ln\left(\frac{N_+}{N_-}\right)$ örnek skoru ile başla.

AdaBoost Öğrenme

$t = 1, \dots, T$ için:

1. Havuzdaki her bir Haar-benzeri özellik $h(x)$ için, Z skoru L^T 'yi minimize etmek üzere optimum eşik değeri H ve güven skoru c_1 ve c_2 'yi bul.
2. Minimum L^T 'ye sahip olan en iyi özelliği seç.
3. $F^t(x_i) = F^{t-1}(x_i) + f_1(x_i), i = 1, \dots, N$ olarak güncelle.
4. $W_{+1j}, W_{-1j}, j = 1, 2$ olarak güncelle.

Çıktı Nihai sınıflandırıcı $F^T(x)$

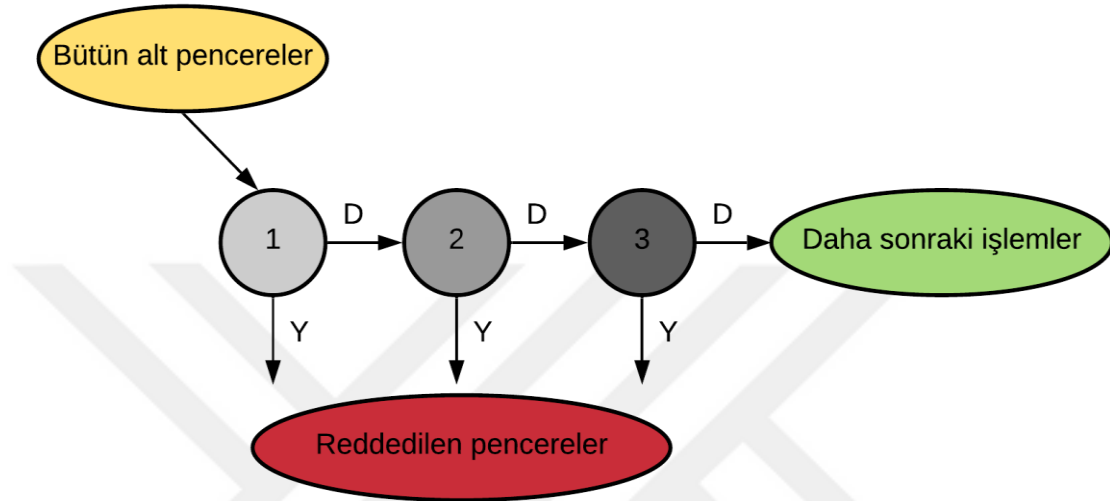
Şekil 2.2. AdaBoost algoritması için temel bir pseudo kod

Dikkat Mekanizmasına Sahip Kademeli Yapı

Dikkat mekanizmasına sahip kademeli yapı, Viola-Jones dedektörünün en önemli bileşenlerinden biridir. Bu yapının ana amacı, daha küçük ve olan küçük yükseltilmiş sınıflandırıcılar kullanarak, negatif alt-pencerelerin çoğunu reddetmek ve pozitif pencerelerin hemen hemen hepsini kabul etmektir. Bu sayede alt pencerelerin birçoğu dedektörün erken aşamalarında reddedilir ve tespit işleminin verimi artar.

Kademeli yapı olarak adlandırılan çatı, aslında bir karar ağacıdır. Şekil 2.3'de gösterildiği üzere, her bir alt-pencere girdisi, tespit esnasında sırası ile birtakım düğümlerden geçer. Her bir düğüm, ilgili alt pencerenin reddedilmesi veya bir sonraki aşamaya gönderilmesi üzerine bir seçim yapar. Genellikle düğümlerde ileriye gidildikçe, her bir düğümdeki zayıf sınıflandırıcı sayısı artar. Örneğin [28]'de, ilk beş düğüm sırası ile 1, 10, 25, 25 ve 50 zayıf sınıflandırıcı içermektedir.

Sezgisel olarak bu mantıklı bir yaklaşımdır, zira her bir düğüm olumlu örnekleri korumaya çalışırken olumsuz örnekleri reddetmeye çalışır ve aşamalar ilerledikçe bu iş daha da zorlaşır. Erken aşamalarda az sayıda sınıflandırıcının bulunması, daha önce de bahsedildiği üzere dedektörün hızını artıran bir faktördür.



Şekil 2.3. Dikkat mekanizmasına sahip kademeli yapının şeması.

Kademeli yapı, eğitim aşaması üzerinde de etkilidir. Yüz tespiti, sunulan her bir pencerede çok düşük olasılıkla yüz bulunmasına istinaden nadir gerçekleşen bir olaydır. Bu nedenle yüksek performanslı bir yüz dedektörünün geliştirilmesi için milyonlar mertebesinde negatif örnek gerekmektedir. Bu çok sayıda negatif örnekle başa çıkmak için, Viola-Jones yüz dedektörü bir önyükleyici kullanmaktadır [28]. Bunun için öncelikle her bir düğümde manuel olarak bir eşik değeri seçilmektedir. Devamında ise kısmi sınıflandırıcı kullanılarak bir sonraki düğümün eğitimi için daha fazla reddedilmemiş negatif örnekler taranmaktadır. Kademeli yapı boyunca her bir düğüm, diğer düğümlerden bağımsız olarak ayrı ayrı eğitilir. Bu tarz bir yaklaşımın seçilmesinin ana sebebi, başarıyı artırmak için eğitim işlemine doğrusal olmayan bir yapı katmaktır. Düğümlerin ayrı ayrı eğitimi, her ne kadar başarıyı artırmak amacıyla gerçekleştirilmiş olsa da yakın zamanda yapılan çalışmalar, farklı düğümlerin eğitim süreçlerinin tamamen ayrılmamasının eğitim sürecine daha faydalı olduğuna dair kanıtlar sunmuşlardır [63].

Kademeli yapının zayıf sınıflandırıcılarının sayısını ve eşik değerlerini belirlemek manuel bir şekilde gerçekleştirilmektedir. Fakat bu değerler dikkatli bir şekilde seçilmelidir. Seçim eşik değerlerinin yüksek olması durumunda dedektör çok hızlı çalışabilmekte fakat genel tespit başarımları düşebilmektedir. Öte yandan seçim eşik değerleri düşük seçildiği zaman, çok sayıda pencerenin

çok sayıda düğümden geçmesi gerekmektedir ve bu da dedektörü yavaşlatmaktadır. 2000’li yılların kısıtlı hesaplama kaynakları göz önüne alındığında, iyi bir yüz dedektörünün eğitimi ve ince ayarları için gereken süren aylar mertebesinde [11].

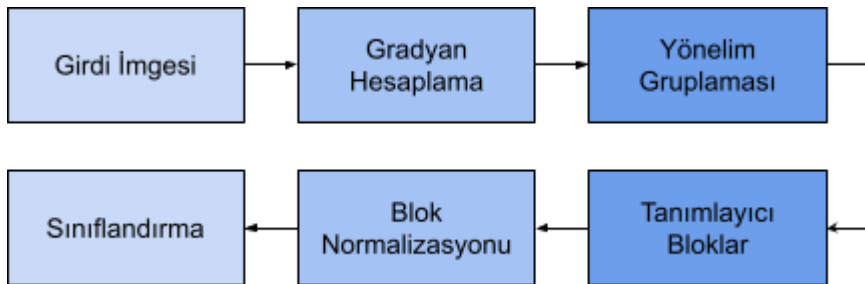
2.1.2. Yönlü Gradyanlar Histogramı

Yönlü Gradyanlar Histogramı, nesne tespiti amacıyla bilgisayarla görü ve görüntü işleme alanlarında kullanılan bir özellik tanımlayıcıdır. Bu yaklaşım, görüntünün belirli bölgelerindeki gradyanların yönlerinin sayısını hesaplayarak gerçekleştirilir.

YGH arkasındaki temel düşünce, bir görüntü içerisindeki nesneye ait görünüm ve şeklin, yoğunluk gradyanları veya kenar yönlerinin dağılımı ile tanımlanabileceğidir. Bunun için öncelikle görüntü, hücre denilen küçük bağlantılı alanlara bölünür. Ardından her bir hücre içerisindeki pikseller için gradyan yönlerinin histogramı hesaplanır. Bu histogramların eklenmesiyle tanımlayıcı oluşturulur. Başarımı artırmak için yerel histogramlar, blok olarak tanımlanan resmin daha büyük alanlarının yoğunluk ölçütleriyle normalleştirilebilir. Bu normalizasyon, düzensiz ışıklandırma ve gölge oluşumuna karşı yöntemin daha gürbüz sonuçlar vermesine olanak sağlar.

YGH tanımlayıcısının diğer tanımlayıcılara karşı en önemli avantajı geometrik ve fotometrik değişikliklere karşı duyarsız olmasıdır. Resmin bölünmüş alt alanı olan hücre seviyesinde çalıştığı için, nesne yönelimi hariç olmak kaydıyla geometrik ve fotometrik değişimlerden etkilenmez. Zira bu tarz değişimler hücre seviyesinde değil, geniş uzamsal bölgelerde gerçekleşir.

YGH’ye dair akış şeması Şekil 2.4’de verilmiştir. YGH algoritmasının gerçekleştirilmesi şu bölümlerden oluşmaktadır:



Şekil 2.4. YGH için akış şeması.

Gradyan Hesaplama

Birçok özellik dedektörünün görüntü ön işlemedeki ilk adımı, renk ve gamma değerlerinin normalleştirilmesidir. YGH algoritmasındaki tanımlayıcı normalizasyonu aynı işlevi gördüğünden dolayı bu adım atlanabilir. Dolayısıyla YGH’nin ilk adımı gradyan değerlerinin hesaplanmasıdır. Bu hesaplama, yatay veya dikey yönlere tek boyutlu ayrık noktalı türev maskesinin uygulanmasıyla

gerçekleştirilir. Spesifik olarak bu yöntem, görüntünün renk veya yoğunluk verilerinin $[-1,0,1]$ ve $[-1,0,1]^T$ filtre çekirdekleri ile filtrelenmesiyle gerçekleştirilir. Dalal ve Triggs [30], 3×3 Sobel veya diyagonal maskeler gibi daha karmaşık maskeleri denemişler, fakat bu maskelerin daha düşük başarımlar sergilediklerini gözlemlemişlerdir. Ayrıca türev maskesini kullanmadan önce görüntü üzerinde Gauss yumuşatmasını denemişler, fakat yine aynı şekilde, görüntü yumuşatmanın da başarımlar üzerinde bir etkisi olmadığını gözlemlemişlerdir.

Yönelim gruplaması

Hesaplamanın ikinci adımı hücre histogramlarının oluşturulmasıdır. Hücre içindeki her piksel, gradyan hesaplamasında bulunan değerlere bağlı olarak, yönelim tabanlı histogram kanalı için bir oy verir. Hücreler daire veya dikdörtgen şeklinde olabilir ve histogram kanalları, 0 ile 180 veya 0 ile 360 derece arasında eşit aralıklara dağıtılır.

Tanımlayıcı bloklar

Aydınlatma ve kontrasttaki değişiklikleri düzenlemek için, gradyan değerlerinin yerel olarak normalleştirilmesi gerekmektedir. Bu da hücrelerin birleştirilerek, daha büyük ve uzamsal olarak bağlantılı blokların oluşturulmasıyla gerçekleştirilir. YGH tanımlayıcısı, tüm blok bölgelerinden elde edilen normalleştirilmiş hücre histogramlarına ait bileşenlerin birleştirilmiş vektörü olarak elde edilir.

Blok Normalizasyonu

Dalal ve Triggs[30], blok normalizasyonu için dört farklı yöntem denemişlerdir. v , verilen bir bloktaki bütün histogramları içeren normalleştirilmemiş vektör; $\|v\|_k$, $k = 1, 2$ için k -normu ve e küçük bir sabit olsun. Bu durumda aşağıdakilerden herhangi birisi normalizasyon faktörü olarak kullanılabilir:

- L2-norm: $f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}$
- L2-hys: [29]'deki gibi L2-norm'un kırıldıktan (maksimum v değerlerini 0.2'de limitleyerek) sonra yeniden normalleştirilmesi.
- L1-norm: $f = \frac{v}{\sqrt{\|v\|_1 + e}}$
- L1-kök: $f = \sqrt{\frac{v}{\|v\|_1 + e}}$

Nesne Tespiti

YGH tanımlayıcıları, nesne tanıma amacıyla makine öğrenmesi algoritmalarına öznitelik olarak verilebilir. Dalal ve Triggs, çalışmalarında DVM’de kullanılmak üzere YGH tanımlayıcılarını öznitelik olarak tanımlamışlardır. YGH öznitelikleri, DVM ile sınırlı kalmamakla birlikte farklı sınıflandırıcılara da öznitelik olarak verilebilir.

2.2. Yüz İfadesi Tanıma

Yüz ifadeleri, insanların duygusal durumlarını ve niyetlerini iletmeleri için en güçlü, doğal ve evrensel sinyallerden biridir [64]. Yüz ifadelerinin evrenselliği, başta psikoloji ve fizyoloji olmak üzere birçok bilim dalında tartışılacak olan bir konudur. Bu alanda, insan duyguları ve bunların yüz ifadeleri üstündeki etkileriyle ilgili yaptığı çalışmalarıyla öncü olan psikolog Paul Ekman’ın çalışmaları ve bulguları kayda değerdir. Ekman, kültürler arası araştırmaların temellendirilebilmesi için dış kirlenmeye çok az maruz kalmış izole bir kültüre sahip olan Papua Yeni Gine yerlileri üzerinde gerçekleştirdiği çalışmalar ile evrensel yüz ifadelerinin varlığı üzerine önemli tespitlerde bulunmuştur [6]. Daha önce yabancılar ile teması geçmemiş ve izole bir ortamda yaşayan bu topluluğa ait bireylere, cevap olarak yüz ifadelerinin sergilenmesi gereken birtakım sorular yöneltilmiştir. Herhangi bir kültürel etkileşime sahip olmayan veya medya vasıtasıyla başka kültürlerin benzer durumlara karşı sergiledikleri yüz ifadelerinin empoze edilmediği bu topluluğa ait bireylerin, sorulan sorulara istinaden sergiledikleri yüz ifadelerinin, dünyanın geri kalan kültür ve alt kültürleriyle aynı olduğu tespit edilmiştir.

Ekman ve Friesen [65], gerçekleştirdikleri kültürler arası çalışmalar vasıtasıyla, insanların kültürlerden bağımsız olarak altı temel duyguyu sergilediklerini belirlemişlerdir. Bu duyguları öfke, tiksinti, korku, mutluluk, üzüntü ve şaşkınlık olarak tanımlamışlardır. Yakın zamanda gerçekleştirilen çalışmalar ise bu altı duygunun kültürler arası özgü olduğunu ve evrensel olmadıklarını öne sürmüşlerdir [66].

Mehrabian [67], genel beğeniye dair mesajların %55’inin yüz ifadesiyle, %7’sinin konuşulan kelimelerle ve geri kalanın da paralinguistik (tonlama ve söyleniş biçimi) bir şekilde aktarıldığını belirtmiştir. Bu bulgular ışığında yüz ifadelerinin, duyguların aktarımında kullanılan en aktif bileşenlerden birisi olduğu sonucu çıkarılabilir.

Dijitale evrilen günümüzde, başta insan bilgisayar etkileşimi olmak üzere, YİT’in kullanıldığı birçok çalışma alanı mevcuttur. Son 40 yılda gerçekleştirilen çalışmaların yavaş temposunun aksine, teknolojinin atılımıyla birlikte son 20 yılda bilgisayar destekli YİT alanında birçok sayıda çalışma gerçekleştirilmiştir.

YİT üzerinde yapılan çalışmaları sınıflandırmak için farklı yaklaşımlar mevcuttur. YİT, veri setinin elde edilme durumuna göre kontrollü veya kontrolsüz, kullanılan girdi türüne göre statik veya

dinamik ve kullanılan yöntem kümesine göre ise geleneksel veya derin öğrenme tabanlı olarak sınıflandırılabilir.

Kontrollü ortamlarda oluşturulan veri tabanları, genellikle laboratuvar ortamlarında sjeler vasıtasıyla elde edilir [68–70]. Bu veri tabanları, doğru ışıklandırmaya, doğru bir şekilde sergilenmiş yüz ifadelerine ve istenilen açılardan çekilmiş imgelere sahiptir. Kontrolsz ortamlarda oluşturulan veri tabanları ise büyük ölçde internet arama motorlarında, söz konusu ifadeler ile yapılan arama sonuçlarının filtrelenmesiyle elde edilir [71–73]. Bu veri tabanları, farklı ışıklandırmalara, net olarak hangi duyguyu sergilediđi belirli olmayan yüz ifadelerine, yüz tanımayı daha karmaşık hale getirecek farklı açılara ve kısmi olarak yüzn belli bir kısmının örtl olduđu imgelere sahip olabilir.

Statik yöntemlerle gerçekleştirilen YİT, sadece bir adet imgenin uzamsal bilgileri kullanılarak gerçekleştirilir. Bu yaklaşımda, yüz ifadesinin tespiti için sadece söz konusu imgeden elde edilen öznitelikler kullanılır. Dinamik yöntemler vasıtasıyla gerçekleştirilen YİT’de ise bir yüz ifade sekansında bulunan ardışık imgeler arasındaki zamansal ilişki dikkate alınır. Yüz ifade sekansı, genellikle yüzn ifadesiz hali ile başlayıp ilgili ifadeyi sergileyen eden yüz imgesi ile biter. Video veya seri çekimler ile elde edilebilecek yüz ifade sekanslarında, sekans içerisindeki imgelerin birbirleri ile arasındaki ilişkiler, farklı yöntemler yardımıyla irdelenir. Görsellik temelli bu iki yöntem kümesine yardımcı olmak amacıyla, yüz ifade başarımının artırılması için ses ve fizyolojik sinyal (EMG, EKG, EEG) verilerinin kullanıldığı yaklaşımlar da mevcuttur [74]. Sadece yüz imgelerine dayalı YİT oldukça başarılı sonuçlar vermesine rağmen, farklı yöntemlerin katkılarıyla oluşturulan çalışmalar, bize daha tanımlayıcı bilgiler vermekle birlikte gerçekleştirilen YİT’i daha grbz kılabilirler [14].

Son olarak, YİT üzerinde yapılagelen çalışmalar, geleneksel ve derin öğrenme tabanlı yöntemler olarak sınıflandırabilir. Geleneksel yöntemler, manuel olarak oluşturulan veya elde edilen öznitelikleri kullanmaktadırlar. Bunun için genellikle ç aşamadan oluşan bir akış şeması kullanmaktadırlar: Yüz tespiti, yüze ait özniteliklerin çıkarılması ve sınıflandırma. İlk adım olan yüz tespiti üzerine yapılan çalışmalara ve uygulanan yöntemlere bir önceki başlıkta değinmiştik. Yüz tespiti aşamasından sonra elde edilen yüz imgelerinden, yüz ile ilgili özniteliklerin çıkarılması gerekmektedir ve bunun için birçok farklı yöntem önerilmiştir. Bu yöntemlere örnek olarak YİÖ [75], ç dikey dzlem üzerinde YİÖ [76] ve NMF [77] verilebilir. Özellikler elde edildikten sonra son aşamada, farklı sınıflandırma yöntemleri kullanılarak ifade sınıflandırma gerçekleştirilir.

Derin öğrenme tabanlı yöntemler, geleneksel yöntemlerden farklı olarak özellik çıkarımını DSA’ların yardımıyla gerçekleştirmektedirler. Bilgisayar donanımlarının etkin olarak kullanıldığı bu yaklaşımda, çıkarılan çok sayıda özellik yardımıyla yüksek başarımlar elde edilmektedir. Derin öğrenme tabanlı yöntemlerin farklılaştıđı bir diđer nokta, öznitelik çıkarımı ve sınıflandırma işlemlerini aynı çatı altında birleştiren bir akış hattına sahip olmasıdır. Bu sayede eğitim ve test

aşamalarının işleyişi hakkında daha fazla bilgiye sahip olunabilmekte ve bu aşamalar esnasında ortaya çıkabilecek sorunlar erkenden tespit edilebilmektedir.

2.2.1. Yüz Hareketi Kodlama Sistemi

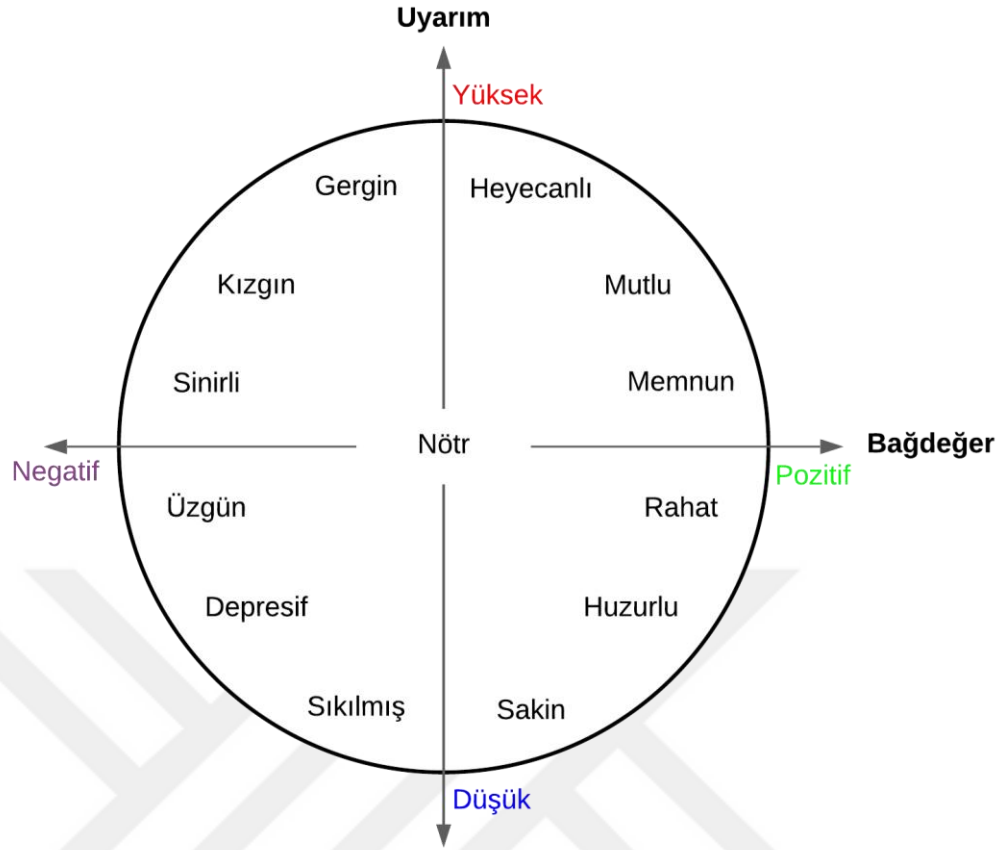
Yüz Hareketi Kodlama Sistemi (Facial Action Coding System – FACS), aslen İsveçli anatomist Carl-Herman Hjortsjö tarafından geliştirilen bir sisteme dayanan ve insan yüzü üzerindeki hareketleri sınıflandırmak amacıyla kullanılan bir sistemdir. Ekman vd. [78] tarafından birçok güncelleme alan son sürümü 2002 yılında yayınlanmıştır. Duyguların fiziksel dışavurumlarını sistematik bir şekilde kategorize eden YHKS, başta psikoloji ve bilgisayar destekli animasyon olmak üzere birçok disiplin tarafından kullanılmaktadır. Uygulamalara örnek olarak depresyon analizi [79] ve kendilerini sözel bir biçimde ifade edemeyen hastaların acı seviyelerinin ölçülmesi [80] verilebilir.

YHKS, yüz üzerinde yüze ait kaslar ile gerçekleşen en küçük değişimleri kodlayan bir sistemdir. Araştırmacılar, YHKS yardımı ile bir yüz ifadesini, olabilecek en küçük Hareket Birimlerine (HB) ayrıştırabilmektedir. YHKS, HB'yi bir veya daha fazla kasın kasılması veya gevşemesi olarak tanımlamaktadır. HB'ler farklı kombinasyonlar ile farklı yüz ifadelerini meydana getirebilmektedir.

YHKS şiddetleri A'dan E'ye kadar olan harfler ile artan bir şekilde ifade edilir. Örneğin HB 1A, ilgili kişinin HB 1'i olabilecek en düşük temsil ile sergilediğini, HB 1E ise olabilecek en içten temsil ile sergilediğini belirtir. Bu sayede, ifadelerin içten veya sahte olup olmadığı anlaşılabilir. Örneğin sahte gülümsemelerde sadece zygomatic major (ağza açı vererek gülmeyi sağlayan kas) kasılırken, içten gülümsemelerde zygomatic major ve orbicularis oculi (göz kapaklarını kontrol eden kas) birlikte kasılır. Gülümseme esnasında YHKS yardımıyla bütün HB'ler incelendiği zaman, yüz ifadesi hakkında net bir bilgiye sahip olunabilir.

2.2.2. Bağdeğer – Uyarım Uzayı

Bağdeğer – Uyarım Uzayı (Valence-Arousal Space), görsel, işitsel ve fizyolojik olarak duygu analizi için evrensel olarak kullanılan ve kabul edilen bir modeldir. Bu uzay üzerinde bağdeğer, duygulanımın olumlu veya olumsuzluğunu temsil ederken, uyarım ise duygulanımın sakinliği veya şiddeti hakkında bilgi verir. Şekil 2.5'te bağdeğer – uyarım uzayı ve farklı duyguların bu uzay üzerindeki konumları verilmiştir. Bu uzayda duygular, x ekseninde negatif değerden pozitif değere, y ekseninde ise düşük uyarımdan yüksek uyarıma doğru olacak şekilde konumlanmıştır. Bu model, duygular arasındaki geçişi uygun bir şekilde temsil etmesinden ötürü literatürde sıklıkla sürekli duygu takibi için kullanılır [81].



Şekil 2.5. Bağdeğer – uyarım uzayı.

2.2.3. Yüz İfadelerinin Kategorilendirilmesi

Yüz ifadeleri, farklı uzmanlar tarafından farklı şekilde kategorilendirilmektedir. Literatür incelendiğinde, sıklıkla şu sınıflandırmalar kullanılır: Temel ifadeler, bileşik ifadeler ve mikro ifadeler.

Paul Ekman [82], altı temel ifadeyi mutluluk, şaşırma, üzüntü, kızgınlık, iğrenme ve korku olarak tanımlamıştır. YİT ile ilgili veri tabanları genellikle bu altı temel ifade ile etiketlenmektedir.

Bileşik ifadeler, iki temel ifadenin birleşiminden oluşmaktadır. Du vd. [83], gerçekleştirdiği çalışmada 7 temel ifade (6 temel ifade ve nötr ifade), 12 bileşik ifade ve 3 adet ek ifade (hayranlık, dehşete düşme ve nefret) olmak üzere 22 farklı ifade tanımlamıştır.

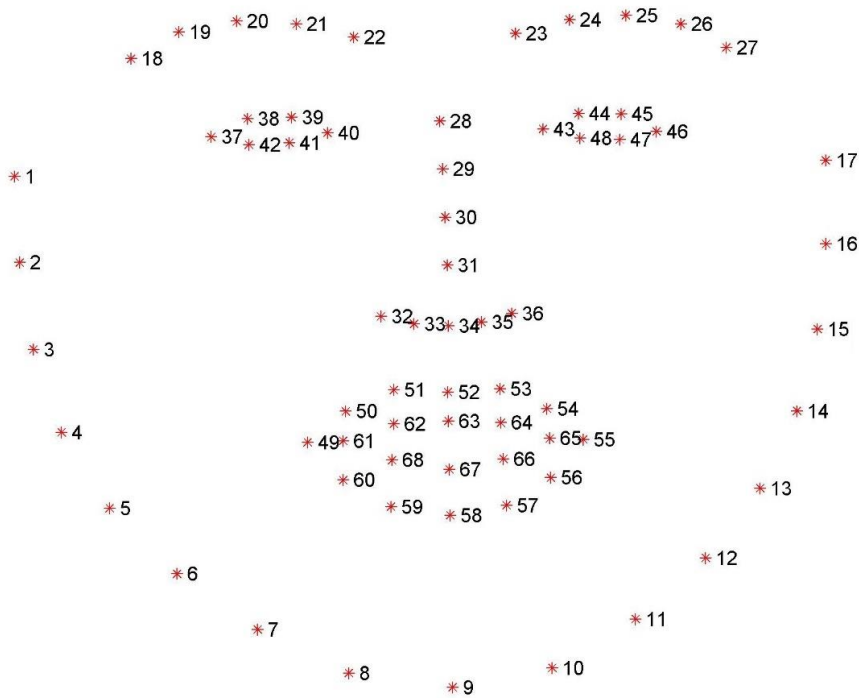
Mikro ifadeler, istemsiz olarak sergilenen ve spontane olarak gelişen küçük yüz ifadelerini temsil ederler [84]. Bu ifadeler, kısıtlı bir süreliğine bir kişinin gerçek ve potansiyel ifadelerini açığa çıkarma eğilimindedirler. Mikro ifadelerin sergilenme süreleri çok kısadır ve genellikle 40 milisaniye ile 200 milisaniye arasında gerçekleşirler. Mikro ifadeler üzerinde yapılan çalışmalar, genellikle psikoloji ve güvenlik sorgulamaları alanlarında gerçekleştirilmektedir.

2.2.4. Yüz Anahtar Noktaları

Yüz anahtar noktaları, yüz çevresini ve yüze ait öğeleri kapsayan noktalar kümesine verilen isimdir. Bu noktalar, bir yüz bileşeninin (göz kenarı) kesin konumunu gösterebileceği gibi, farklı bileşenleri birleştiren interpolasyonlu bir noktalar kümesi de olabilir. Yüz anahtar noktaları, başta yüz tespiti ve YİT olmak üzere, baş eğimi ve oryantasyonunun ayarlanması, göz merkezlerinin tespiti ve özellik çıkarımı için sıklıkla kullanılmaktadırlar [85]. Şekil 2.6’da 68 adet yüz anahtar noktasına sahip olan bir şablon verilmiştir [86].

Yüz anahtar noktalarını tespit için kullanılan algoritmaları, holistik yöntemler, Kısıtlı Yerel Model (KYM) yöntemleri ve regresyon tabanlı yöntemler olarak üç sınıf altında özetlenebilir [85]. Bu yöntemler, yüz görünümünü ve şekil bilgilerini kullanım şekilleriyle farklılaşmaktadırlar.

Holistik yöntemler, yüz anahtar noktalarının tespiti için global yüz görünümü ve şekil bilgisini doğrudan kullanarak model oluştururlar. AGM, bu yöntem kümesinin en iyi bilinen örneğidir [87]. KYM yöntemleri, global yüz modeli ile birlikte yüzün farklı bölümlerinin de görünümünü kullanırlar [88]. Bu sayede, holistik yaklaşıma oranla aydınlatma ve yüzün kısmi olarak kapalı olduğu durumlarda daha iyi başarımlar gösterirler. Regresyon tabanlı yöntemler ise anahtar noktalarını doğrudan imge görünümüne eşleştirmeye çalışırlar. Holistik ve KYM tabanlı yöntemlerden farklı olarak, regresyon tabanlı yöntemler global bir yüz modeli oluşturmazlar. Bu yöntem kümesi altındaki çalışmalar, yaklaşımlarına göre doğrudan regresyon, kaskad regresyon ve derin öğrenme tabanlı regresyon yaklaşımları olarak üçe ayrılırlar [85].



Şekil 2.6. 68 adet yüz anahtar noktasına sahip olan bir şablon

3. DERİN ÖĞRENME

Bu bölümde son yılların en popüler konularından biri olan derin öğrenmeden bahsedilecektir. Bu kapsamda öncelikle derin öğrenmenin kısa bir tarihinden bahsedilecektir. Devamında ise bilgisayarla görü alanında çığır açan ESA'lara yer verilmiştir. Son olarak çalışma kapsamında yüz sentezi için kullanılan ÇÜA'lara ilişkin genel birtakım bilgilerin verilmesi uygun görülmüştür.

3.1. Derin Öğrenme

Derin öğrenme, hiyerarşik bir şekilde çok sayıda gizli katman içeren yapay sinir ağları ve benzeri makine öğrenme algoritmalarını kapsayan çalışma alanıdır. Buradaki “derin” terimi, sunulan modellerin içerdikleri fazla sayıda katman ve parametrelere verilen bir atıftır.

Derin öğrenme, geçmişi 1940'lara dayanan uzun ve zengin bir tarihe sahiptir. 1940 ve 1960'lı yıllarda sibernetik, 1980 ve 1990'lı yıllarda yapay sinir ağları olarak adlandırılan uygulama alanı, 2006 yılından itibaren derin öğrenme adıyla anılmıştır [89]. Uygulamaya yönelik çalışmaların zaman içerisindeki değişkenliğine istinaden, popülerliği dönemden döneme farklılıklar göstermiştir. İnternetin ve arama motorlarının yaygınlaşmasıyla birlikte eğitim verilerinin miktarının artmasına paralel olarak derin öğrenme, günümüzde daha uygulanabilir bir hale gelmiştir. Buna paralel bir şekilde derin öğrenmenin başarımı, üzerinde çalıştığı bilgisayar yazılımları ve donanımlarının gelişmesiyle birlikte artış göstermiştir. Derin öğrenme sayesinde nesne tanıma, nesne tespiti, doğal dil işleme, konuşma ve ses tespiti gibi birçok alanda çığır açacak seviyede başarımlar elde edilmiştir.

Makine öğrenmesi algoritmalarının başarımları, büyük ölçüde girdi verilerinin temsilinin düzgün olmasına bağlıdır. Kötü bir veri temsili, genellikle iyi bir veri temsiline kıyasla daha düşük performans sergiler. Bu sebeple, ham verilerden özellikler oluşturmaya odaklanan özellik mühendisliği, uzun süredir makine öğrenmesi alanında önemli bir çalışma konusu olmuştur. Özellik mühendisliği genel olarak alana özgüdür ve otomatize edilemediği için uygulanması esnasında insan etkileşimi gerektirmektedir. Bilgisayarla görü alanında farklı uygulamalarda kullanılan YGH, SIFT ve BoW gibi farklı özellikler mevcuttur. Ortaya konulan yeni bir özellik, kayda değer bir başarımla sergilediği takdirde, uzun yıllar boyunca trend olarak kalmakta ve kullanılmaktadır [90].

Derin öğrenme algoritmaları, özellik mühendisliği yaklaşımlarından farklı olarak, özellik çıkarımını otomatik bir şekilde gerçekleştirirler. Bu da araştırmacıların minimum alan bilgisi ve insan çabasıyla ayırt edici özellikleri çıkarmasına olanak tanır [91]. Bu algoritmalarda veri temsili, katmanlı bir mimariye gerçekleştirilir. İlk katmanlarda düşük seviye özellikler elde edilirken, son katmanlara doğru gidildikçe yüksek seviye özellikler elde edilmektedir.

Derin öğrenme mimarilerine, derin sinir ağları, derin inanç ağları, derin otokodlayıcılar, tekrarlayan sinir ağları, evrişimsel sinir ağları ve çekişmeli üretici ağlar örnek olarak verilebilir. Bu bölüm altında, tez çalışması kapsamında kullanılan ESA'lara ve ÇÜA'lara değinilecektir.

3.2. Evrişimsel Sinir Ağları

Özellikle son yıllarda görsel bilginin analiz edilmesinde sıklıkla kullanılan ESA'lar, çok katmanlı mimari ile özellik çıkaran bir sinir ağı yapısıdır. Günümüzde kullanılan modern ESA yaklaşımının ilk örneği, Yann LeCun tarafından 1990'da sunulmuştur [92]. İlk örneklerinden günümüze ESA'lar üzerinde birçok çalışma gerçekleştirilmiştir. Fakat ihtiyaç duyduğu işlem yükü ve zamanın bilgi işlem araçlarının bu konudaki yetersizliğinden dolayı yeterli ilgiyi görememiştir. Bu sebeple ESA'lar üzerinde gerçekleştirilen çalışmalar, gerçek dünya koşullarından uzak kısıtlı veri tabanları ile sınırlı kalmıştır. ILSVRC-2012 yarışmasında, GPGPU'ların paralel işleme kabiliyetini öne çıkaran Alex Krizhevsky'nin önerdiği AlexNET [45] ile tarihi bir başarımla elde eden ESA'lar haklı bir üne kavuşmuşlardır. Bu aşamadan sonra birçok farklı araştırmacı ve araştırma grubu tarafından ESA'lar üzerinde gerçekleştirilen çalışmalar ile bu alana bir çok farklı yaklaşım ve yorumlar getirilmiştir [93].

ESA'lar, verileri işlemek için ızgara tipi çok katmanlı bir topolojiye sahip olan özel bir sinir ağı türüdür. Bir ESA, temel olarak verilen girdiyi farklı katmanlardan geçirerek bir öznitelik çıktısı üretir. Bu girdi, değişkenlerden oluşan tek boyutlu bir zaman serisi, piksellerden oluşan 2 boyutlu bir imge veya voksellerden oluşan 3 boyutlu bir hacim olabilir. ESA'larda, "evrişim" adı verilen matematiksel bir işlem kullanılır. Evrişim, özel bir tür lineer işlemdir. Evrişimsel ağlar, katmanlarından en az birinde genel matris çarpımı yerine evrişim işlemini kullanan basit sinir ağlarıdır [94].

3.2.1. Katmanlar

Bir ESA, genel olarak dört katmandan oluşur: Evrişim katmanı, aktivasyon katmanı, havuzlama katmanı ve tam bağlı katman. Bu katmanlara ek olarak sınıflandırma işleminde yardımcı olmak amacıyla Softmax katmanı da sıklıkla kullanılabilir.

Evrişim Katmanı

Evrişim katmanı, ESA'lara adını veren ve temel yapı taşı olan katmandır. Özellik çıkarımı gerçekleştirilen bu katmanda, bir dizi filtre yardımıyla özellik haritaları hesaplanmaktadır. Lineer bir işlem olan evrişim işlemi, bir girdiye tekrar eden bir şekilde bir filtre uygulanmasıyla gerçekleştirilir. Bu işlem sonucunda özellik haritası denilen bir aktivasyon haritası elde edilir.

İki boyutlu bir imge üzerinde evrişim işlemini gerçekleştirmek için bir takım imge ve filtre parametreleri hakkında ön bilgi sahibi olmak gerekmektedir. Bu bilgiler şu şekilde sıralanabilir: İmgenin eni ve boyu, kullanılan filtrenin eni ve boyu, kullanılan filtrenin kaydırma (stride) miktarı, imgeye uygulanan kenar ekleme (padding) miktarı.

İmgenin eni ve boyu, x ve y boyutlarında imgenin kapladığı piksel alanının boyut bilgisidir. Yine aynı şekilde kullanılan filtrenin eni ve boyu, filtrenin x ve y boyutlarında kapladığı piksel alanının boyut bilgisi şeklindedir. Evrişim işlemi, imge üzerindeki x_0, y_0 koordinatlarından başlanarak, filtrenin imgeyi kapladığı alanla yapılan skaler çarpım ile gerçekleştirilir. Her bir evrişim işlemi sonrasında filtre, kaydırma miktarınca (stride) kaydırılır ve tekrar skaler çarpım gerçekleştirilir. Her kaydırma öncesinde elde edilen sonuçlar, oluşturulan özellik haritasında ilgili konuma yazılır. Bunun haricinde, evrişim işlemi sonucunda ortaya çıkan özellik haritasının boyutunu ayarlamak için girdi imgesinin kenarlarına “sıfır” değerinde pikseller eklenebilir. Zira evrişim işlemi sonucunda ortaya çıkan özellik haritasının boyutu girdi imgesinden küçüktür. Bu işleme kenar ekleme (padding) adı verilir.

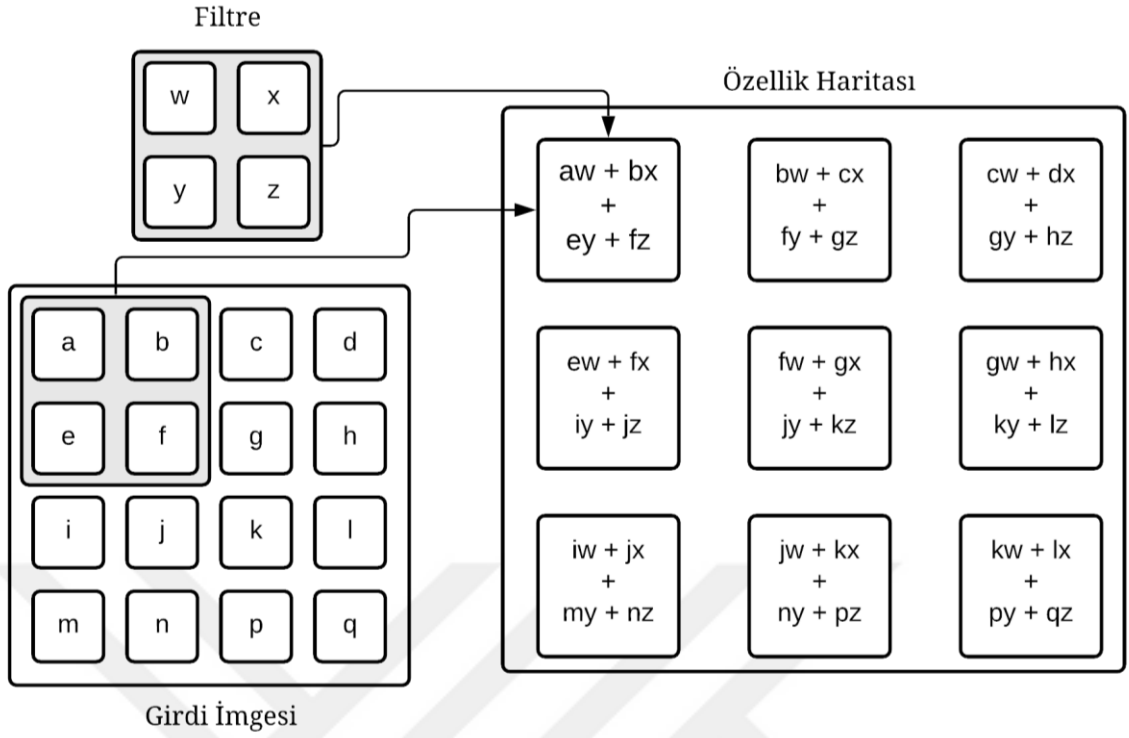
Verilen bu bilgiler ışığında n_h ve n_w girdi imgesinin boyu ve eni, k_h ve k_w kullanılan filtrenin boyu ve eni, p_h ve p_w dikeyde ve yatayda eklenen kenar pikseli sayısı ve s_h ve s_w dikeyde ve yatayda kaydırılan piksel sayısı olsun. Bu durumda elde edilecek olan özellik haritasının boyutu $F(x, y)$, denklem 3.1’de verildiği şekilde hesaplanır:

$$F(x, y) = [(n_h - k_h + p_h + s_h)/s_h] \times [(n_w - k_w + p_w + s_w)/s_w] \quad (3.1)$$

Şekil 3.1’de 4x4’lük bir giriş matrisinin 2x2’lik bir filtre ile evrişimi gösterilmiştir. Bu işlem sonucunda 3x3’lük bir özellik haritası elde edilmiştir.

Havuzlama Katmanı

ESA’lardaki amaç, girdi olarak verilen bir imgeyi ağ boyunca işleyerek bu imgeyi temsil eden bir özellik vektörü elde etmektir. Yüksek boyutlu bir imgeyi daha düşük boyutlu özellik vektörüne dönüştürmek için birtakım bilgilerin ve özelliklerin elenmesi gerekmektedir. Evrişim katmanı, bilgidan özellikleri çıkarır fakat boyut indirgemedi yeterli desteği sağlamaz. Bu nedenle ESA’larda, boyut indirgeme için havuzlama katmanı kullanılır. Havuzlama katmanı sayesinde, evrişim katmanı sonucunda elde edilen özellik haritalarının boyutlarını küçültülür. Böylece ağ yapısı boyunca, öğrenilen parametre sayısı ve yapılan işlem miktarının azaltılması hedeflenmektedir.



Şekil 3.1. 4x4'lük bir girdi imgesi üzerinde, 2x2'lik bir filtre ve 1x1'lik kaydırma ile gerçekleştirilen evrişim işlemi.

Havuzlama işlemi, özellik haritası üzerinde, evrişim işlemine benzer bir şekilde pencere kaydırılması ile gerçekleştirilir. Her bir pencerenin altında kalan alan, seçilen havuzlama türüne göre bir değer döndürür. Daha sonrasında bu değer, yeni oluşturulan özellik haritasının ilgili kısmına yazılır. Havuzlama işlemi için seçilen pencere ve kaydırma parametresi boyutuna göre, çıktı olarak oluşturulan özellik haritasının boyutu değişmektedir. Seçilen pencere boyutu ve kaydırma parametresi, havuzlama sonucu ortaya çıkacak özellik haritasının boyutu ile ters orantılıdır. Pencere boyutu ve kaydırma parametresi arttıkça, özellik haritasının boyutu azalır, tersi durumda ise özellik haritasının boyutu artar. Havuzlama için genellikle pencere boyutu ve kaydırma parametresi aynı seçilir. 2x2 filtre boyutu ile 2x2 kaydırma parametresi literatürde sıklıkla kullanılmaktadır. Bu şekilde gerçekleştirilen işlem sonucunda özellik haritasının boyutu yarıya indirgenir. Bunun haricinde komşu piksellerdeki baskın değerlerin daha güçlü temsil edilmesi için örtüşmeli havuzlama olarak da adlandırılan 3x3 filtre boyutu ile 2x2 kaydırma parametresi de kullanılabilir.

$W \times H$, iki boyutlu bir imgenin eni ve boyu olsun. Pencere boyu P ve kaydırma parametresi S olarak seçildiği takdirde, oluşacak olan $F(x, y)$ özellik haritasının boyutu Denklem 3.2'deki gibi hesaplanır:

$$F(x, y) = ((W - P)/S + 1), ((H - P)/S + 1) \quad (3.2)$$

Havuzlama için genellikle üç farklı yaklaşım mevcuttur. Bunlar, maksimum havuzlama, ortalama havuzlama ve global havuzlamadır.

Maksimum havuzlama, bir pencere altındaki maksimum değeri seçen bir havuzlama işlemidir. Bu işlem neticesinde oluşturulan özellik haritası, girdi olarak verilen özellik haritasının en belirgin özelliklerini içeren bir özellik haritasıdır.

Ortalama havuzlama, bir pencere altındaki değerlerin aritmetik ortalamasını seçen bir havuzlama işlemidir. Geçmişte sıklıkla kullanılan ortalama havuzlama, pratikte maksimum havuzlamanın daha iyi başarımlar gösterdiğini ortaya koyan çalışmalar nedeniyle fazla kullanılmamaktadır.

Global havuzlama, özellik haritasındaki her bir kanalı tek bir değere indirgeyen bir havuzlama işlemidir. ESA'larda sıklıkla kullanılan bir yaklaşım, ilk katmanlardan sonlara doğru gidildikçe özellik haritalarının boyutlarını küçültmek ve kanal sayısını artırmaktadır. Bu sayede ilk katmanlarda, girdiye ait az sayıda genel özellik elde edilirken, son katmanlarda daha çok sayıda spesifik özellik elde edilmektedir. Son katmandaki özellik haritalarının, tam bağlı katmana bağlanması esnasında tek boyutlu bir özellik vektörünün indirgenmeleri global havuzlama ile sağlanır. Global havuzlama, her bir kanal için maksimum değerin seçilmesi veya her bir kanaldaki değerlerin aritmetik ortalamasının seçilmesi olmak üzere global maksimum havuzlama ve global ortalama havuzlama olarak iki şekilde gerçekleştirilebilir. Global havuzlama sonucunda $h \times w \times c$ boyutundaki bir özellik haritası $1 \times 1 \times c$ boyutuna indirgenir.

Şekil 3.2'de bir özellik haritasına uygulanan maksimum, ortalama ve global havuzlama işlemleri ve bu işlemler sonucunda elde edilen özellik haritalarına dair görseller verilmiştir. Havuzlama işlemi sonucunda özellik haritasının boyutu küçülür.

Aktivasyon Katmanı:

Transfer fonksiyonu olarak da bilinen aktivasyon fonksiyonu, her bir özellik haritası ve ağırlık çarpımına uygulanan lineer olmayan bir fonksiyondur. Bu fonksiyon sayesinde hesaplanan değer bir sonraki katmana gönderilir.

Aktivasyon fonksiyonları, ağ yapılarının veri içerisindeki karmaşık örüntüleri öğrenmesine, lineer olmayan yapıları sayesinde katkı sağlar. Ayrıca çıktı değerlerini belirli bir aralıkta tutarak, gradyanların çok yüksek değerlere ulaşması anlamına gelen gradyan patlamasının önüne geçerler. ESA'larda sıklıkla kullanılan aktivasyon fonksiyonlarını şu şekilde sıralayabiliriz: Sigmoid, tanh, ReLU, Leaky ReLU ve Parametric ReLU.

Sigmoid fonksiyonu, yapay sinir ağlarında en sık kullanılan aktivasyon fonksiyonlarından birisidir [95]. x 'in aktivasyon fonksiyonuna girdi olduğu durumda, sigmoid fonksiyonu $f(x)$, denklem 3.3'deki gibi hesaplanır.

a. Özellik Haritası
2 x 2 pencere boyutu, 2 x 2 kaydırma

3	4	6	0
9	2	5	7
3	6	1	8
2	2	0	6

9	7
6	8

b. Maksimum Havuzlama

4,5	4,5
3,25	3,75

c. Ortalama Havuzlama

9

d. Global Maksimum Havuzlama

4

e. Global Ortalama Havuzlama

Şekil 3.2. Farklı şekillerde uygulanan havuzlama işlemi sonucunda elde edilen özellik haritaları

$$f(x) = \frac{1}{1+e^{-x}} \quad (3.3)$$

Sigmoid fonksiyonu, (0,1) aralığında sınırlı, sürekli ve türevlenebilirdir. Fonksiyon, üssel bir terim içerdiğinden dolayı yüksek hesaplama maliyetine sahiptir. Fakat türevi, hesaplama bakımından maliyetli değildir ve $f'(x) = f(x)(1 - f(x))$ formülüyle hesaplanabilir.

Sigmoid fonksiyonunun temelde birkaç dezavantajı vardır. (0,1) aralığında sınırlı olduğundan ötürü çıktı olarak her zaman negatif olmayan değerler üretmektedir. Bu yüzden sıfır merkezli bir aktivasyon fonksiyonu değildir. Ayrıca büyük bir aralıktaki girdi değerlerini (0,1) aralığına dönüştürdüğünden dolayı, çok büyük girdiler bile çıktı değerlerinde sadece küçük değişiklikler yapabilmektedir. Bunun sonucu olarak gradyanları da küçük olmaktadır. Küçük gradyanlar ise kaybolan gradyanlar sorununa yol açabilmektedir.

Hiperbolik tanjant ya da daha çok bilinen adıyla tanh fonksiyonu, çok katmanlı sinir ağlarında sigmoid fonksiyonuna göre daha iyi eğitim başarımı sağlamaktadır. x 'in aktivasyon fonksiyonuna girdi olduğu durumda, tanh fonksiyonu $f(x)$, denklem 3.4'deki gibi hesaplanır.

$$f(x) = \frac{1-e^{-x}}{1+e^{-x}} \quad (3.4)$$

Tanh fonksiyonu, $\tanh(x) = 2\text{sigmoid}(2x) - 1$ olduğundan ötürü sigmoid fonksiyonunun modifiye edilmiş bir şekli olarak kabul edilebilir. Bu yüzden tanh fonksiyonu, sigmoid fonksiyonu gibi sürekli, sınırlı ve türevlenebilirdir. Farklı olarak sınırı (-1,1) aralığındadır. Böylece negatif, pozitif ve sıfır çıktılar üretebilmektedir. Sıfır merkezli bir aktivasyon fonksiyonu olduğundan dolayı, geri yayılım sürecine katkıda bulunmaktadır [95].

Açılımı Rectified Linear Unit olan ReLU fonksiyonu, Nair ve Hinton tarafından 2010 yılında önerilmiştir [96]. Ortaya çıktığı günden bu yana sergilediği başarımından ötürü derin öğrenme uygulamalarında en çok kullanılan aktivasyon fonksiyonlarından birisidir. ReLU'nun yüksek başarımının altında yatan en önemli etken, hızlı bir şekilde öğrenmesidir [97]. ReLU, sigmoid ve tanh aktivasyon fonksiyonlarına kıyasla, derin öğrenmede daha iyi başarımlar ve genelleştirilebilirlik sunmaktadır. Ayrıca bu iki fonksiyona kıyasla işlem yükü daha azdır.

ReLU fonksiyonu, girdinin sıfırdan düşük olduğu durumlarda girdiyi sıfır olarak eşiklemektedir. Girdinin sıfırdan büyük olduğu durumlarda ise çıktı girdiye eşittir. ReLU fonksiyonu denklem 3.5'deki gibi hesaplanmaktadır.

$$f(x) = \max(0, x) = \begin{cases} x_i, & \text{eğer } x_i \geq 0 \\ 0, & \text{eğer } x_i < 0 \end{cases} \quad (3.5)$$

ReLU fonksiyonu, sıfırdan daha küçük girdileri sıfır olarak düzeltir ve bu şekilde, diğer aktivasyon fonksiyonlarında görülen kaybolan gradyanlar sorununa çözüm getirmektedir. Hesaplama ReLU kullanmanın bir diğer avantajı, üssel ve bölmeli işlemler gerçekleştirmediğinden dolayı işlem hızının çok yüksek olmasıdır [98].

ReLU'nun eğitim esnasında hassas olması ve bu nedenle bazı gradyanların ölmesi, en büyük dezavantajlarından biridir. Bu da bazı nöronların ölmesi ve ağırlık değerlerinin güncellenmemesi anlamına gelmektedir[89]. Bu ölü nöron sorununu çözmek için leaky ReLU önerilmiştir.

2013 yılında Maas vd. tarafından önerilen Leaky ReLU, geri yayılım sürecinde ağırlık güncellemelerinin devam etmesi için ReLU'ya küçük bir negatif eğim önermektedir. ReLU'nun ölü nöron problemine çözüm olarak getirilen alpha parametresi yardımıyla eğitim esnasında gradyanlar hiçbir zaman sıfır olmamaktadır. LReLU, sıfırdan küçük girdiler için çıktıyı $\alpha = 0.01$ gibi küçük bir değerle çarpılmaktadır. Sıfır ve sıfırdan büyük değerler için ise ReLU'da olduğu gibi çıktı girdiye eşittir. LReLU fonksiyonu denklem 3.6'daki gibi hesaplanmaktadır.

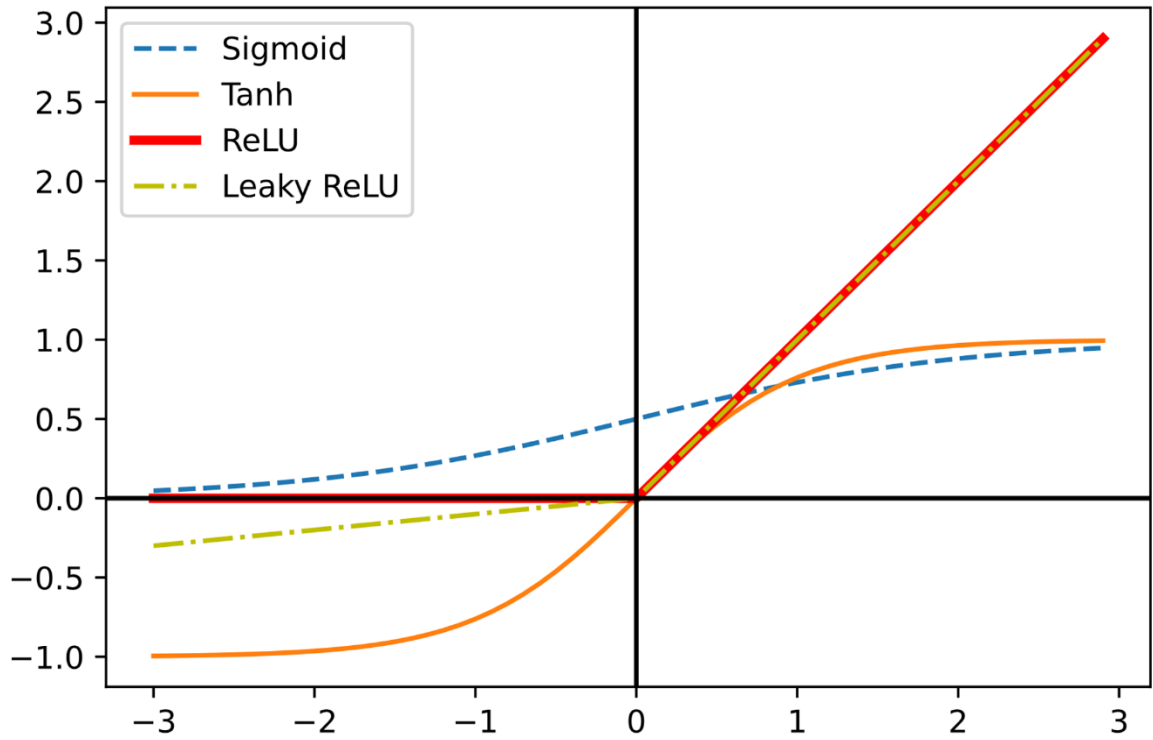
$$f(x) = \begin{cases} x_i, & \text{eğer } x_i > 0 \\ \alpha x, & \text{eğer } x_i \leq 0 \end{cases} \quad (3.6)$$

Parametrik ReLU (PReLU) fonksiyonu, He vd. tarafından 2015 yılında önerilmiştir [99]. PReLU fonksiyonu Denklem 3.7'deki gibi hesaplanmaktadır.

$$f(x) = \begin{cases} x, & \text{eğer } x > 0 \\ ax, & \text{eğer } x \leq 0 \end{cases} \quad (3.7)$$

Burada a öğrenilebilir bir parametre, x ise aktivasyon fonksiyonunun girdisidir. PReLU fonksiyonu, $a = 0.01$ olduğu durumlarda LReLU ve $a = 0$ olduğu durumlarda ise ReLU olmaktadır.

Sigmoid, Tanh, ReLU ve Leaky ReLU fonksiyonlarının grafikleri Şekil 3.3'te verilmiştir. Parametrik ReLU genelleştirilmiş ve değişken bir fonksiyon olduğundan dolayı çizime dahil edilmemiştir.



Şekil 3.3. Sigmoid, Tanh, ReLU ve Leaky ReLU fonksiyonlarının grafik üzerinde gösterimi.

3.2.2. Kayıp Fonksiyonları

Optimizasyon problemlerinde, elde edilen bir çözümü değerlendirmek için kullanılan fonksiyona hedef fonksiyonu denir. Problemin türüne göre hedef fonksiyonun minimize veya maksimize edilmesi gerekmektedir. YSA'lar da kullanılan hedef fonksiyonları, tahmin edilen sınıf ile gerçek sınıf arasındaki farkı en aza indirecek şekilde optimize edilirler. Bu nedenle kullanılan hedef fonksiyonlarına, kayıp fonksiyonu, hata fonksiyonu veya maliyet fonksiyonu da denilebilmektedir. İleri beslemeli sinir ağlarının bir üyesi olan ESA'ların eğitim süreçlerinde geri yayılım algoritmasından faydalanılır. Geri yayılım sürecinde, kayıp fonksiyonunun gradyanları yardımıyla ağırlıklar güncellenir. Eğitim devam ettikçe güncellenen bu değerler yardımıyla kayıp

fonksiyonunun minimize edilmesi planlanmaktadır. ESA'larda sıklıkla kullanılan hata fonksiyonları aşağıda verilmiştir.

Ortalama Kare Hata

Ortalama kare hata, tahmin edilen ve gerçek sınıfın farklarının karelerinin ortalaması olarak hesaplanır. Sonucu, tahmin edilen ve gerçek sınıfın pozitif veya negatif olmasına bağlı olmaksızın pozitifdir. n adet örnek ile yapılan tahmin işlemi sonucunda Y gerçek sınıfa ait etiket vektörü $\hat{Y}$ tahmin edilen sınıfa ait etiket vektörü olmak üzere ortalama kare hata, denklem 3.8'deki gibi hesaplanır.

$$OKH = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (3.8)$$

Çapraz Entropi Hata

Çapraz entropi hata, iki etiketli sınıflandırma işlemleri için kullanılmakta ve 0 ile 1 aralığında bir olasılık değeri vermektedir. Tahmin edilen her örnek için sınıf değeri, gerçek sınıf değeri ile karşılaştırılmaktadır ve aralarındaki mesafeye oranla hata belirlenmektedir. Hata belirleme işlemi, küçük farklar için küçük, büyük farklar için ise çok büyük olacak şekilde logaritmik bir şekilde hesaplanmaktadır. y , gerçek sınıf ve p , tahmin edilen sınıf olmak üzere çapraz entropi hata Denklem 3.9'daki gibi hesaplanır.

$$\text{ÇEH} = -(y \log(p) + (1 - y) \log(1 - p)) \quad (3.9)$$

Kategorik Çapraz Entropi Hata

Çapraz entropi hatanın, ikiden fazla etiketli sınıflandırma için genelleştirilmiş şeklidir. M , sınıf sayısı, y 'nin, c sınıf etiketinin o gözlemi için doğru sınıflandırma olduğuna dair ikili bir indikatör (0, 1) ve p 'nin, c sınıf etiketi için o gözleminin tahmin edilen olasılık değeri olmak üzere kategorik çapraz entropi hata, Denklem 3.10'daki gibi hesaplanır.

$$K\text{ÇEH} = - \sum_{c=1}^M y_{o,c} \log(p_{o,c}) \quad (3.10)$$

3.2.3. Eğitim Sürecinde Kullanılan Parametreler

İleri beslemeli sinir ağlarının eğitim süreçleri iki aşamadan oluşmaktadır. Bunlardan ilki girdi verisinin ağ boyunca ilerlemesidir. İkinci aşama ise hesaplanan kayıp fonksiyonu çıktısı yardımıyla geri yayılım algoritması ile ağırlıkların yeniden hesaplanmasıdır.

Bu eğitim süreci, birtakım parametreler yardımıyla gerçekleşmektedir. Bu parametreleri şu şekilde sıralayabiliriz: Optimizasyon yöntemi, öğrenme oranı, yığın boyutu ve döngü sayısı.

Optimizasyon yöntemi

Optimizasyon algoritmaları, model parametrelerini güncelleyerek kayıp fonksiyonunun değerini en aza indirmeye olanak sağlayan araçlardır. Derin öğrenme için optimizasyon algoritmasının seçimi önemlidir. Çok fazla veriye sahip derin modellerin eğitimi günler veya haftalar sürebilmektedir. Bu tarz bir senaryoda modelin eğitiminin etkinliğini artıracak optimizasyon algoritmasının seçimi büyük önem kazanmaktadır. Buna ek olarak seçilen optimizasyon algoritmasının hiper parametrelerini probleme en uygun şekilde uyarlayarak modelin performansı daha da artırılabilir. ESA ile gerçekleştirilen çalışmalarda en sık kullanılan optimizasyon algoritmaları şunlardır: SGD, Adam, Adagrad, Adadelta, RMSprop.

Öğrenme Oranı

Optimizasyon algoritmalarının, geri yayılım esnasında model ağırlıklarını ne oranda değiştirecekleri öğrenme oranı ile belirlenir. Öğrenme oranı 0 ile 1 arasında değer alan bir hiper parametredir ve modelin yakınsama ve uzaksaması ile yakından ilişkilidir. Düşük öğrenme oranı, her bir döngüde model ağırlıklarında küçük değişiklikler yapar ve model yavaş bir şekilde yakınsar. Eğitim için uzun süre gerektiren bu yaklaşım, yerel minimumlara takılabilir. Öte yandan yüksek öğrenme oranı, model ağırlıklarında büyük değişiklikler yapar. Bu yaklaşımda ise öğrenme, lokal minimumların üzerinden atlayabilir ve uzaksayabilir.

Öğrenme oranı, bir optimizasyon yöntemi için seçilecek en önemli parametredir. Genellikle 0.1 ve 0.01 arası bir öğrenme oranı, birçok model için iyi bir başlangıç noktasıdır. Öğrenme oranını eğitimin başında yüksek tutmak ve ilerledikçe aşağılara çekmek sıklıkla uygulanan bir yöntemdir. Bu yaklaşım ile öncelikle lokal minimumların yerleri tespit edilmekte, ardından küçük adımlar ile lokal minimuma yakınsanmaktadır. Öğrenme oranını dinamik olarak eğitimin seyrine göre ayarlamak için genellikle polinomsal azalma ve üssel azalma şemaları kullanılmaktadır. Bunun yanı sıra öğrenme oranını sabit tutan yaklaşımlar da olmakla beraber başarımları istenilen düzeyde değildir.

Yığın Boyutu ve Döngü Sayısı

Bir ESA'nın eğitiminin ileri besleme ve geri yayılım olmak üzere iki aşamadan oluştuğunu belirtmiştik. İleri besleme esnasında girdiler ağ üzerinde ilerlemektedirler. Geri yayılım esnasında ise zincir kuralı yardımıyla hata fonksiyonunun gradyanı boyunca ilerlenerek model ağırlıkları güncellenmektedir. Günümüz bilgisayar donanımı göz önüne alındığında, bu işlemin farklı girdiler için paralel gerçekleştirmek mümkündür. Girdilerin yığın şeklinde alınarak işlenmesi aynı zamanda öğrenmenin yakınsamasını da hızlandırmaktadır.

Yığın boyutu miktarına göre öğrenme algoritması üç şekilde isimlendirilir: Stokastik gradyan iniş (yığın boyutu = 1), mini-yığın gradyan iniş ($1 < \text{yığın boyutu} < \text{veri seti boyutu}$) ve yığın gradyan

iniş (yığın boyutu = veri seti). Stokastik gradyan iniş, her bir iterasyonda bir adet veriyi işler ve bu nedenle gürültülü bir optimizasyon sağlar. Her bir girdi sonrasında optimizasyon gerçekleştiği için çok fazla sayıda parametre güncellenir ve bu da işlem yükü oluşturur. Mini-yığın gradyan iniş, yığın boyutu ile seçilen belli bir sayıda girdinin aynı anda ağıdan geçirilmesi, hatanın hesaplanması ve ağırlıkların güncellenmesi ile gerçekleştirilir. Stokastik gradyan inişin aksine, daha az gürültülü bir optimizasyon sağlar ve daha az sayıda parametre güncellemesi ile daha az işlem yükü oluşturur. Son olarak yığın gradyan iniş, bütün veriyi eş zamanlı olarak ağıdan geçirir ve geri yayılım ile parametre değerlerini günceller. Her ne kadar diğerlerine oranla daha avantajlı olsa da çok büyük veri setlerinin, kullanılan donanımın kapasitesini aşması nedeniyle kullanımı oldukça kısıtlıdır. Genel olarak ESA üzerine gerçekleştirilen çalışmalarda, kullanılan donanımın bellek kapasitesini dolduracak şekilde yığın boyutu 32'nin katları (32, 64, 128, ...) olarak seçilmektedir.

Bir veri setindeki bütün verilerin, ileri beslenmesi ve geri yayılım ile model ağırlıklarının güncellenmesine döngü adı verilir. Döngü sayısı, söz konusu veri setinin, kaç kez bu döngüyü tekrarlayacağını belirtir. Döngü sayısının, en azından ağına yakınsamasını sağlayacak şekilde seçilmesi önemlidir. Düşük döngü sayısı yetersiz öğrenmeye sebep olabilir. Çok yüksek döngü sayısı ise belli bir aşamadan sonra aşırı öğrenmeye neden olabilir. Pratikte sıklıkla kullanılan yaklaşım, yüksek bir döngü sayısı ile ağı eğitmek, fakat ağına başarımlı ve hata sonuçlarında bir iyileşme görülmediği takdirde eğitimi sonlandırmak üzeredir.

3.3. Evrişimsel Sinir Ağları Tabanlı Derin Öğrenme Modelleri

Son yıllarda özellikle bilgisayarla görü alanında kullanılmak üzere birçok ESA tabanlı derin öğrenme mimarileri önerilmiştir. Bu kısımda, 2012 yılında önerilen AlexNET modeline ve daha sonrasında bilgisayarla görü alanındaki başarımlı bir üst seviyeye taşıyan VGGnet ailesi, Inception ailesi ve ResNET ailesine üye modellere yer verilmiştir.

3.3.1. AlexNET

Bilgisayarla görü alanında ESA'ları tekrar popüler hale getiren AlexNET, Alex Krizhevsky, Ilya Sutskever ve Geoffrey Hinton tarafından geliştirilmiştir. ILSVRC-2012'de %15,3 top-5 hata oranına ulaşan AlexNET, bu sonuç ile ikinciden %10,8 daha yüksek başarımlı göstermiş ve derin ESA'ların bilgisayarla görü alanındaki potansiyelini gözler önüne sermiştir. AlexNET, modelin derinliğinin yüksek performans için kesinlikle gerekli olduğunu ileri sürmektedir. Model derinliği ile ortaya çıkan işlem yükünü ise GPGPU'ların kullanımı ile çözmüştür.

AlexNET modeli, 5 evrişimsel katman, 3 havuzlama katmanı, 2 tam bağlı katman ve 1 softmax katmanından oluşmaktadır. Bütün evrişim katmanlarının sonrasında aktivasyon fonksiyonu olarak

ReLU kullanılmış, ek olarak ilk iki evrişim katmanında ReLU aktivasyonu sonrası normalizasyon olarak LRN kullanılmıştır. Her bir tam bağlı katmandan sonra aktivasyon fonksiyonu olarak ReLU ve 0,5 oranında seyreltme kullanılmıştır. Havuzlama işlemi, maksimum havuzlama ile gerçekleştirilmiştir. AlexNET modeli ortalama 60 milyon parametre içermektedir. İlk AlexNET modeline ilişkin katmanlar ve boyutları Tablo 3.1’de verilmiştir.

AlexNET’in literatüre katkıları şu şekilde özetlenebilir:

- Veri artırımı yardımıyla aşırı öğrenmenin engellenmesi. Veri setindeki varyasyonu artırmak için aynalama ve kesme tarzı veri artırımı yaklaşımları kullanılmıştır.
- AlexNET öncesinde en sık kullanılan aktivasyon fonksiyonları sigmoid ve tanh’dır. Bu fonksiyonlar, yapıları gereği doygunluğa ulaşan fonksiyonlardır ve bu nedenle kaybolan gradyan sorunundan etkilenmektedirler. AlexNET, aktivasyon fonksiyonu olarak ReLU’yu kullanmaktadır. Çalışmalarında, ReLU’yu kullanarak %25 hata oranına, tanh ile ulaştıklarından 6 kat daha hızlı ulaştıklarını belirtmişlerdir.
- ReLU, her ne kadar kaybolan gradyanlar problemine çözüm getirse de çıktıları oldukça yüksek olabilmektedir. Bu nedenle AlexNET, LRN adında bir normalizasyon şeması sunarak, yüksek değerde çıktı veren ağırlıkların etrafındaki diğer ağırlıklara göre normalleştirilmesini mümkün kılmıştır.
- AlexNET aynı zamanda aşırı öğrenmeyi engellemek adına seyreltme katmanlarını da kullanmıştır. $P = 0,5$ olarak belirlenen seyreltme olasılığıyla ağ, kötü yerel minimumlardan kaçınmakta fakat yakınsama için gereken iterasyon miktarı da iki katına çıkmaktadır.

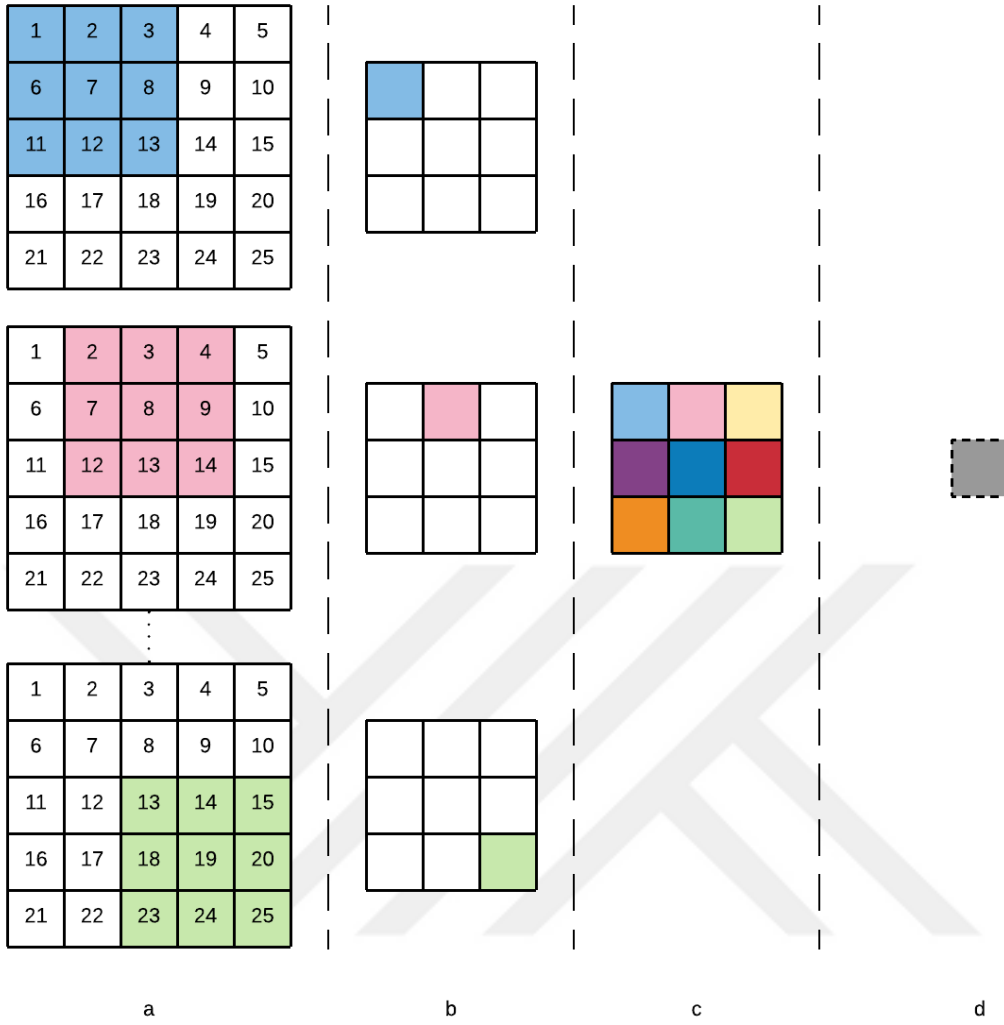
3.3.2. VGGNet mimarisi

Simonyan ve Zisserman tarafından sunulan VGGNet, temel olarak AlexNET’in çok yüksek olan parametre sayısını azaltmayı ve ağırlık eğitimini hızlandırmayı hedeflemektedir [100]. Bunun yanı sıra AlexNET’in, derin ağların daha iyi sonuçlar vereceği öngörüsünü, kendi ağlarını daha derin tasarlayarak sınamışlardır.

Tablo 3.1. AlexNET modelinin katmanları, boyutları ve filtre parametreleri.

Katmanlar	Boyut	Filtre/Kaydırma
Girdi Katmanı	224x224x3	
Evrişim_1 (ReLU-LRN)	55x55x96	11x11x3/4x4
Maksimum Havuzlama_1	27x27x96	3x3/2x2
Evrişim_2 (ReLU-LRN)	27x27x256	5x5/1x1
Maksimum Havuzlama_2	13x13x256	3x3/2x2
Evrişim_3 (ReLU)	13x13x384	3x3/1x1
Evrişim_4 (ReLU)	13x13x384	3x3/1x1
Evrişim_5 (ReLU)	13x13x256	3x3/1x1
Maksimum Havuzlama_3	6x6x256	3x3/2x2
Tam Bağlı_1 (ReLU – Seyreltme (0.5))	4096	
Tam Bağlı_2 (ReLU – Seyreltme (0.5))	4096	
Softmax	1000	

Parametre sayısını azaltmak için kullandıkları temel yaklaşım, AlexNET’te kullanılan farklı büyüklükteki filtreleri (11x11 ve 5x5), daha küçük boyuta sahip olan 3x3 filtreler ile değiştirmektir. Bu yaklaşımla özellik haritaları üzerindeki evrişim işlemi, daha az sayıda parametre ile gerçekleştirilecektir. Bunu bir örnekle açıklayalım: 5x5 boyutunda bir özellik haritasına, 5x5 boyutunda bir filtre uygulanarak 1x1 boyutunda bir özellik haritası elde edilmektedir. 5x5 boyutundaki aynı özellik haritasına, 3x3 boyutunda bir filtre uygulandığı takdirde 3x3 boyutunda bir özellik haritası, ardından bu özellik haritasına tekrar 3x3 boyutunda bir filtre uygulandığı zaman 1x1 boyutunda bir özellik haritası elde edilir. İlk yaklaşımdaki 5x5’lik filtre için 25 parametre kullanılırken, ikinci yaklaşımdaki iki adet 3x3 filtre için 9+9=18 parametre kullanılmaktadır. Bu da %28 oranında bir parametre tasarrufuna denk gelmektedir. Benzer bir şekilde, 7x7 boyutundaki filtre, 3 adet 3x3 boyutundaki filtre ile temsil edildiğinde %45 oranında, 11x11 boyutundaki filtre, 5 adet 3x3 boyutundaki filtre ile temsil edildiğinde ise %63 oranında daha az parametre gerektirmektedir. Parametre sayısının azaltılması, hızlı bir eğitim sürecine destek olmakla birlikte parametre sayısının yüksekliğinden kaynaklanan aşırı öğrenme sorununa bir çözüm getirmektedir. Şekil 3.4’de, 5x5 boyutundaki özellik haritası ve uygulanan 3x3’lük filtrenin pencere konumları (a), her bir pencere konumu için elde edilen sonucun ara özellik haritası üzerindeki konumu (b), ilk evrişim katmanı sonucu elde edilen ara özellik haritası (c) ve 3x3’lük ara özellik haritasına uygulanan 3x3’lük evrişim işleminin sonucu (d) olmak üzere 5x5 boyutundaki bir özellik haritasına uygulanan parçalı evrişim yaklaşımına ait görseller verilmiştir.



Şekil 3.4. 5x5 boyutundaki bir özellik haritası üzerinde gerçekleştirilen parçalı evrişimin gösterimi.

VGGNet ailesi, sonundaki rakam katman sayısını belirtmek üzere VGG11, VGG13, VGG16, VGG19 ve bunların türevlerinden oluşan toplam 6 adet modelden oluşmaktadır. Çalışmalarında, ILSVRC-2012 veri tabanı için, 11 katmanlı modelden 16 katmanlı modele doğru gidildikçe başarımın arttığını daha sonrasında ise kayda değer bir değişim olmadığını belirtmişlerdir. Bu nedenle VGGNet mimarisi söz konusu olduğunda, bahsedilen modeller genellikle 16 katmanlı VGG16 ve 19 katmanlı VGG19'dur. VGGNet mimarisine ait farklı konfigürasyona sahip modellerin katmanları Tablo 3.2'de verilmiştir. İlgili tabloda A'dan E'ye doğru gidildikçe model derinliği artmaktadır. Konfigürasyonların altlarında katman ve toplam parametre sayıları verilmiştir. Her bir konfigürasyonda eklenen katmanlar, siyah ile gösterilmiştir. Evrişim katmanı parametreleri “conv(pencere boyutu)-(kanal sayısı)” olarak gösterilmiştir. ReLU aktivasyon fonksiyonu, sadelik olması açısından yazılmamıştır.

Tablo 3.2. VGGNet mimarisi modellerinin konfigürasyonları.

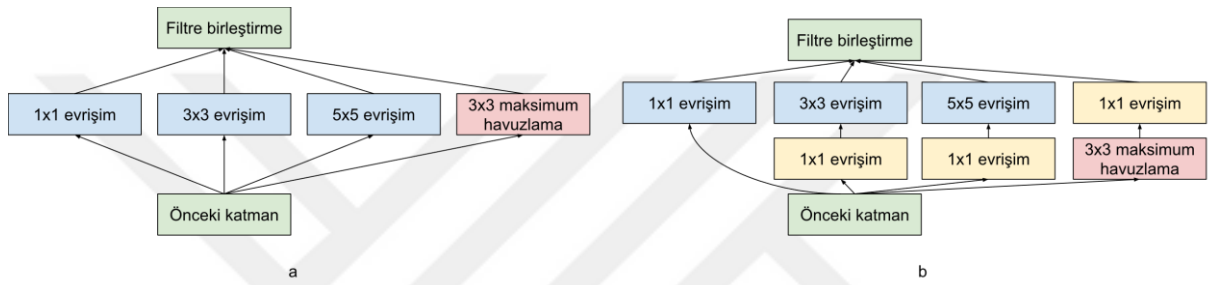
Konfigürasyonlar					
A	A-LRN	B	C	D	E
11 katman (133 m)	11 katman (133m)	13 katman (133m)	16 katman (134m)	16 katman (138m)	19 katman (144m)
girdi (224 x 244 RGB imge)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maksimum havuzlama					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maksimum havuzlama					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maksimum havuzlama					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maksimum havuzlama					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maksimum havuzlama					
FC-4096					
FC-4096					
FC-1000					
Softmax					

3.3.3. Inception mimarisi

Inception mimarisinin ilk örneği olan Inception-v1, Szegedy vd. tarafından ILSVRC-2014 yarışması için tasarlanmış ve söz konusu yarışmayı birincilikle tamamlamıştır. Bu başarı üzerine geliştirilmeye devam edilen mimari, Inception-v1, Inception-v2, Inception-v3, Inception-v4, Inception-ResNet-v1 ve Inception-ResNet-v2 olmak üzere altı farklı üyeden oluşmaktadır.

Bir imge üzerinde bulunan belirleyici özellikler, farklı ölçeklerde bulunabilmektedir. Örneğin bir kedinin sadece kafasının bulunduğu bir imgenin kedi olarak sınıflandırılması gerekirken, aynı kedinin ağaç üzerindeki görüntüsünün de kedi olarak sınıflandırılması gerekir. Sınıflandırılacak olan nesnenin farklı konum varyasyonlarının bulunması, kullanılacak olan evrişim filtresinin seçimini güçleştirmektedir. Çünkü büyük filtreler resmin geneline yayılan özellikleri yakalamakta başarılıyken küçük filtreler resmin kısmi bölümlerinde bulunan özellikleri yakalamakta başarılıdır. Bu nedenle seçilecek filtrenin, imgenin karakteristiğine uygun olması gerekmektedir. Bu ise büyük veri tabanlarında mümkün değildir. Inception mimarisi, bu sorunu, aynı katmanda bulunan ve

paralel olarak çalışan farklı boyutlardaki filtreler ile çözmüştür. Bu mimariyi kendisinden önceki tasarımlardan ayıran temel özellik, ağ derinliği sağlamak için evrişim katmanlarını üst üste yığmaktan farklı olarak tasarlanan Inception modüllerini kullanmasıdır. Şekil 3.5’de gösterilen her bir Inception-v1 modülü, paralel olarak birbirlerine bağlı olan bir dizi evrişim katmanı içerir. Bu evrişim katmanlarında, görüntüden farklı özellikler çıkarmak için 1x1, 3x3 ve 5x5 boyutlarında filtreler kullanılmaktadır. Her bir katman dizisi, modülün sonunda birleştirilerek modülün çıktısını oluşturmaktadır. 1x1 filtreler, boyut indirgenmesi için kullanılmakta ve bu sayede işlem yükünü azaltılmaktadır. Buna ek olarak 1x1 filtrelerde kullanılan aktivasyon fonksiyonları lineerliği bozmakta, bu da ağın genellemesine yardımcı olmaktadır.

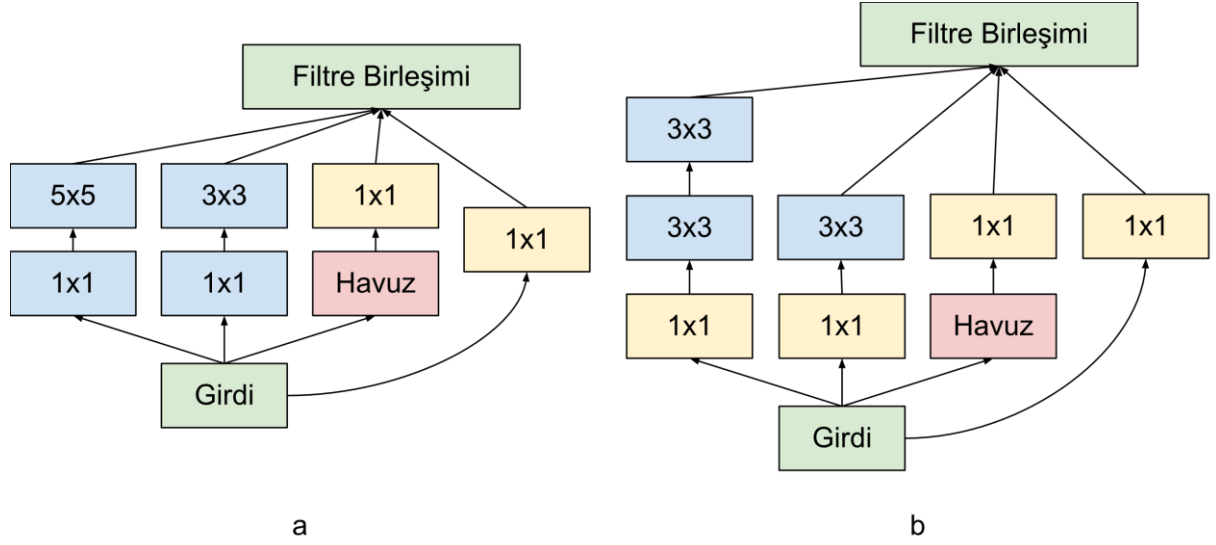


Şekil 3.5. Inception-v1 sürümünde önerilen inception modülleri a. Saf inception modülü b. Boyut indirgemeli inception modülü

Inception-v2 modeli, Inception-v1 modelinden farklı olarak, VGGNet benzeri filtre faktörizasyonu yoluna gitmiştir. 5x5 boyutundaki filtreler iki adet 3x3 filtre ile değiştirilmiş ve bu sayede %28 oranında bir parametre tasarrufu sağlanmıştır. Şekil 3.6’da orijinal inception modülü ve faktörize edilmiş inception modülü beraber verilmiştir. Bunun yanı sıra parametre regularizasyonu için yığın normalizasyonu kullanılmıştır.

Inception-v3 modeli, işlem hızını artırmak ve bu hız artışını ağ derinliği tarafında kullanmak için farklı bir faktörizasyon tekniği sunmuştur. 7x7 boyutundaki evrişim filtresinin işlevini, 7x1 ve 1x7’lik iki adet evrişim filtresi ile gerçekleştirmişlerdir. Aynı şekilde 3x3 boyutundaki evrişim filtresi ise 3x1 ve 1x3’lük filtre ile değiştirilmiştir. Bu sayede ciddi oranda parametre azaltışına gitmişler ve bu kapasiteyi ağı daha da derinleştirme de kullanmışlardır.

Inception-v4 modeli, saf Inception yaklaşımının kullanıldığı son modeldir. Inception-v3 modeline nazaran daha fazla alt modül içermektedir. Inception-ResNet-v1 ve Inception-ResNet-v2 modellerinde, ResNET mimarisinden esinlenerek, bloklar arasında residual bağlantılar kullanılmıştır. Çalışmalarında bu blokları ağıın başarımını artırmak için değil, yakınsama hızını artırmak için kullandıklarını belirtmişlerdir. Residual bağlantılar sayesinde ağıın yakınsama hızı artmış fakat başarımında küçük bir düşüş gözlemlenmiştir.



Şekil 3.6. Inception modülünün faktörizasyon işlemi. a. 5x5 evrişim filtresi içeren modül b. 5x5 evrişim filtrelerinin 3x3 evrişim filtreleri ile değiştirilmesi sonucu elde edilen modül.

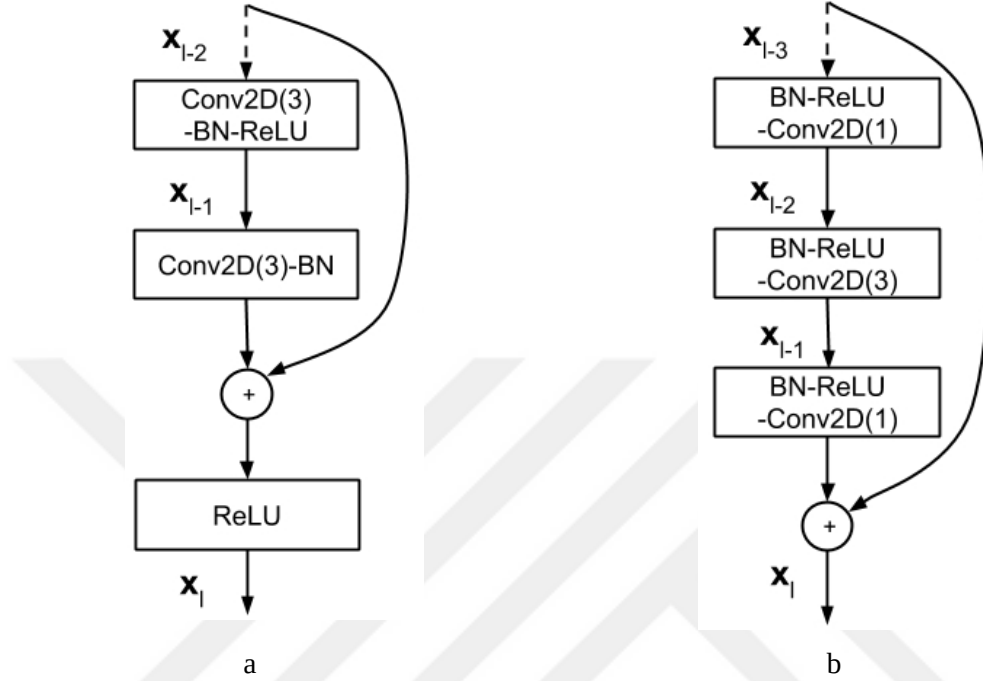
3.3.4. ResNet mimarisi

ResNet, He vd. tarafından 2015 yılında ILSVRC2015 yarışması için önerilmiş ve söz konusu yarışmayı birincilikle tamamlamıştır. ResNet mimarisi, derinleşen ağlardaki eğitimi zorlaştıran süreçlere ilişkin bir çözüm sunmaktadır.

Derin ağların, derin olmayan ağlara oranla daha başarılı olduğunun görülmesi üzerine birçok araştırmacı kullandıkları ağları derinleştirmeye odaklanmıştır. İlk bakışta bir ESA'yı daha derin tasarlamaya yönelik tek dezavantajın, parametre sayısının artmasıyla oluşacak olan işlem yükünün artması olduğu düşünülebilir. Fakat YSA'larda olduğu gibi ESA'lar da geri yayılım algoritmasıyla eğitilen modellerdir. Geri yayılım sürecinde, hata fonksiyonunun çıktısının yardımıyla ağ üzerindeki ağırlıklar tekrar hesaplanır. Bu hesaplama esnasında zincir kuralından faydalanılır. Zincir kuralındaki çarpma teriminden dolayı üst katmanlara doğru gidildikçe gradyan değerleri azalır. Çok derin ağlarda bu değer sıfıra yakınsar ve sıfır olan ağırlıklar öğrenmeye herhangi bir katkıda bulunmaz. Bu olguya kaybolan gradyan problemi adı verilmektedir.

ResNet, kaybolan gradyan sorununu çözmek için residual bağlantıları önermiştir. Geleneksel sinir ağlarında her bir katmanın çıktısı bir sonraki katmana girdi olarak verilmektedir. Residual bağlantılar sayesinde bir katmanın çıktısı kendisinden sonraki katmana değil daha sonraki katmanlara girdi olarak iletebilmektedir. Bu şekilde iki fayda sağlanmaktadır. Bunlardan birincisi, ağın ilk aşamalarındaki özelliklerin, çok fazla değişime uğramadan alt katmanlara iletebilmesidir. İkincisi ise geri yayılım esnasında bu bağlantılardan geçen gradyanların değişime uğramadan birkaç yukarıdaki katmana iletebilmelidir. Böylelikle gradyanlar yukarıya doğru aktıkça büyüklükleri kontrol altında tutulmakta ve kaybolan gradyan sorununa bir çözüm getirilmektedir.

ResnetV2'nin ResnetV1'den farklılaştığı nokta, kısa yol bağlantılarının daha fazla blok sonrasında gerçekleşmesi ve blokların içerisindeki işlem sıralarının değişmesidir. Şekil 3.7'de ResNetV1 ve ResNetV2 için önerilen residual bağlantılar verilmiştir.



Şekil 3.7. ResNet mimarisinde kullanılan residual modüller. a. Residual Modül v1 b. Residual Modül v2

3.3.5. Kullanılan ESA Modelleri ve Seçim Nedenleri

Bu çalışma kapsamında Inceptionv3, ResNet-50 ve önerilen 9 katmanlı bir ESA modeli kullanılmıştır. Bu ağların seçilmesindeki asıl amaç, farklı ağ mimarisi yaklaşımlarının, YİT problemi üzerindeki başarımlarını incelemektir.

AlexNET modeli ve bundan daha fazla katmana sahip olan VGGNet modelleri, lineer bir şekilde bağlı olan ve ağ derinliğini ön plana çıkaran yaklaşımlardır. Inceptionv3 modeli, Inception modüllerini kullanarak, aynı katman seviyesinde farklı özelliklerin eş zamanlı olarak çıkarılması yardımıyla daha az parametre içeren ağlar ile daha yüksek başarıyı hedeflemiştir. ResNet-50 modeli ise residual bağlantılar yardımıyla gradyanların muhafaza edilmesini sağlayarak kaybolan gradyanlar sorununa bir çözüm getirmektedir.

Söz konusu ağlardan, VGGNet-16 ağı 138 milyon, Inceptionv3 ağı 24 milyon, ResNet-50 ağı 26 milyon ve önerilen 9 katmanlı ağ ise 420 bin parametre içermektedir. VGGNet-16 ağı, başta ILSVRC-2012 olmak üzere farklı veri tabanları üzerinde yapılan kıyaslamalar göz önüne alındığında, Inception ve ResNet mimarilerine oranla, daha fazla parametre içermesine karşın çok daha düşük başarımla sergilenmektedir. Bu nedenle bu kadar fazla parametre içeren çok derin bir ağı

çalışma kapsamında kullanılması uygun görülmemiştir. Zira kısıtlı sayıda örnek içeren veri setinin, bu tarz ağlarda aşırı öğrenmesi muhtemeldir. Lineer bir şekilde bağlı mimarileri temsil etmek amacıyla 9 katmandan oluşan bir ağ önerilmiş ve kullanılmıştır. Inceptionv3 ve ResNet-50 ağlarına ise, getirdikleri farklı mimari yaklaşımlardan dolayı çalışmalarda yer verilmiştir.

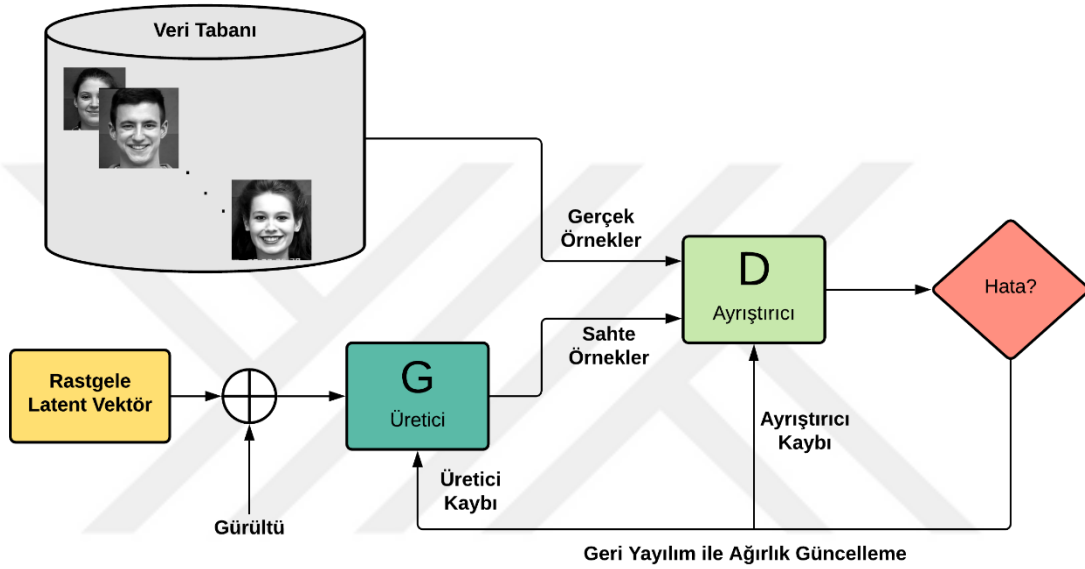
3.4. Çekişmeli Üretici Ağlar

Makine öğrenmesi yöntemleri, öğrenme yöntemlerine göre denetimli öğrenme, denetimsiz öğrenme ve takviyeli öğrenme olarak genellikle üç sınıfa ayrılır. Denetimli öğrenmede, eğitim esnasında etiketlenen girdilerden faydalanılarak öğrenme işlemi gerçekleşir. SVM, lineer regresyon, k en yakın komşu algoritması denetimli öğrenmeye örnek olarak verilebilir. Denetimsiz öğrenmede girdiler salt olarak verilir ve bu girdiler arasındaki yapının öğrenilmesi beklenir. PCA, k-means ve diğer kümeleme algoritmaları denetimsiz öğrenmeye örnek olarak verilebilir. Takviyeli öğrenme ise, diğer iki öğrenme tipinden farklı olarak girdi ve çıktıları değil, gözlem ve mevcut bilginin kullanılması ile gerçekleştirilir. Bu genel sınıflandırmanın dışında makine öğrenmesi yöntemleri, parametrik – parametrik olmayan, ağaç tabanlı – ağaç tabanlı olmayan veya ayrıştırıcı modeller – üretici modeller olarak da sınıflara ayrıştırılabilmektedir. Bu kısımda ayrıştırıcı ve üretici modellere ve üretici modellerin son yıllardaki en iyi örneklerinden ÇÜA’lardan bahsedilecektir.

Üretici modeller, veri kümesindeki farklı sınıfların dağılımına odaklanır ve veri noktalarının dağılımlarını ve desenlerini modellemeye çalışırlar. Üretici modeller bu modelleme esnasında ortak olasılık dağılımı $p(x, y)$ ’yi öğrenirler. Üretici sınıflandırıcılara, Naive Bayes, Bayes ağları ve gizli Markov modelleri örnek olarak verilebilir. Ayrıştırıcı modeller, gözlemlediği verilerden faydalanarak sınıflar arasındaki karar sınırını öğrenmeye çalışırlar. Bunun için koşullu olasılık dağılımı $p(y|x)$ ’i öğrenirler. Ayrıştırıcı sınıflandırıcılara, lojistik regresyon, geleneksel YSA’lar ve en yakın komşu algoritması örnek olarak verilebilir. Daha öz bir şekilde anlatmak gerekirse, üretici modeller yeni örnekler üretebilirken, ayrıştırıcı modeller ise iki farklı örnek arasındaki farkı ayırt etmeyi hedeflerler. İki modelin de genel uygulama alanları, denetimli öğrenme problemleridir. ÇÜA’lar, YSA’ları temel alan ve imge veri setleri üzerinde yapay örnek oluşturmada çığır açan üretici modellerdir. Şekil 3.8’de ÇÜA’ya ait bir illüstrasyon verilmiştir. İki sinir ağının birbiri ile yarışmasını öngören bu fikir, 2014 yılında Ian Goodfellow tarafından ortaya konulmuştur [101]. Bu iki sinir ağı, üretici ağ (G) ve ayrıştırıcı ağ (D) olarak adlandırılır. Üretici ağ, gerçeğe benzetmeye çalıştığı sahte veriler üretirken, ayrıştırıcı ağ, gerçek ve sahte verileri ayırt etmeye çalışır. Üretici ağ, ayrıştırıcı ağın kayıp fonksiyonunu kullanarak geri yayılım algoritması ile ağırlıklarını günceller. Bu sayede orijinal veri setindeki örneklerle benzer sahte veriler üretmeye çalışır. Bu eğitim işlemi, ayrıştırıcı ağın, sahte ve orijinal verileri ayırt edemediği noktaya kadar sürer. Ayrıştırıcı ağın farklı iki kaynaktan gelen verileri ayrıştırıramaması, üretici ağın dağılım

bakımından gerçek veri seti ile aynı dağılımda çıktı ürettiği anlamına gelmektedir. Verilen bir $z \sim p_z$ dağılımında G , $G(z)$ örneklerinin dağılımı olan bir p_g olasılık dağılımını tanımlasın. ÇÜA'nın amacı, gerçek veri dağılımı p_r 'ye yakınsayan, p_g üretici dağılımını öğrenmektir. ÇÜA optimizasyonu, Denklem 3.11'de verilen D ve G 'ye bağlı bileşik hata fonksiyonu ile gerçekleştirilir:

$$\min_G \max_D \mathbb{E}_{x \sim p_r} \log[D(x)] + \mathbb{E}_{z \sim p_z} \log[1 - D(G(z))] \quad (3.11)$$



Şekil 3.8. Geleneksel bir Çekişmeli Üretici Ağ mimarisinin gösterimi.

Derin Üretici Modellerin bir üyesi olan ÇÜA'lar, geleneksel DÜM yaklaşımlarına kıyasla sağladığı avantajlardan dolayı derin öğrenme toplulukları tarafından aşırı bir ilgi ile karşılanmaktadır [102]. Sağlanan bu avantajlar, çıktı üretimi, üretici ağ türü ve gizli değişken boyutu ile özetlenebilir. Öncelikle ÇÜA'lar, diğer DÜM'lere nazaran daha iyi çıktılar üretebilmektedirler. Bilinen en iyi DÜM'lerden biri olan VAE'ler ile kıyaslanırsa, ÇÜA'lar her türlü olasılık yoğunluğunu üretebilirken, VAE'ler keskin imgeler üretememektedirler [103]. Bunun yanı sıra ÇÜA mimarisi, herhangi bir üretici ağ türü ile eğitilebilmekteyken diğer DÜM'ler, üretici çıktı katmanının Gauss olması gibi ön şartlara sahip olabilmektedirler [104]. Son olarak ÇÜA'larda gizli değişkenin boyutu ile ilgili bir kısıtlama olmazken farklı DÜM'lerde, ağ şemasına bağlı olarak gizli değişken boyutunda kısıtlamalar bulunabilmektedir. Bütün bu avantajlar, özellikle sentetik imge verilerinin üretiminde ÇÜA'ların olağanüstü sonuçlar ortaya koymasına imkan sağlamaktadır [102].

ESA'ların kullanıldığı ÇÜA'lara ilk örnek olarak Radford v.d tarafından 2015 yılında önerilen DCGAN verilebilir [105]. Bu çalışma, evrimsel katmanların ÇÜA'lar ile nasıl kullanılabileceği açıklamakta ve bunun için bir dizi mimari öneride bulunmaktadır. Bunun yanı sıra bu çalışma, ÇÜA

özelliklerinin görüntülenmesi, gizli uzay interpolasyonu ve sınıflandırıcı eğitimi için ayrıştırıcı özelliklerinin kullanılması ve sonuçların değerlendirilmesi gibi konularda da önemli bilgiler sunmuştur.

ÇÜA üzerinde yapılan çalışmaları sınıflandırmak için güncel bir taksonomi, Wang vd. tarafından sunulmuştur [102]. Bu sınıflandırma, ÇÜA'ları, mimari açıdan ağ mimarisi, gizli uzay ve uygulama odağı olarak, kayıp azaltma açısından ise kayıp türleri ve regularizasyon olarak sınıflandırmaktadır. Bu tez çalışmasının kapsamı yüz ve YİT ile sınırlı kaldığı için, bu bölümün devamında genel olarak sentetik yüz oluşturma odaklı ÇÜA tabanlı çalışmalarından bahsedilecektir.

3.4.1. FCGAN

2014 yılında Ian Goodfellow vd. tarafından önerilen çalışma, ÇÜA'lar ile sentetik yüz oluşturmaya yönelik ortaya konulan ilk çalışmadır [101]. ÇÜA fikrinin ilk kez ortaya konulduğu bu çalışma, üretici ve ayrıştırıcı ağlarda birbirlerine tam bağlı çok katmanlı algılayıcıları kullandığından ötürü genellikle FCGAN (Fully Connected GAN) olarak adlandırılır. Sentetik yüzler oluşturmak için Toronto yüz veri setinin [106] kullanıldığı bu çalışmada üretilen sentetik yüzler, Şekil 3.9'da verilmiştir. FCGAN ile üretilen yüzler, ağırlık kapasitesinden dolayı düşük çözünürlüklüdür.



Şekil 3.9. FCGAN ile Toronto Face Dataset üzerinde gerçekleştirilen eğitim sonucu üretilen yüzler.

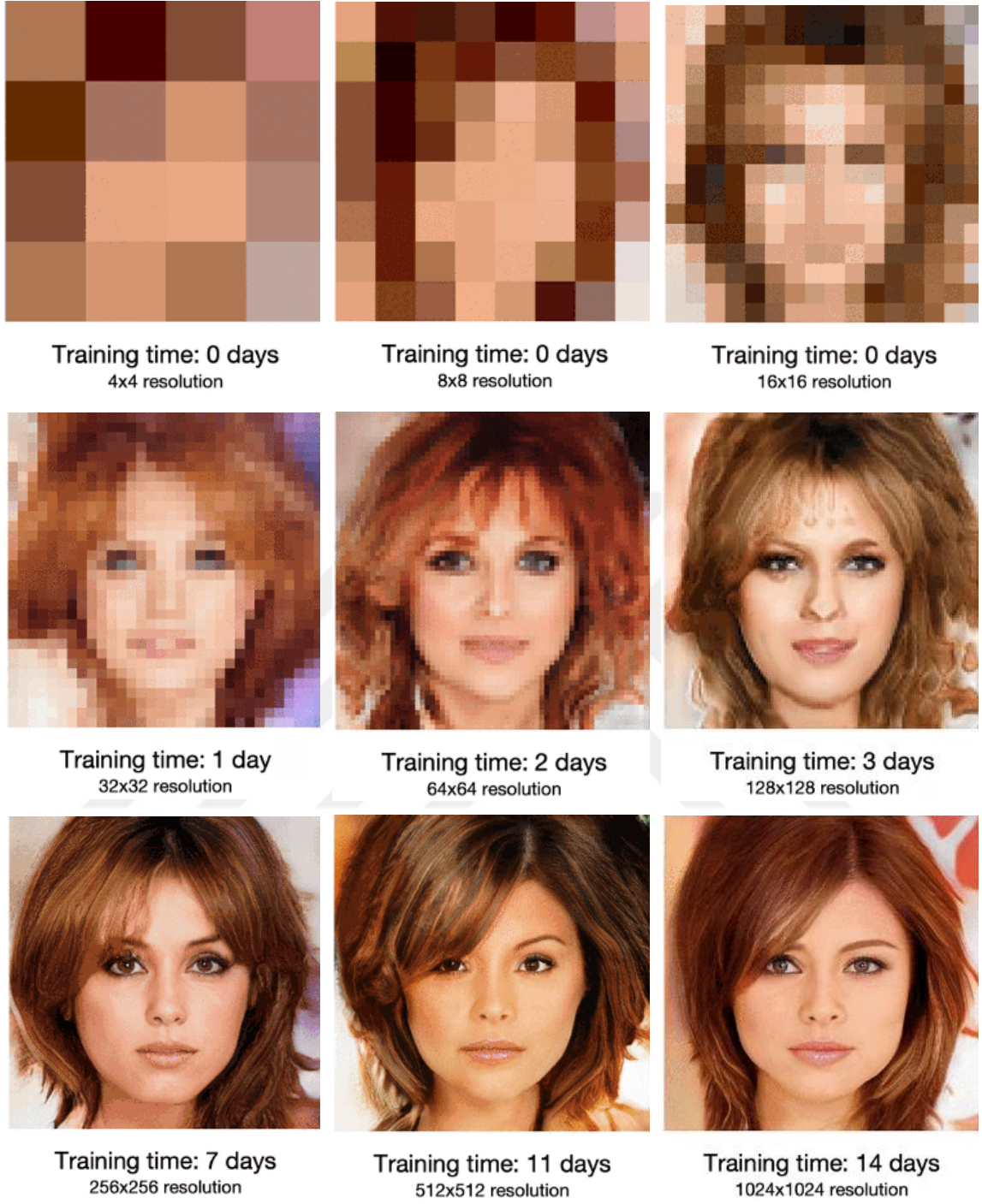
3.4.2. ProGAN

ÇÜA'lar ile yüksek çözünürlük ve detaya sahip sentetik yüz üretmeye yönelik çalışmalar, başlıca NVIDIA araştırma grubu önderliğinde gerçekleştirilmektedir. Bu alanda gerçekleştirilen ilk önemli

çalışma Karras vd. tarafından 2018 yılında önerilen ve aşamalı olarak büyüyen bir eğitim yaklaşımına sahip olan ProGAN olmuştur [107]. Yine NVIDIA araştırma grubu önderliğinde Karras vd. tarafından 2019 yılında sunulan StyleGAN, ProGAN'ı temel alarak farklı bir üretici mimarisi sunmuş ve sentetik yüz üretimine farklı bir boyut kazandırmıştır[108]. Son olarak StyleGAN'ı öneren ekip, söz konusu mimaride birtakım değişiklikler uygulamış ve daha kaliteli görüntüler üreten StyleGAN2'yi sunmuşlardır [7]. An itibarıyla StyleGAN2, sentetik yüz üretiminde ÇÜA'ların geldiği en son noktayı temsil etmektedir.

ProGAN (Progressive Growing GAN), aşamalı olarak büyüyen bir eğitim yaklaşımına sahip olan bir ÇÜA'dır. Genel olarak ÇÜA'larda üretici (Ü) ve ayrıştırıcı (A) ağlar eğitim öncesinde tasarlanır, eğitim ve güncelleme parametreleri belirlenir ve ardından eğitim sürecine geçilir. Genel ÇÜA eğitim yaklaşımının aksine ProGAN, burada farklı bir yaklaşım sunmaktadır. Öncelikle eğitim görüntüleri çok küçük boyutlara dönüştürülmekte (4x4 piksel), ve az sayıda katman içeren bir üretici ve ayrıştırıcı yardımıyla eğitim görüntüleri sentezlenmeye çalışılmaktadır. Ağlar çok küçük oldukları için eğitim hızlı bir şekilde yakınsamakta ve bulanık olarak verilen imgelerde görülen büyük ölçekli yapılar (ten rengi, saç rengi) öğrenilmektedir. İlk katmanlar eğitildikten sonra, üretici ve ayrıştırıcıya yeni katmanlar eklenmekte, çıktı boyutu 8x8 olarak ayarlanmakta ve eğitim süreci kaldığı yerden devam etmektedir. İstenilen çözünürlük ve sentezleme kalitesine ulaşıncaya kadar katman ekleme ve eğitim süreci devam etmektedir. Şekil 3.10'da ProGAN'a ait eğitim süreci ve her aşamada oluşturulan görüntüler verilmiştir. Eğitim 4x4 çözünürlükte başlamakta ve 1024x1024 çözünürlükle birlikte 14 günde tamamlanmaktadır. Soldan sağa ve yukarıdan aşağıya doğru artan bir şekilde, ölçek veya eğitim günü değişimine göre örneklenmiş ağ boyutu ve sentezlenen görüntüler şekilde gösterilmektedir.

ProGAN, çözünürlüğü kademeli olarak artırarak, ağlardan sürekli olarak genel sorunun çok daha basit bir parçasını öğrenmelerini hedeflemektedir. Ayrıca kademeli öğrenim, eğitim sürecinin stabilize bir şekilde devam etmesine büyük ölçüde yardımcı olur. Düşük çözünürlükte yüksek çözünürlüğe doğru gidildikçe, öncelikle bulanık görüntülerde seçilebilen büyük örüntüler öğrenilmekte, ardından ince detaylar öğrenilmektedir. Bu şekilde ağı, büyük ölçekli yapıları yanlış temsil etmesinin de önüne geçilmektedir. Son olarak ağ boyutunun kademeli bir şekilde artırılması, bütün katmanları eş zamanlı olarak çalıştıran geleneksel yaklaşımlara oranla ProGAN'ı hesaplama açısından daha verimli kılmaktadır. Daha az sayıda katman, daha az parametre içerdiğinden dolayı daha hızlı eğitilir. Eğitim işleminin son kademesinde eğitilen en büyük ağ hariç diğer bütün ağlar, genel ağı bir alt kümesi olduğundan dolayı hesaplama verimliğinde kayda değer bir verimlilik gözlenir. Karras vd., çıktı çözünürlüğüne bağlı olarak ProGAN'ın, eşdeğer bir ÇÜA'ya oranla 2-6 kat arası daha hızlı eğitildiğini belirtmiştir.



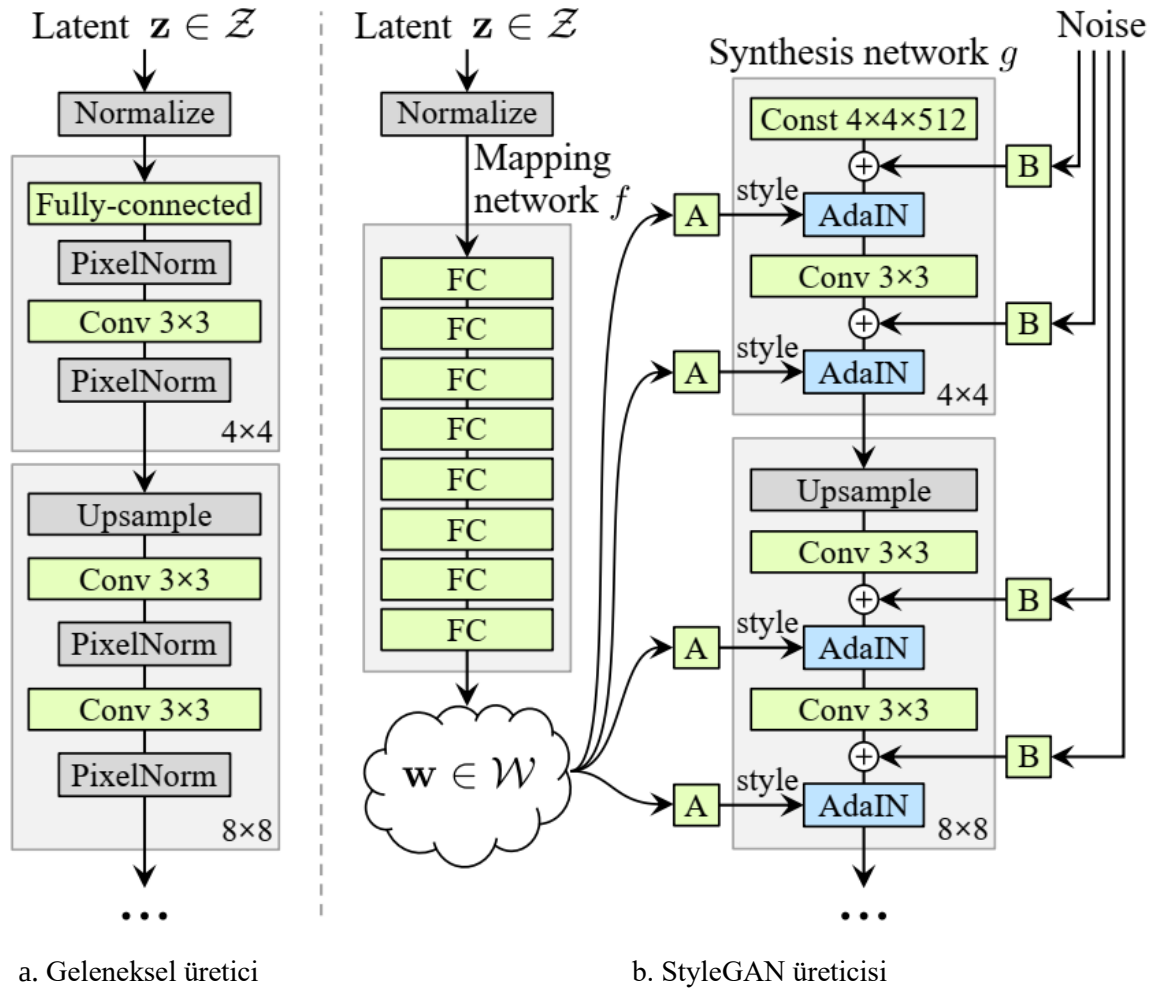
Şekil 3.10. ProGAN'a ait eğitim süreci ve oluşturulan görüntüler.

3.4.3. StyleGAN

StyleGAN, Karras vd. tarafından 2018 yılından stil transferi literatüründen esinlenerek geliştirilmiştir [108]. ProGAN'ı temel alan StyleGAN, üretici mimarisini değiştirerek görüntü sentezleme sürecini yeni yollarla kontrol etmeye çalışmıştır. StyleGAN'ın getirdiği farklılıklar

haritalama ağının önerilmesi, normalleştirme için AdaIN modülü ve stokastik varyasyon olarak üç başlık altında özetlenebilir:

StyleGAN'ın getirdiği en büyük değişiklik, haritalama ağının önerilmesidir. Geleneksel olarak ÇÜA'larda Z gizli uzayının bir elemanı olan gizli vektör z , üretici ağı bir girdi katmanını aracılığıyla verilir. StyleGAN ise öncelikle gizli vektör z 'yi bir haritalama ağı yardımıyla, ara bir gizli uzay olan W 'ya haritalar. Ardından bu ara gizli uzay W 'dan elde edilen ara gizli vektör w yardımıyla, farklı katmanlarda farklı özellikleri kontrol eder. Bu şekilde her bir katmanda farklı bir özellik kontrol edilerek, üretici ağı daha özgür bir şekilde modellemesi sağlanır. AdaIN yardımıyla w ara gizli vektöründen elde edilen stiller, üreticinin ürettiği imgeye aktarılır. Son olarak çil, saç ve gamze gibi yüzden yüze değişen stokastik özellikler, AdaIN modülünden önce her bir katmana eklenen gürültü yardımıyla oluşturulmaya çalışılmıştır. Şekil 3.11'de ProGAN ve StyleGAN ağlarının üreticileri karşılaştırılmıştır.



Şekil 3.11. ProGAN ve StyleGAN üreticilerinin karşılaştırılması. a. Geleneksel bir üretici olan ProGAN üreticisi b. Stil tabanlı StyleGAN üreticisi

3.4.4. StyleGAN2

StyleGAN her ne kadar sentetik yüz üretiminde çıtayı çok üst bir seviyeye taşısa da görsel ve mimari açıdan birtakım kusurları mevcuttur. Karras vd. tarafından 2020 yılında yayınlanan StyleGAN2 bu sorunlara çözümler üreterek sentetik yüz üretmedeki en son noktayı yeniden belirlemiştir [7].

StyleGAN mimarisindeki en önemi sorunlardan birisi damlacık kalıntıları (droplet artifacts) olarak adlandırılan ve üretilen görüntülerde ortaya çıkan damla şeklindeki bozulmalardır. Şekil 3.12'de gösterildiği üzere, damlacık son görüntüde belirgin olmasa bile, üreticinin ara özellik haritalarında mevcuttur. Bu anomaliler, 64x64 çözünürlükte görünmeye başlamakta, tüm özellik haritalarında bulunmakta ve yüksek çözünürlüklerde giderek daha güçlü hale gelmektedirler. Bu sorun, AdaIN yerine istatistiki olarak gerçekleştirilen normalleştirme yardımıyla çözülmüştür.

StyleGAN mimarisin bir diğer kusuru geçiş kalıntılarıdır (phase artifacts). Kademeli büyüme, her ne kadar yüksek yüksek çözünürlüklü görüntü sentezini stabilize etmede çok başarılı olsa da tasarım karakteristiğinden dolayı bazı dezavantajları vardır. Kademeli olarak büyüyen üretici, güçlü bir konum tercihine sahiptir. Düşük çözünürlükte yerleri belirlenen özelliklerin, yüksek çözünürlükler de başka yerlere taşınamamaktadır. Şekil 3.13'de geçiş kalıntılarına ait bir örnek gösterilmiştir. Bu sorunun üstesinden gelmek için, MSG-GAN benzeri atlamalı bağlantıları olan hiyerarşik bir üretici önerilmiştir [109].

StyleGAN2 mimarisinde gerçekleştirilen son iyileştirme, FID'nin (Frechet Inception Distance) azaltılarak görüntü kalitesinin artırılmasıdır. FID, üretilen imgelerin dağılımının, gerçek imgelerin dağılımına olan yakınlığı ölçen bir metriktir [110]. Sunulan farklı üretici yapıları ile StyleGAN ağına oranla çok daha düşük FID değerleri elde edilmiştir.



Şekil 3.12. AdaIN normalizasyonu sonucu imgelerde oluşan su damlacıkları kalıntıları. Damlacıklar, oluşturulan imgelerde tam olarak belli olmasa da üretici içindeki aktivasyon katmanlarında net olarak görülmektedir.



Şekil 3.13. Kademeli büyümenin neden olduğu geçiş kalıntıları. Bu örnekte dişler pozunu takip etmemekte, mavi çizgide gösterildiği şekilde kamera açısına bağlı kalmaktadır.

4. EVRİŞİMSEL SINIR AĞLARI VE ÇEKİŞMELİ ÜRETİCİ AĞLAR KULLANILARAK GERÇEKLEŞTİRİLEN YÜZ İFADESİ TANIMAYA YÖNELİK ÇALIŞMALAR

4.1. Kullanılan Veri Tabanı

Uygulamalar, CK+ (Extended Cohn-Kanade Dataset) veri seti üzerinde gerçekleştirilmiştir [68]. CK+ veri seti, 123 süjeden elde edilen 593 ifade sekansından oluşmaktadır. Her bir ifade sekansı, nötr yüz ifadesinden başlamakta ve ilgili yüz ifadesinin en yüksek temsili ile sonlanmaktadır. Çalışmalar kapsamında, her bir sekansın ilk imgesi nötr yüz ifadesi, en son imge veya en son üç imge ise etikette belirtilen yüz ifadesi olarak kullanılmıştır. Veri setindeki süjelerden, çalışma kapsamında kullanılmak üzere şu ifadeler toplanmıştır: Mutlu, kızgın, üzgün, şaşırma, iğrenme, korku ve nötr. Şekil 4.1’de veri setini oluşturmak için bir süjeden elde edilen ifadeler gösterilmiştir.



Şekil 4.1. Veri seti için seçilen örnek görüntüler. a. nötr b. mutlu c. iğrenme d. şaşırma e. korku

4.2. Evrişimsel Sinir Ağları Kullanılarak Gerçekleştirilen Yüz İfadesi Tanıma

Gerçekleştirilen ilk uygulamada, farklı ölçeklerde kırılan yüz imgelerinin, ESA’lar ile gerçekleştirilen YİT üzerindeki etkisi incelenmiştir. Bunun için öncelikle yüz bölgesi tespiti ve yüz hizalaması gerçekleştirilmiştir. Hizalanmış yüzler üzerinde AGM yardımıyla yüz noktaları belirlenmiştir. Ardından önerilen yüz kırpma yaklaşımıyla birlikte bu noktalar kullanılarak farklı

ölçeklerde yüzler kırılmıştır. Farklı ölçekteki bu yüzler kullanılarak bir dizi ESA, YİT için eğitilmiştir. Bu uygulama sonucunda kırılan farklı boyuttaki yüz ifadelerinin ve farklı ESA'ların, YİT üzerindeki etkilerine ilişkin çıkarımlarda bulunulmuştur.

4.2.1. Önerilen Yüz Kırma Metodolojisi

Gülbüz bir şekilde çalışması istenilen makine öğrenmesi uygulamaları için veri hazırlığı ve ön işlem önemli bir adımdır. Bu kısımda öncelikle kullanılan veri hazırlama ve ön işlem aşamalarından bahsedilecektir. Ardından tez çalışması kapsamında sunulan, farklı ölçekler ile yüzlerin kırılmasına dair yaklaşımdan bahsedilecektir.

Veri setindeki imgeler üzerinde ilk olarak yüz tespiti gerçekleştirilmiştir. Yüz tespitinin doğru bir şekilde gerçekleştirilmesi, bir sonraki adım olan yüz özellik noktalarının gülbüz bir şekilde seçilebilmesi için oldukça önemlidir. Bu uygulama kapsamında yüz tespiti için YGH + DVM tabanlı bir yaklaşım kullanılmıştır. Girdi imgelerine uygulanan yöntem sonucu elde edilen yüz bölgesi çıktıları Şekil 4.2'de gösterilmektedir.



Şekil 4.2. YGH + DVM ile gerçekleştirilen yüz tespit işlemi. a. Orijinal imge b. Orijinal imge üzerinde tespit edilen yüz bölgesi kırmızı çerçeve ile işaretlenmiştir.

Yüz bölgeleri elde edildikten sonra yüz özellik noktalarının elde edilmesi gerekmektedir. Yüz özellik noktalarını elde etmek için AGM tabanlı bir yaklaşım kullanılmıştır. 68 adet yüz özellik noktası içeren bir şablon, AGM yardımıyla her bir yüze ayrı olarak uydurulmuş ve 68 adet noktanın koordinatları belirlenmiştir. Şekil 4.3'de bir imge üzerinde gerçekleştirilen uydurma işleminin öncesi ve sonrasındaki yüz özellik noktalarının konumları gösterilmiştir. Özellik noktaları, uydurma işlemi sonrasında yüz anahtar öğeleri üzerinde konumlanmıştır.

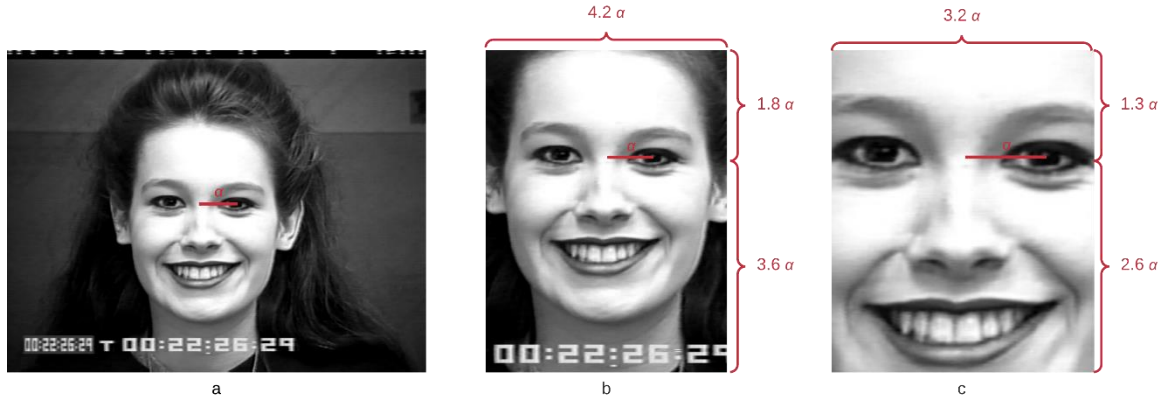
Bu uygulama kapsamında farklı boyutlarda kırılan yüz imgelerinin, YİT üzerindeki etkisi incelenmiştir. Bu kapsamda üç farklı ilgi bölgesi üzerinde çalışmalar gerçekleştirilmiştir. İlk ilgi

bölgesi, ölçek 1 olarak adlandırılmıştır. Ölçek 1, üzerinde herhangi bir işlem gerçekleştirilmeyen orijinal imgeleri içermektedir. Bu imgeler, eğitim açısından işleri zorlaştırabilecek olan kıyafet, saç, zaman damgası ve arka plan öğelerini içermektedir ve 640x490 piksel boyutlarındadır. İkinci ilgi bölgesi, ölçek 2 olarak adlandırılmıştır. Ölçek 2, göz, burun, ağız, çene ve kulakları içeren bütün yüz yapısını içermektedir ve ortalama 260x350 piksel boyutlarındadır. Üçüncü ilgi bölgesi, ölçek 3 olarak adlandırılmıştır. Ölçek 3, yüz ifadesini sergileyen en önemli bileşenler olan ağız, gözler ve burnu içermektedir ve ortalama 160x200 piksel boyutlarındadır.



Şekil 4.3. AGM kullanılarak uydurulan 68 adet yüz özellik noktası. a. Uydurma öncesi 68 adet yüz özellik noktasının yüz üzerindeki konumu b. Uydurma sonrası 68 adet yüz özellik noktasının yüz üzerindeki konumu. Özellik noktaları yüz anahtar öğelerini çevrelemektedir.

Ölçek 2 ve ölçek 3'ün kırılması için, her imgede referans olarak kullanılacak bir birim mesafeye ve pivot noktasına ihtiyaç vardır. Göz merkezleri arasındaki mesafenin yarısı olarak belirlenen bu birim mesafe " α " olarak adlandırılmıştır. Göz merkezleri arasındaki mesafenin hesaplanması için göz merkezlerinin koordinatlarının bilinmesi gereklidir. Fakat kullanılan şablonda göz merkezlerinin konumunu belirten bir nokta bulunmamaktadır. Bunun yerine göz alanını çevreleyen bir noktalar kümesi mevcuttur. Uygulama kapsamında, her bir gözü çevreleyen noktalar kümesinin ağırlık merkezi, göz merkezi olarak belirlenmiştir. Daha sonrasında göz merkezleri arasındaki mesafe ikiye bölünerek α birim mesafesi elde edilmiştir. Göz merkezleri arasındaki doğru parçasının orta noktası pivot olarak belirlenmiş ve pivot noktasından gerçekleştirilen α birim mesafesinin deneysel yöntemlerle elde edilen katları yardımıyla ölçek 2 ve ölçek 3 kırılmıştır. Deneysel olarak hesaplanan bu mesafelerin belirlenmesi için veri tabanından rastgele örnekler seçilmiş ve söz konusu alanları kapsamak için gereken α birim mesafesinin katları hesaplanmıştır. Ardından bu mesafelerin ortalamaları alınarak, ölçek 2 ve ölçek 3'te kullanılacak olan nihai ölçek katsayıları belirlenmiştir. Şekil 4.4'de örnek bir α mesafesinin gösterimi ile birlikte ölçek 2 ve ölçek 3'ün α parametresi ile hesaplanışına dair örnekler gösterilmiştir.



Şekil 4.4. Uygulama kapsamında tasarlanan üç adet ilgi bölgesi. Her bir ilgi bölgesinde kullanılan sabit birim mesafe α imge üzerinde gösterilmiştir. a. Ölçek 0, orijinal imge. b. Ölçek 1, genel yüz görüntüsü c. Ölçek 2, yüz ifadelerini sergileyen ana bölge

4.2.2. Ağların Hazırlanması ve Eğitimi

Bu kısımda, yüz ifadesini temsil eden yüz üzerindeki ilgili alanın seçiminin, YİT başarımına etkisi incelenmiştir. Farklı ağ mimarilerinin ilgili alanlara ilişkin sınıflandırma başarımlarını ölçmek amacıyla bu uygulamada birbirlerinden farklı üç adet ESA modeli kullanılmıştır. Bu modeller sırasıyla şunlardır: Önerilen 9 katmanlı bir model, Inception mimarisine dayalı Inceptionv3 modeli ve ResNet mimarisine dayalı ResNET-50 modeli.

Önerilen 9 Katmanlı Mimari

Önerilen 9 katmanlı modelin mimarisi yapısı Tablo 4.1’de verilmiştir. Girdi olarak görüntüler öncelikle 200x200 olarak boyutlandırılmıştır. Kullanılan bütün evrişim katmanlarında 3x3 filtre boyutu 1x1 kaydırma ile kullanılmıştır. Ayrıca çıktı haritasının boyutunun korunması için evrişim katmanlarında imgelere 1 piksel kenar ekleme yapılmıştır. ReLU fonksiyonu, evrişim katmanlarından sonra aktivasyon fonksiyonu olarak kullanılmıştır. Havuzlama katmanında maksimum havuzlama işlemi, 2x2 filtre boyutu ve 2x2 kaydırma ile gerçekleştirilmiştir. Kullanılan filtre boyutu ve kaydırma ile özellik haritalarının boyutları yarı yarıya indirilmiştir. Toplam 3 kez gerçekleştirilen evrişim ve havuzlama işlemleri sonrasında tam bağlı katmanlar kullanılmıştır. 2 adet tam bağlı katmanın kullanıldığı modelde her bir katman sonrasında aktivasyon fonksiyonu olarak ReLU ile birlikte Seyreltme (0,5) katmanı kullanılmıştır. Son olarak sınıflandırma için 6 sınıflı bir Softmax katmanı ile model sonlandırılmıştır.

Tablo 4.1. Önerilen 9 katmanlı modelin mimari yapısı.

Katmanlar	Boyut	Filtre/Kaydırma
Girdi Katmanı	200x200x1	
Evrişim_1 (ReLU)	200x200x32	3x3/1x1
Maksimum Havuzlama_1	100x100x32	2x2/2x2
Evrişim_2 (ReLU)	100x100x64	3x3/1x1
Maksimum Havuzlama_2	50x50x64	2x2/2x2
Evrişim_3 (ReLU)	50x50x128	3x3/1x1
Global Maksimum Havuzlama	1x1x128	
Tam Bağlı_1 (ReLU – Seyretme (0.5))	512	
Tam Bağlı_2 (ReLU – Seyretme (0.5))	512	
Softmax	6	

Inceptionv3 modeli

Bu uygulamada kullanılan ikinci model Inception mimarisini kullanan Inceptionv3 modelidir. Inception mimarisinin katmanları üst üste ekleyen geleneksel ağlardan farkı, aynı seviyede farklı katmanları paralel olarak kullanmasıdır. Inception modülü olarak adlandırılan bu bloklar, aynı katmanda verinin farklı özelliklerini yakalama konusunda başarılıdır. Inceptionv3 modelinin seçilmesinin nedeni, geleneksel yaklaşımdan farklı olan Inception modüllerinin YİT üzerindeki başarımını incelemektir. Çalışma kapsamında kullanılan Inceptionv3 modelinin sınıflandırıcısı olan 1000 sınıflı softmax katmanı, YİT problemine uyarlamak amacıyla, 6 sınıflı bir softmax katmanı ile değiştirilmiştir.

ResNet50 modeli

İlk uygulamada kullanılan son model ResNet mimarisinin bir üyesi olan ResNet-50 modelidir. ResNet mimarisi, derin ağlardaki gradyan kaybolması sorununa residual bağlantılar ile çözüm getirmektedir. Böylece gürbüz bir şekilde eğitilebilen daha derin ağların tasarlanmasına imkân tanımaktadır. Çalışma kapsamında Resnet-50 modeli kullanılarak, çok derin ESA'ların YİT üzerindeki etkisinin incelenmesi planlanmıştır. Çalışmada kullanılan ResNet-50 modelinin sınıflandırıcısı olan 1000 sınıflı softmax katmanı, YİT problemine uyarlanmak amacıyla, 6 sınıflı bir softmax katmanı ile değiştirilmiştir.

Girdi Katmanlarının Yeniden Boyutlandırılması

Eğitim aşamasına geçilmeden önce modeller üzerinde girdi boyutu düzenlemesi gerçekleştirilmiştir. Kullanılan ağların varsayılan girdi boyutları birbirlerinden farklıdır ve şu şekildedir: Önerilen ağ 200x200, Inceptionv3 ağı 299x299 ve ResNet ağı 224x224. Yüksek çözünürlüklü girdiler daha fazla detay içerebilmekte, bu da ağın özellik çıkarmasına pozitif olarak

katkıda bulunabilmektedir. Bu sebeple eğitim ve test esnasında ağların, girdi boyutuna yönelik bir avantaj sağlamamaları için girdi boyutları 200x200 olarak değiştirilmiştir.

Uygulanan Önişlemler

İmgelere, girdi olarak kullanılmadan önce birtakım önişlemler uygulanmıştır. Önişlem, makine öğrenmesi alanında kullanılan verinin ihtiyaç duyulan duruma getirilmesi için kullanılan işlemlerin tümüne verilen isimdir. Odaklanılan problem türüne göre verilere farklı önişlemler uygulanabilmektedir. Görüntü ön işleme, bilgisayarla görü kapsamında, sınıflandırıcıların beklenen şekilde çalışmaları için görüntü verilerini manipüle eden işlemlerin tümüne verilen genel bir isimdir. Hızlı özellik çıkarımı için kullanılan ESA’larda, verinin doğru bir şekilde girdi olarak verilmesi önem arz etmektedir.

Gerçekleştirilen ilk önişlem yeniden boyutlandırma’dır. Kullanılan ESA’ların girdi boyutuna göre, veri kümesindeki görüntülerin yeniden boyutlandırılması gerekebilir. Yeniden boyutlandırma, yakın komşu interpolasyonu, bilinear interpolasyon, biküçük interpolasyon, Fourier dönüşüm yöntemleri, Lanczos yeniden örnekleme, mipmap, kutu örnekleme vb. yöntemler ile gerçekleştirilebilmektedir. Çalışmada kullanılan üç farklı ölçeğe ait imgeler, verilerin hazırlanması bölümünde ifade edildiği üzere farklı çözünürlüklere sahiptir. Bu nedenle imgelerin aynı çözünürlüğe sahip olmaları için yeniden boyutlandırılmaları gerekmektedir. Boyutlandırma işlemleri için Lanczos yeniden örnekleme kullanılmıştır. Lanczos yeniden örnekleme, diğer boyutlandırma yaklaşımlarına nazaran daha yumuşak bir geçiş sunmaktadır.

Normalleştirme, merkezileştirme ve standardizasyon, eğitim aşamasından önce veri seti üzerinde uygulanabilecek üç önemli önişlemdir. Bu önişlemler, eğitim esnasında karşılaşılabilecek gradyan patlaması ve kaybolan gradyanlar sorunlarına yardımcı olmakla birlikte, ağın hızlı bir şekilde yakınsamasında önemli bir rol oynamaktadır.

Normalleştirme, verideki genlik değişimini gidermek ve verinin içerdiği dağılım şekline odaklanmak için kullanılmaktadır. Normalleştirme işlemi sonucunda veri [0,1] aralığında ölçeklenir. Verilen bir $X_{(i,j)}$ imgesinde, j kanalları, i ise bu kanaldaki imgeyi gösterebilir. (i,j) elemanı için [0, 1] aralığında normalleştirilmiş X_{norm} matrisi, Denklem 4.1’de gösterildiği şekilde hesaplanır. Bu işlem sonucunda her kanaldaki minimum piksel yoğunluk değeri 0, maximum piksel yoğunluk değeri ise 1 olmaktadır.

$$X_{norm(i,j)} = \frac{X_{(i,j)} - \text{minimum}(X_j)}{\text{maksimum}(X_j) - \text{minimum}(X_j)} \quad (4.1)$$

Merkezileştirme, normalleştirmeye benzer bir şekilde verinin genlik değişimini hedef alan bir önişlemdir. Merkezileştirme sonucunda veri [-1, 1] aralığında ölçeklenir. Verilen bir $X_{(i,j)}$ imgesinde, j kanalları, i ise bu kanaldaki imgeyi gösterebilir. (i,j) elemanı için [0, 1] aralığında

merkezleştirilmiş X_{merk} matrisi, Denklem 4.2'deki gibi hesaplanır. Bu işlem sonucunda her kanaldaki minimum piksel yoğunluk değeri -1, maximum piksel yoğunluk değeri ise 1 olmaktadır.

$$X_{merk(i,j)} = 2 * X_{norm(i,j)} - 1 \quad (4.2)$$

Standardizasyon, verinin dağılımının Gauss dağılımı olduğunu varsayarak, ortalamayı 0 standart sapmayı ise 1 olacak şekilde ayarlayan bir ön işlemdir. Ortalama çıkarma işlemi sonrasında elde edilen verinin, her bir kanal için kanalın standart sapması ile bölünmesi ile gerçekleştirilir. Verilen bir $X_{(i,j)}$ imgesinde, j kanalları, i ise bu kanaldaki imgeyi gösterebilir. (i,j) elemanı için standartlaştırılmış X_{sta} matrisi, Denklem 4.3'deki gibi hesaplanır. Bu işlem sonucunda her kanaldaki ortalama 0 ve standart sapma 1 olmaktadır.

$$X_{sta(i,j)} = \frac{X_{(i,j)} - ortalama(X_j)}{standartsapma(X_j)} \quad (4.3)$$

Çalışma kapsamında piksel değerlerini ölçeklemek için merkezleştirme yöntemi kullanılmıştır. Merkezleştirme sonucunda ağ eğitimin daha hızlı yakınsadığı gözlemlenmiştir. Yeniden boyutlandırma ve merkezleştirme aşamaları sonrasında veri, ağlarda eğitim için uygun hale getirilmiştir. Bu aşamadan sonra ağların eğitimine geçilmiştir.

Çapraz Doğrulama ve Eğitim Parametreleri

Eğitim işlemi, k katlamalı çapraz doğrulama yöntemi ile gerçekleştirilmiştir. Bu işlem için k değeri 8 olarak seçilmiştir. Bu kapsamda veri setleri, aynı deneğe ait imgelerin farklı gruplarda olmayacağı şekilde 8 gruba bölünmüştür. Her bir ağ, verilen bu gruplar kullanılarak 8 kez eğitilmiş ve test edilmiştir. Önerilen ağ ile birlikte Inceptionv3 ve ResNet-50 ağları eğitilirken kullanılan parametreler şu şekildedir:

- Yığın boyutu: 128
- Döngü sayısı: 50
- Girdi imgesi boyutu: 200x200
- Optimizasyon yöntemi: Adam (lr=0.001, beta_1=0.9, beta_2=0.999, epsilon=1e-07)

4.3. Çekişmeli Üretici Ağlar ile Örnek Artırımı

Gerçekleştirilen ikinci uygulamada, YİT başarımını artırmak üzere ÇÜA tabanlı bir örnek artırım yaklaşımı önerilmiştir. Bu iş için öncelikle kullanılan veri setindeki imgeler bir takım ön işlemden geçirilmiştir. Ardından bu imgelerin, kullanılan ÇÜA'daki temsilleri bulunmuştur. Aynı ifadeyi sergileyen farklı bireylerin yüz imgelerinin temsillerinin ortalamaları alınarak yeni yüz ifadeleri oluşturulmuştur. Önerilen yaklaşım ile aynı öz nitelik uzayında dağılımına sahip olacak bir şekilde

veri tabanı genişletilmiştir. Bu veri tabanı ile gerçekleştirilen eğitim ve test işlemleri sonucunda uygulanan yöntemin başarımı hakkında önemli bilgiler elde edilmiştir.

Derin Sinir Ağı (DSA) tabanlı yaklaşımlar, son yıllarda bilgisayarla görü alanında birçok uygulamada kullanım alanı bulmuşlardır. DSA'lar, geleneksel öznitelik çıkarımı yaklaşımlarının aksine geniş kapsamlı öznitelik çıkarımında başarılıdır. Bu nedenle DSA'lar, büyük öznitelik uzaylarında optimum çözümleri bulmak için kullanışlı araçlar olarak kabul edilmektedirler.

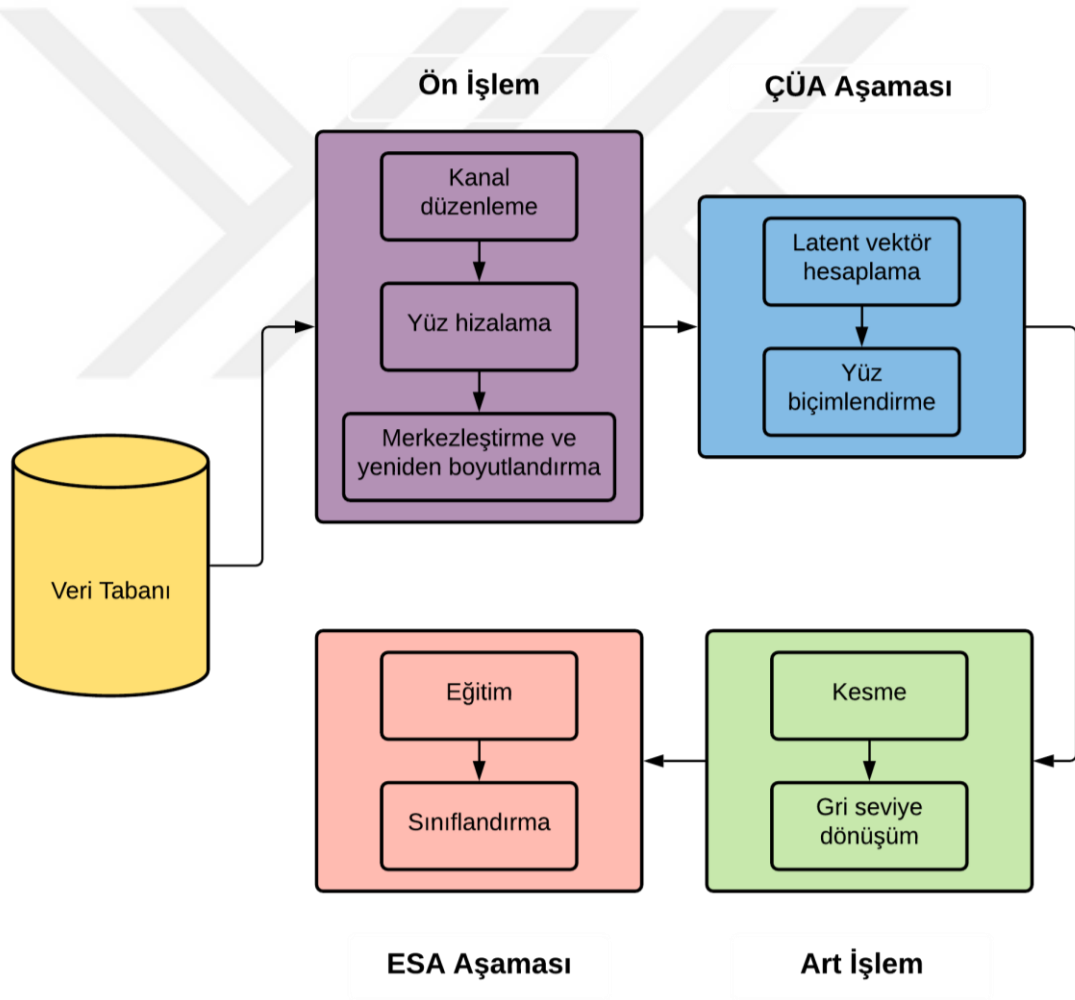
DSA'lar, birçok katman ve parametre içermektedirler. Bu tarz büyük mimarilerde aşırı öğrenmenin önüne geçmek için çok sayıda eğitim verisine ihtiyaç duyulmaktadır. Kontrollü ortamlarda elde edilen yüz ifade veri tabanları söz konusu olduğunda, bu ne yazık ki mümkün değildir. Zira bu veri tabanları kısıtlı sayıda süje içermektedirler. Az sayıda eğitim verisi ile gerçekleştirilen YİT sonucunda genelleştiremeyen bir DSA'nın elde edilmesi ise kuvvetle muhtemeldir. Genelleştiremeyen DSA'lar, eğitim esnasında gördüğü örnekleri başarılı bir şekilde sınıflandırabilmekte, fakat eğitim kümesi haricindeki örnekleri sınıflandırırken aynı başarımı gösterememektedirler. DSA eğitimlerinde aşırı öğrenmeyi kontrol altına adına L1 ve L2 normalizasyonu, yığın normalizasyonu, seyreltme ve öğrenme transferi gibi yöntemler literatürde sıklıkla kullanılmaktadır.

Yukarıda verilen tekniklerin aksine veri artırma yaklaşımları, aşırı öğrenmeyi çözmek adına veri setini doğrudan artırmayı öngörmektedir. Yardımcı eğitim verisi elde etmenin mümkün olmadığı durumlarda veri artırma yaklaşımlarına sıklıkla başvurulmaktadır. Genel olarak kullanılan veri artırma teknikleri, uzamsal uzayda çalışmalar ve temel görüntü işlemlerini (yakınlaştırma, kırpma, kaydırma, çevirme, beyazlatma, döndürme, renk uzayı manipülasyonları, rastgele silme vb.), çekirdek filtrelerini (bulanıklaştırma, keskinleştirme vb.) ve veri çarpıtmayı içerirler.

Yüz ifadeleri, yapıları gereği manipülasyonlara ve dönüşümlere duyarlıdır. Bu nedenle, uzamsal alandaki yüzlerin dönüştürülmesi beklenen sonuçları veremeyebilir. Örneğin, kırpma, yakınlaştırma ve kaydırma işlemleri sonucunda elde edilen yüz, ifadeyi sergileyen ağız, kaş ve göz gibi öğelerden yoksun olabilir. Buna benzer bir şekilde yüz üzerinde gerçekleştirilen elastik deformasyonlar, yüz ifadesini tamamen farklı bir ifadeye dönüştürebilir (örneğin ağzın, gözlerin ve kaşların yapısını değiştirilmesi). Ek olarak, uzamsal manipülasyon, verilerin özniteliksel temsilini hesaba katmaz. Bu nedenle, elde edilen örnekler, verilerden farklı bir özniteliksel temsile sahip olabilir ve bu da eğitim ve test süreçlerini olumsuz etkileyebilir.

Geleneksel veri artırma tekniklerinin bahsedilen eksikliklerini gidermek üzere, bu çalışmada YİT için ÇÜA tabanlı bir veri artırma yaklaşımı önerilmiştir. ÇÜA'lar, verilen girdi vektörlerini üretici ağlar vasıtasıyla imgelere haritalayan ağlardır. Bunu gerçekleştirirken uzamsal uzayda değil öznitelik uzayında çalışırlar. Bu sayede ÇÜA'lar, eğitildikleri veri kümesiyle aynı özniteliksel dağılıma sahip olan imgeler oluşturabilirler. Aynı öznitelik dağılımına sahip olan imgeler, veri setinin dağılımını daha iyi temsil edebilmekte ve bu imgelerle gerçekleştirilen ESA eğitiminin daha

gürbüz ve başarılı olması beklenmektedir. Bu varsayım ile gerçekleştirilen bu çalışmada, ÇÜA yaklaşımı ile az sayıda örnek içeren yüz ifade veri tabanlarında örnek artırımı ve bu örnekler kullanılarak ESA ile gerçekleştirilen YİT'in başarımının artırılması hedeflenmiştir. Çalışma kapsamında öncelikle veri tabanındaki imgelere birtakım ön işlemler uygulanmıştır. Bu işlemler sonrasında imgeler, ÇÜA aşaması için uygun hale getirilmiştir. ÇÜA aşamasında, ön işlemden geçirilmiş imgeler kullanılarak latent vektörler hesaplanmıştır. Aynı ifade sınıfında olup farklı kişilere ait olan latent vektörler ikili olarak kombine edilmiştir. Bu şekilde iki farklı görüntünün ortalaması bulunmuştur. Elde edilen ortalama yüzler, art işlem aşamasında kırılarak gri-seviye imgelere dönüştürülmüş ve Inceptionv3 ağı ile eğitilmiştir. Devamında ise orijinal imgeler kullanılarak ağın başarımı test edilmiştir. Gerçekleştirilen çalışmaya ait olan akış şeması Şekil 4.5'de verilmiştir.



Şekil 4.5. Tez kapsamında gerçekleştirilen ikinci çalışmaya ait akış şeması.

4.3.1. Gerçekleştirilen Ön İşlem Aşamaları

Gerçekleştirilen ön işlem adımları Şekil 4.6’da verilmiştir. Ön işlem, kanal düzenleme ile başlamaktadır. CK+ veri tabanındaki imgelerin çoğu gri-seviye olmakla birlikte RGB imgeler de mevcuttur. Doğru bir ön işleme için bütün imgelerin aynı renk uzayında olması gerekmektedir. Bu nedenle gri seviye renk uzayını hedef renk uzayı olarak belirlenmiş ve RGB görüntüler gri seviye görüntülere çevrilmiştir. RGB’den gri seviye dönüşümü için Denklem 4.6’da verilen ITU-R 601-2 luma dönüşümü kullanılmıştır.

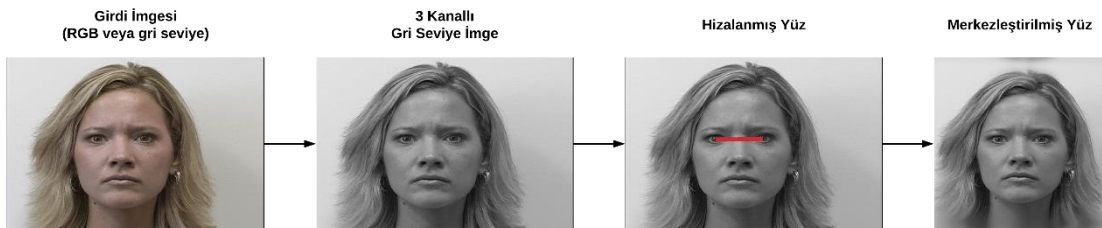
$$Çıktı = R * \frac{299}{1000} + G * \frac{587}{1000} + B * \frac{114}{1000} \quad (4.6)$$

Bu işlem sonrasında veri setindeki bütün imgeler, tek kanallı gri seviye imgelere dönüştürülmüştür. StyleGAN2 mimarisi, 3 kanallı girdi imgeleri üzerinde çalışmaktadır. Bu nedenle tek kanallı olan imgeler, yayınlama işlemi kullanılarak 3 kanallı imgelere dönüştürülmüştür.

StyleGAN2 mimarisi, x ekseninde hizalanmış ve ortalananmış 1024x1024 boyutunda yüz görüntüleri ile eğitilmiştir. İmgeleri başarılı bir şekilde latent uzaya gömmek için, StyleGAN2 eğitim örneklerine uygulandığı şekilde yüz hizalama ve merkezleştirme gerçekleştirilmiştir. İlk olarak HOG yaklaşımı yardımıyla yüzler lokalize edilmiştir. Yüzler lokalize edildikten sonra iBUG 300-W kullanılarak 68 yüz noktası koordinatı tespit edilmiştir. Her bir gözü kapsayan noktalar kümesinin ağırlık merkezi, göz merkezi olarak varsayılmıştır. Belirlenen göz merkezleri kullanılarak, Denklem 4.7’de gösterildiği şekilde iki göz merkezini birleştiren doğru parçasının, x eksenine ile yaptığı açı hesaplanmıştır. Devamında bu açı yardımıyla, göz merkezleri x ekseninde hizalanacak şekilde görüntü döndürülmüştür.

$$açı = \tan^{-1} \frac{\sum_{n=43}^{48} y_n - \sum_{n=37}^{42} y_n}{\sum_{n=43}^{48} x_n - \sum_{n=37}^{42} x_n} \quad (4.7)$$

Burada açı, göz merkezlerini birleştiren çizginin x eksenine ile yaptığı açı, x_n ile y_n ise n -inci yüz işaret noktasının x ve y koordinatlarıdır. Yüzler hizalandıktan sonra yüz bölgeleri imgelerde ortalananmış ve Lanczos yeniden örnekleme kullanılarak 1024x1024 boyutunda yeniden boyutlandırılmıştır.



Şekil 4.6. Gerçekleştirilen ön işlem adımları

4.3.2. Çekişmeli Üretici Ağ Yardımıyla Gerçekleştirilen Örnek Artırımı

Çalışma kapsamında StyleGAN2 ÇÜA'sı kullanılarak örnek artırımı yoluna gidilmiştir. StyleGAN ağının halefi olan StyleGAN2 ağı, sentetik yüz üretiminde geline en son noktayı temsil etmektedir. Ağın bu konudaki başarımı, eğitimsiz gözlerin orijinal ve sahte imgeleri birbirlerinden ayırt edemeyeceği düzeydedir.

StyleGAN2, diğer ÇÜA'lara benzer bir şekilde üretici ağ ve ayrıştırıcı ağ olmak üzere iki ana bileşenden oluşur. Eğitim esnasında üretici ağın ürettiği sahte imgeler ile veri setine ait olan orijinal imgeler, ayrıştırıcı ağ yardımıyla karşılaştırılır. Hesaplanan hata neticesinde geri yayılım ile üretici ağ ve ayrıştırıcı ağ ayrı ayrı eğitilir. Bu eğitim süreci, ayrıştırıcı ağın, sahte imgeler ve orijinal imgeleri birbirlerinden ayırt edemeyeceği duruma kadar devam eder.

Eğitimi tamamlanan üretici ağ, girdi olarak verilen bir latent vektör yardımıyla eğitildiği veri setine benzer imgeler üretebilmektedir. Üretilen veri, latent vektör üzerindeki değişkenler ile kontrol edilir. Hangi değişkenlerin, çıktı imgesinde hangi alanlar üzerinde değişikliğe sebep olduğunu araştıran çalışma konusuna “öznitelik dolanması” adı verilir. Öznitelik dolanması sayesinde farklı vektörel bileşenlerin, çıktı imgesi üzerindeki etkileri kategorize edilmiş olur. StyleGAN2 ağı için, gülümseme, yaş, cinsiyet, ten rengi gibi öğrenilen vektör yönleri kullanılarak bir imge üzerinde değişiklikler yapılabilir. Tabii ki bunun öncesinde söz konusu imgenin latent vektörünün bilinmesi gerekmektedir. Değişiklikler, imgenin latent vektörüne, öğrenilen latent vektörlerin eklenip çıkarmasıyla gerçekleştirilir.

Latent vektörler ile gerçekleştirilebilecek bir diğer işlem ise latent aritmetiğidir. Latent aritmetiği sayesinde farklı imgeler arasında interpolasyona benzer şekilde geçişler yapılabilir. Bu sayede iki farklı imgenin ortalamasından oluşan ortanca bir imge elde edilebilir. Fakat bu işlem için de söz konusu imgelerin latent vektörlerinin bilinmesi gerekmektedir. Zira bu interpolasyon, uzamsal uzayda değil latent uzayında gerçekleştirilir.

Bir imgenin, latent vektörünün elde edilmesi için latent uzaya gömülme işlemine “ÇÜA tersine çevirimi” adı verilir. Tersine çevirim işlemi genel olarak; verilen imgenin hatasını minimize etmek için latent vektörün doğrudan optimize edilmesi ile [111], verilen imgeyi latent uzaya haritalamak için bir kodlayıcı eğitilerek [112] veya bu iki yaklaşımın birlikte kullanıldığı hibrit yöntemler kullanılarak [113] gerçekleştirilir.

Gerçekleştirilen çalışma kapsamında, ÇÜA tersine çevirimi için latent vektörün doğrudan optimize edildiği bir yaklaşım kullanılmıştır. Bu iş için ilk olarak latent vektörü bulunacak olan imge R ile birlikte z_G latent vektörü yardımıyla üretilen imge G 'ye ihtiyaç vardır. R ve G imgeleri arasındaki fark, algısal kayıp (perceptual loss) için önceden eğitilmiş olan bir ESA yardımıyla hesaplanır. Algısal kayıp, mevcut ve hedef imgeler arasındaki kaybı, ağın çıktısı olan yüksek seviye imge özelliklerini kullanarak hesaplar. Kayıp hesaplandıktan sonra geri yayılım yardımıyla z_G güncellenir. Bu işlem esnasında üretici ağ ve algısal kayıp ağı üzerinde herhangi bir güncelleme

yapılmaz. Algısal kaybı azaltmak için güncellenen z_G latent vektörü ile üretilen G imgesi, R imgesine yakınsar. Sonuç olarak, R imgesinin Z latent uzayındaki latent vektörü olan z_R elde edilir. z_R latent vektörü kullanılarak yukarıda bahsedilen latent uzay manipülasyonları ve latent aritmetik işlemleri, G imgesinin üzerinde gerçekleştirilebilir.

Uygulama kapsamında üretici ağ olarak StyleGAN2 üretici ağı, algısal kaybın hesaplanması için ise StyleGAN2 ağı üzerinde algısal kayıp için eğitilen VGG16 ağı kullanılmıştır. Latent optimizasyon işlemi toplam 1000 adımda gerçekleştirilmiştir. Latent optimizasyonu sonrasında elde edilen latentler kullanılarak, orijinal imgelerin StyleGAN2 uzayındaki karşılıkları elde edilmiştir. Şekil 4.7’de CK+ veri tabanından alınan örnek yüz imgeleri ve bu imgelerin StyleGAN2 latent uzayındaki karşılıkları verilmiştir. Şekilde de görüleceği üzere, yüz ifadesi ile ilgisi olmayan arka plan ve zaman damgaları gibi detaylar, üretici tarafından dikkate alınmamıştır. Zira üretici, yüz üretmeye ve şekillendirmeye yarayan bileşenlerden oluşmaktadır.



Şekil 4.7. CK+ veri tabanından alınan imgeler ve bu imgelerin StyleGAN2 latent uzayındaki karşılıkları.

Orijinal imgelerin latent vektörleri elde edildikten sonra latent aritmetiği yöntemleri ile örnek artırımına gidilmiştir. Latent aritmetiğine örnek vermek adına, V uzayında tanımlı v_1 ve v_2 vektörlerini ele alalım. Bu iki vektörün ortalaması, vektör aritmetiği yardımıyla $(v_1 + v_2)/2$ olarak

hesaplanır. Bu çalışmanın örnekleminde ise Z latent uzayındaki z_1 ve z_2 latent vektörlerinin ortalaması kullanılarak üretici tarafından üretilen yüz, ortalama yüzü verir. Ortalama yüz, kendisini oluşturan yüzlerin özelliklerini eşit olarak yansıtır. Çalışma kapsamında, aynı ifade sınıfına ait yüzler kullanılarak ortalama yüzler elde edilmiş ve ilgili ifade sınıfında örnek artırımına gidilmiştir. Bunun için bir ifade sınıfının altındaki her bir yüz imgesinin, aynı ifade sınıfı altındaki geri kalan her bir yüz imgesi ile ortalamaları hesaplanmıştır. Bu şekilde her bir ifade sınıfı için, n ilgili sınıftaki örnek sayısı olmaz üzere n 'nin ikili kombinasyonu olan $C(n, 2)$ adet yeni örnek üretilmiştir. Farklı sınıflar için latent vektör aritmetiği ile üretilen örnekler Şekil 4.8'de verilmiştir. Söz konusu şekilde üst ve alt satırlar latent vektörler ile üretilen yüzleri, ortanca satır ise bu latent vektörlerin ortalaması alınarak üretilen yüzleri göstermektedir. Bu işlem için üst ve alt satırdaki yüzler %50-%50 oranında kullanılmıştır. Her bir yüz ifade sınıfı için baz örnek sayısı ve artırılmış örnek sayısı Tablo 4.2'de verilmiştir. Sınıf içi imgelerin kombinasyonu kullanılarak toplam 415 adet örnekten 14693 örnek üretilmiştir. Şunu belirtmek gerekir ki, vektör aritmetiği yardımıyla sadece iki yüz imgesinin ortalamasının elde edilmesinin yanı sıra, çok sayıda yüz ifadesinin farklı oranlarda sentezlenmesi de mümkündür. Bu şekilde, üretilen yüz ifade sayısından çok daha fazlası da sentezlenebilir. Çalışma kapsamında $C(n, 2)$ olarak gerçekleştirilen örnek üretimi, uygulama için yeterli imge sayısını sağlamıştır.



Şekil 4.8. Farklı sınıflar için latent vektör aritmetiği ile üretilen örnekler.

Tablo 4.2. CK+ veri tabanındaki her bir sınıf için baz örnek ve artırılmış örnek sayıları

İfade	Baz Örnek Sayısı (n)	Artırılmış Örnek Sayısı (C(n,2))
Kızgın	45	990
İğrenme	59	1711
Korku	25	300
Mutlu	69	2346
Üzgün	28	378
Şaşırma	83	3403
Nötr	106	5565
Toplam	415	14693

Art İşlemler

ÇÜA yardımıyla oluşturulan örneklerle, ESA eğitimi için bir takım art işlem uygulanmıştır. Hızlı bir şekilde özellik çıkarmak için kullanılan ESA'lar, özellik çıkarımını verilen girdi imgesinin bütününe bakarak gerçekleştirirler. Gürbüz bir özellik çıkarımı işleminin gerçekleştirilmesi için ise, problem uzayı ile alakasız özelliklerin bu uzaydan ayrıştırılması gerekmektedir. Ağız, gözler ve kaşlar, bir yüz ifadesini sergileyen temel bileşenlerdir. Yüzün geri kalan morfolojisi (kafa türü, burun türü, çene hattı ve deri rengi) dikkate alınmadan, temel bileşenler yardımıyla YİT yüksek başarımla gerçekleştirilebilir. Bu da ağız, göz ve burnun, problem uzayıyla ilişkili olduğu anlamına gelmektedir. Arka plan, zaman damgası, saç, gözlükler ve kıyafetler ise YİT için önemsiz bileşenlerdir. Bu bileşenlerden kaçınmanın başarıma pozitif olarak katkı sağlaması muhtemeldir. Bu nedenle art işlem aşamasında, ilk olarak YİT ile ilgili olan yüz bölgesinin tespiti gerçekleştirilmiştir.

Yüz ifadesini temsil eden ilgili bölgenin seçimi önemlidir. Bu alanda gerçekleştirilen çalışmaların çoğunda ve tez kapsamında gerçekleştirilen ilk çalışmada, göz merkezlerini arasındaki mesafeyi temel alarak gerçekleştirilen bir kırpma yaklaşımı kullanılmaktadır [8,114]. Bu çok sağlıklı bir işlem değildir zira yüz anatomisi, cinsiyete ve irksal koşullara bağlı olarak bireyler arasında büyük farklılıklar gösterebilmektedir. Örneğin, beyaz ırk ve çekik gözlü ırklarda, yüzün fizyonomik yüksekliği erkek ve dişiler arasında değişkendir. Yüz genişliği açısından bakılacak olursa, kadınların ortalama yüz genişliği 120 mm ve 140mm aralığında iken erkeklerde bu değerler 115 mm ve 160 mm aralığındadır [115]. Ek olarak sabit bir mesafenin kullanılması, şaşkınlık ve korku gibi ağız bölgesinin normalden daha fazla yer kapladığı ifadelerin kırpılmasında beklenmedik sonuçlara yol açabilir. Irklar arası yapılan çalışmaların sonuçlarında dayanarak, bu çalışma kapsamında yüz ifadesi içeren alanın seçimi için yüz özellik noktalarını içeren bir yaklaşım kullanılmıştır. Bu kapsamda ağız, gözleri ve kaşları çevreleyen dikdörtgen bölge, yüz ifadesi ile ilgili alan olarak seçilmiştir. Özellik noktalarının tespiti esnasında detektörün hata oranı da göz

önüne alınarak, işaret noktalarına 5 piksellik kenar boşluğu eklenmiştir. Bu kenar boşluğu, yüz elemanları ve imgenin kenarları arasındaki bitişikliği önlemek için ayrıca bir tampon görevi görmektedir. Şekil 4.9’da üretici ağ ile elde edilen yüz imgeleri ve bu imgelerin bahsedilen yöntem ile kırılması sonucu elde edilen imgelere ait örnekler verilmiştir.



Şekil 4.9. Üretilen imgeler ve art işlem sonucunda elde edilen yüz ifadesi ile ilgili alanlar.

StyleGAN2 ağına girdi olarak verilen görüntüler 3 kanallı gri seviye görüntülerdir. Üretici ağı ürettiği imgeler, yine aynı şekilde 3 kanallı bir yapıya sahiptir. Son art işleme aşaması olarak söz konusu imgeler eğitilecek olan ESA yapısına uygun olması için tek kanallı gri seviye imgelere dönüştürülmüştür. Bu son işlemten sonra imgeler, belirlenen ESA üzerinde eğitim için uygun hale getirilmiş olur.

Çapraz Doğrulama ve Eğitim Parametreleri

Adil bir ölçüm gerçekleştirmek adına veri tabanı, gruplar arasında bireyler bazında çakışma olmayacak şekilde 8 gruba ayrılmıştır. Bu, bir süjenin bütün imgelerinin aynı grup altında bulunması anlamına gelmektedir. Bu tarz bir yaklaşım ile test gruplarındaki süjelerin eğitim gruplarında bulunmaması sağlanmış olur. Söz konusu yaklaşım ile sınıflandırıcı genelleştirilebilirliğinin artırılabilirliğini belirten çalışmalar mevcuttur [116]. Genelleştirilebilir bir sınıflandırıcı, daha önce görmediği örnekler üzerinde, aşırı öğrenmiş bir sınıflandırıcıdan daha iyi sonuçlar vermektedir.

Çalışma kapsamında ESA olarak Inceptionv3 modeli kullanılmıştır. Söz konusu ESA’nın eğitim ile ilgili parametreleri Tablo 4.3’de verilmiştir. Girdi imgesi boyutu 128x128 piksel olarak seçilmiştir. Bu kapsamda eğitim için kullanılacak olan üretilmiş görüntüler ve test olarak kullanılacak olan orijinal görüntüler 128x128 olacak şekilde Lanczos yeniden örnekleme ile yeniden boyutlandırılmıştır. Eğitim işlemi, k katlamalı çapraz doğrulama yöntemi ile gerçekleştirilmiştir. Bu işlem için k değeri 8 olarak seçilmiştir. Her bir eğitim sürecinde, bir grup test için geri kalan gruplar ise eğitim için kullanılmıştır. Bu şekilde çalışma boyunca toplam 8 kez

ağ eğitimi gerçekleştirilmiştir. Her bir eğitim süreci sonunda ilgili grubun test imgeleri kullanılarak ağların başarımı incelenmiştir.

Tablo 4.3. Kullanılan Inceptionv3 mimarisi için eğitim parametreleri

Model	Girdi Boyutu	Yığın Boyutu	Döngü Sayısı	Optimize Edici	Öğrenme Oranı	Beta_1	Beta_2	Epsilon
Inceptionv3	128x128	128	100	Adam	0.001	0.9	0.999	1e-07

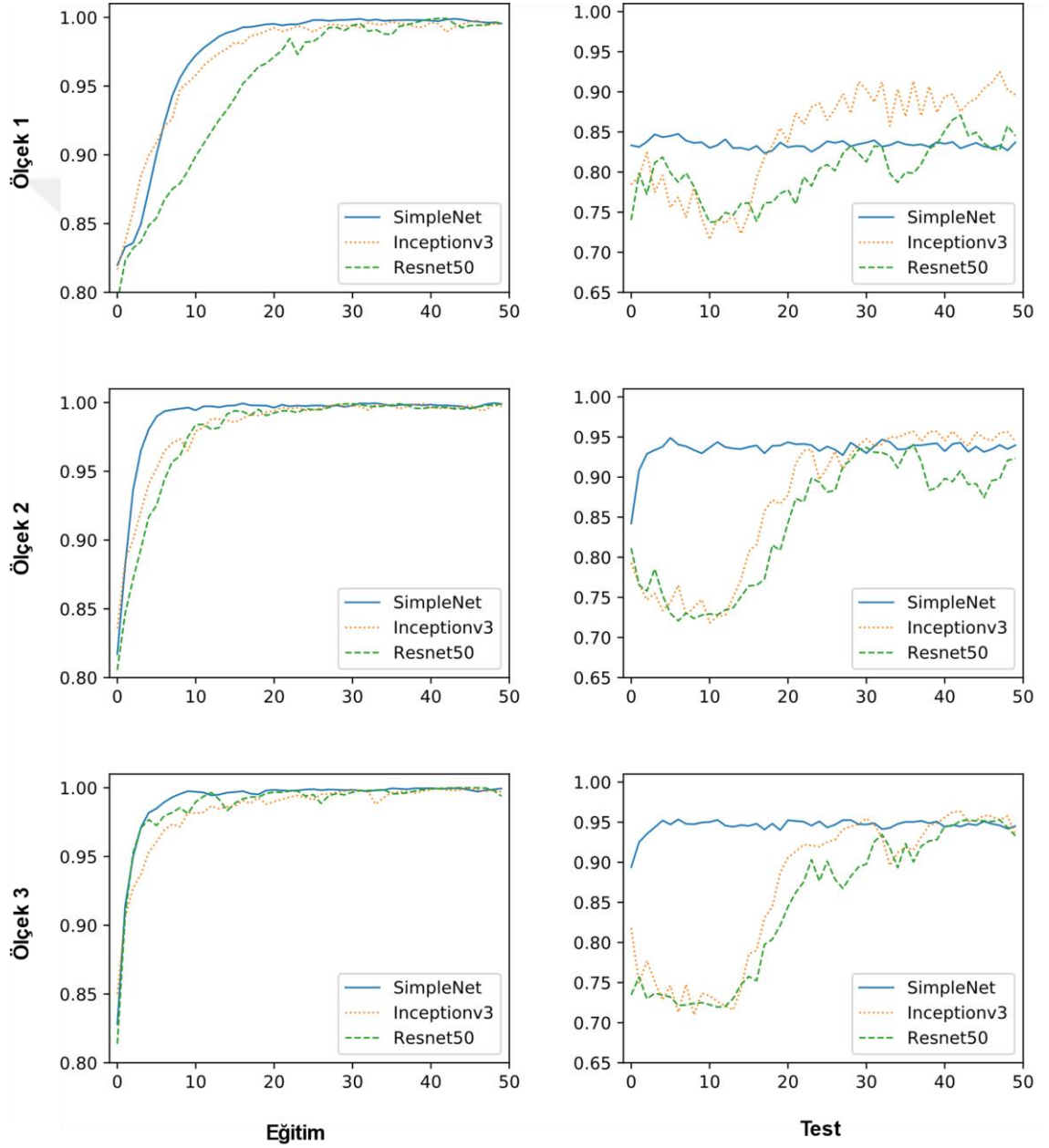
İşlem Süreleri

Ön işlem adımları (kanal düzenleme, yüz hizalama, merkezileştirme ve yeniden boyutlandırma), her imge için 720 ms olmak üzere toplam 415 imge için 298 saniye sürmüştür. 1000 iterasyon ile imgelerin latent uzayına gömülmesi, her bir imge için ortalama 3 dakika sürmüştür. Her ne kadar uzun bir süreç gibi gözüксе de bu işlem her bir imge için sadece bir kere gerçekleştirilmektedir. Latent vektörlerden imge oluşturulması ise bundan çok daha hızlı bir şekilde, imge başına 510 ms sürmüştür. Veri artırımı latent aritmetiği ile gerçekleştirilmiştir. Ortalama vektörün hesaplanması, bu vektörlerden imgelerin oluşturulması ve depolama birimine kaydedilmesi, toplam 14693 imge için 2 saat 9 dakikada sürmüştür.

5. BULGULAR

5.1. Evrişimsel Sinir Ağları Kullanılarak Gerçekleştirilen Yüz İfade Tespitine Dair Bulgular

Çalışmanın bu bölümünde, ESA kullanılarak gerçekleştirilen YİT'e dair bulgular sunulacaktır. Şekil 5.1'de her bir ağ üzerinde eğitilen 3 ölçeğe ait eğitim ve test başarımları verilmiştir.



Şekil 5.1. Bütün ölçeklerde gerçekleştirilen ESA eğitimlerinin, eğitim ve test başarımları. Grafiklerde yatay eksen döngü sayısını, dikey eksen ise başarımları göstermektedir.

Tablo 5.1’de ağların farklı ölçeklerdeki ortalama eğitim ve test başarımları verilmiştir. Bu tablo incelendiğinde, ilgili alan olabilecek en küçük şekilde seçildiği takdirde (gözleri ve ağzı kapsayan ölçek 3) bütün ağların hemen hemen birbirlerine eş değer başarımlar sergiledikleri gözlemlenmektedir. Fakat bu başarımlar, ölçek 1 ve ölçek 2 için söz konusu değildir. Bu ölçeklerde daha derin olan ağlar, daha basit olan ağlara nazaran daha fazla başarımlar göstermektedir. Söz konusu ölçeklerde Inceptionv3, ResNet-50 ve önerilen basit ağdan daha fazla, ResNet-50 ise önerilen basit ağdan daha fazla başarımlar göstermektedir. Bu ise daha derin ve kompleks ağların, daha sık olan ağlara nazaran imge üzerindeki varyasyonların üstesinden daha başarılı bir şekilde geldiğini vurgulamaktadır.

Tablo 5.1. Kullanılan ağların her bir ölçek için ortalama başarımları

	Ölçek 1	Ölçek 2	Ölçek 3
Önerilen Ağ	0.8447	0.8945	0.9706
Resnet-50	0.8945	0.9609	0.9714
Inceptionv3	0.9611	0.9743	0.9804

Tablo 5.2’de ağların, farklı ölçeklerde bir döngü boyunca eğitimi için gereken ortalama süre saniye cinsinden verilmiştir. Bu tablo incelendiğinde, daha büyük ilgili alanların daha fazla işlem yükü gerektirdiği görülmektedir. İlgili alan büyüdükçe önerilen ağın ve Inceptionv3 ağının eğitim süreleri artış göstermektedir. ResNet ise her ölçekte aynı eğitim süresine sahip olarak ayrılmaktadır. Eğitim süresindeki bu artış, büyük ilgili alanların küçük ilgili alanlara oranla daha fazla detay içermesinden dolayıdır. Açıklamak gerekirse, ölçek 3, ölçek 1’e nazaran daha az yüzey varyasyonu içermektedir. Zira ölçek 3 sadece ağzı ve gözleri kapsamakta, ölçek 1 ise saç, alın ve arka plan gibi bir takım değişken özellikleri içermektedir. Bu ek bilgiler, ağların eğitimini güçleştirmekte ve ayrıştırma yeteneklerine ket vurmaktadır.

Tablo 5.2. Farklı ölçeklerde bir döngü boyunca ağ eğitimi için gereken saniye cinsinden ortalama süre

	Ölçek 1	Ölçek 2	Ölçek 3
Önerilen Ağ	3,4	1,3	1,1
Inceptionv3	5,9	5,2	5,2
ResNet-50	8,1	8,1	8,1

Tablo 5.3’de, CK+ veri seti üzerinde 6 sınıflı YİT için en yüksek başarımları sergileyen ölçek 3’e ait hata matrisi verilmiştir. Tablodan görüleceği üzere mutlu ve şaşırma yüz ifadeleri, ağlar tarafından kolaylıkla ayırt edilebilmektedir. Bunun sebebi, mutlu yüz ifadesinde “U” ve şaşırma yüz

ifadesinde “O” şeklinde karakterize edilen ağız yapısıdır. Söz konusu şekilde sergilenen ağız yapıları, ağlar tarafından, diğer ifadelere oranla daha kolay ayrıştırılabilmektedir. Öte yandan diğer yüz ifadeleri, bazı durumlarda birbirileri ile karıştırılabilmektedir. Buna örnek olarak gözlerin kısık olduğu kızgın ve iğrenme ifadeleri veya ağzın daha kapalı olduğu üzgün ve iğrenme ifadeleri verilebilir.

Tablo 5.3. En yüksek başarıyı sağlayan ölçek 3 için 6 ifadeyi içeren hata matrisi.

	Kızgın	İğrenme	Korku	Mutlu	Üzgün	Şaşırma
Kızgın	43	-	-	1	-	1
İğrenme	1	58	-	-	-	-
Korku	-	1	23	1	-	-
Mutlu	-	-	-	69	-	-
Üzgün	-	1	1	-	25	-
Şaşırma	-	-	-	-	-	83

5.2. Çekişmeli Üretici Ağlar Yardımıyla Örnek Artırımı Sonrasında Gerçekleştirilen Yüz İfade Tespitine İlişkin Elde Edilen Bulgular

Çalışmanın bu bölümünde ÇÜA’lar kullanılarak artırılan örnekler ile gerçekleştirilen YİT’e ilişkin bulgular verilecektir. Uygulamalar başlığı altında bahsedildiği üzere veri seti, k katlamalı çapraz doğrulama gereğince süje örtüşmesi olmayan 8 eşit parçaya bölünmüştür. Her bir ağ eğitiminde, bir grup test için, geri kalan gruplar ise eğitim için kullanılmıştır. Eğitim için, ÇÜA yardımıyla oluşturulan imgeler kullanılırken, test için orijinal imgeler kullanılmıştır. Bu tarz bir yaklaşım ile veri sızıntısı engellenmiş olmaktadır. Zira ÇÜA üreticisi, yüzleri oluştururken latent uzay dağılımından faydalanmaktadır. Eğitim ve test imgelerinin aynı dağılımdan örneklenmesi, bir noktada ESA’nın bu dağılıma odaklanmasını mümkün kılmaktadır. Bu ise arzu edilen bir davranış biçimi değildir. Test için orijinal imgeler kullanılarak, ÇÜA’nın yüzleri oluştururken öğrenmiş olduğu dağılımın test imgelerine geçişi engellenmiştir.

Çalışma boyunca toplam 8 kez ağ eğitimi gerçekleştirilmiştir. Her bir grubun eğitim süreci sonrasında test imgeleri için başarımları ve elde edilen ortalama başarımları Tablo 5.4’de verilmiştir. Grupların başarımları %95,2 ile %98,4 aralığında olmakla beraber ortalama başarımları %97,2 düzeyindedir. Burada dikkat edilmesi gereken bir nokta sekizinci grubun diğer gruplara nazaran daha kötü bir performans sergilemesidir. Gruplar incelendiğinde sekizinci grubun, diğer gruplardan farklı olarak renkli imgeler de içerdiği gözlemlenmiştir. Bu başarımların düşüşünün, gri seviye dönüşümü esnasında imgelerin birtakım özelliklerini kaybetmeleri ile ilişkili olduğu düşünülmektedir.

Tablo 5.4. Her bir grup için elde edilen en yüksek başarımlar ve genel ortalama başarımlar

Grup No	1	2	3	4	5	6	7	8
Başarımlar	%96,9	%96,5	%98,3	%98,1	%98,4	%98,1	%96,2	%95,2
Ortalama Başarımlar	%97,2							

5.3. Sonuçların Literatürdeki Diğer Çalışmalar ile Karşılaştırılması

Bu kısımda, gerçekleştirilen iki uygulamaya ait sonuçlar, literatürdeki diğer sonuçlar ile karşılaştırılmıştır. CK+ veri tabanı üzerinde, 6 ve 7 yüz ifade sınıfı (6 ifade sınıfı: kızgın, öğrenme, korku, mutlu, üzgün ve şaşırma, 7 ifade sınıfı: 6 ifade sınıfına ek olarak nötr yüz ifadesi) üzerinde literatürde gerçekleştirilen çalışmalarla birlikte tez kapsamında gerçekleştirilen çalışmaların başarımları Tablo 5.5’de verilmiştir.

Adil bir karşılaştırma olması adına, süje örtüşmesinin olmadığı ve her sınıflandırma için sadece tek bir imgenin kullanıldığı çalışmalar karşılaştırma tablosuna dahil edilmiştir. Eğitim ve test kümelerinde aynı anda bulunan süjeler, aşırı öğrenme ile başarımları artırmaktadır. Fakat daha önce görülmeyen örnekler kullanıldığında ağ başarımları beklenenden düşük olmaktadır. Bu nedenle, adil bir karşılaştırma için süje örtüşmesinin elimine edilmesi gereklidir. Uygulamalar kısmında yer verilmemiş olsa da doğrulama amacıyla süje çakışması ile gerçekleştirilen eğitim ve test işlemleri sonucunda %99’a yakın başarımlar elde edilmiştir. Bu nedenle, açık ve net bir şekilde süje çakışması olmadığını belirtmeyen veya bu konuda tutarsız açıklamalara sahip olan çalışmalar listeye dahil edilmemiştir.

Bir yüz ifadesi imgesi, bir yüz görüntüsünden anlık olarak yakalanan imge örneği olarak tanımlanabilir. Yüz ifade sekansı ise, ardışık olarak birleştirilen yüz ifade imgeleri ile oluşturulur. Tez çalışması kapsamında, YİT için statik yüz imgeleri kullanılmıştır. Buna rağmen literatürde, yüz ifade sekansları ile YİT gerçekleştiren çalışmalar da mevcuttur. Bu çalışmalar, yüz imgeleri ile birlikte zamansal bilgiyi de kullanarak daha başarılı sonuçlar verebilmektedirler. Yüz ifade sekansları üzerinde gerçekleştirilen çalışmaların avantajlı konumlarından dolayı bu çalışmalar, karşılaştırma tablosuna dahil edilmemiştir.

Tablo 5.5’de, tez kapsamında gerçekleştirilen ilk çalışma ESA, ikinci çalışma ise ÇÜA olarak adlandırılmıştır. Önerilen ESA çalışması 6 yüz ifade sınıfı, ÇÜA çalışması ise 7 yüz ifade sınıfı üzerinde gerçekleştirilmiştir. Tablo 5.5’deki diğer çalışmalar ile karşılaştırıldığında ESA çalışması, başarımlar olarak kendisine en yakın çalışmadan %1,28 daha yüksek başarımlar göstermiştir. Yine aynı tablo incelendiğinde, ÇÜA çalışmasının, kendisine en yakın çalışmadan %0,17 daha fazla başarımlar sergilediği görülmektedir.

ESA çalışması kapsamında herhangi bir veri artırımı uygulanmamıştır. Bu çalışma, farklı ağların ve farklı ölçeklerin sınıflandırma başarımı üzerindeki etkilerini incelemek amacıyla gerçekleştirilmiştir. Rastgele bir şekilde gerçekleştirilecek olan veri artırımının, bazı ölçeklerde sınıflandırma performansında avantaj sağlayabileceğinden ötürü böyle bir yol izlenmiştir. Temel veri artırım yaklaşımlarıyla, ESA çalışmasındaki en iyi başarımın bir miktar daha artırılacağı öngörülebilir.

ÇÜA çalışması, kendisine en yakın çalışmalardan biri olan Lopes vd. [8] tarafından gerçekleştirilen çalışmadan daha yüksek başarımlar göstermiştir. Söz konusu çalışma, gauss gürültüsü ile göz konumlarını değiştirerek sentetik yüzler oluşturmaktadır. Bu yaklaşım, her ne kadar veri artırımını sağlasa da oluşturulan örnekler, aynı kişilere ait ve göz harici diğer yüz elemanlarında değişiklik içermeyen örneklerdir. ÇÜA çalışması ise, veri artırımı için her iki süjeye ait imgelerin, yüz özellikleri, renk ve konum gibi bilgilerinin ortalamasını alarak yeni imgeler sentezler. Bu nedenle yüz özellik uzayını temsil etmede daha başarılıdır. Bu yüksek temsil kabiliyetinden ötürü eğitilen ESA ağı ile gerçekleştirilen test işlemi, diğer çalışmalara oranla daha yüksek başarımlar sergilemiştir.

Tablo 5.5. Gerçekleştirilen çalışmaların literatürdeki diğer çalışmalar ile karşılaştırılması

Çalışmalar	6 İfade Sınıfı	7 İfade Sınıfı
Fan ve Tjahjadi [117]	83,70	-
Zavaschi vd. [118]	-	88,90
Rivera vd. [119]	-	89,30
Gu vd. [120]	91,51	-
Zhong vd. [121]	93,30	-
AUDN [122]	93,70	-
Liu vd. [123]	94,19	-
Lee vd. [124]	94,90	-
LBP+SVM [75]	95,10	91,40
IntraFace [125]	96,40	-
BDBN [126]	96,70	-
Jain vd. [127]	-	92,71
Jain vd. [128]	-	93,24
Lopes vd. [8]	96,76	95,75
Yang vd. [129]	-	96,57
Li vd. [130]	-	97,03
ESA	98,04	-
ÇÜA	-	97,20

6. SONUÇLAR VE DEĞERLENDİRME

Bilgisayarlar tarafından gerçekleştirilen otomatik YİT'e ilişkin çalışmalar, teknolojinin de buna uygun hale gelmesiyle geçtiğimiz son on yılda büyük ivme kazanmıştır. YİT'e ilişkin uygulamalar, başta insan bilgisayar etkileşimi olmak üzere, klinik psikoloji, nöroloji ve hasta takibi olmak üzere sağlık alanlarında, müşteri memnuniyeti, reklam geri dönüşü ve akıllı ortamlar olmak üzere ticari alanlarda veya avatar oluşturma ve multimedya uygulamaları olmak üzere kişisel kullanımlarda geniş bir kullanım alanı bulmaktadır. Bu nedenle günümüzde, gürbüz ve hızlı bir şekilde çalışan YİT uygulamalarına büyük bir ihtiyaç duyulmaktadır.

Bu tez çalışmasında, YİT'in başarımını artırmak üzere iki farklı çalışma sunulmuştur. Yapılan ilk çalışmada, yüz ifadesini temsil eden yüz üzerindeki ilgili alanın seçimi için bir kırpma metodolojisi önerilmiştir. Ardından bu metodoloji yardımıyla elde edilen imgeler, farklı mimarilere ait evrışimsel sinir ağları üzerinde eğitilmiştir. Gerçekleştirilen çalışmanın sonucunda, yüz ifadesi içeren alan ve ağ seçiminin, YİT üzerindeki etkisi hakkında önemli bulgular elde edilmiştir.

Tez kapsamında sunulan ikinci çalışmada, kısıtlı sayıda örnek içeren bir yüz ifadesi veri tabanının örnek sayısını artırmak için çekişmeli üretici ağlar temelli bir veri artırımı yaklaşımı sunulmuştur. Uygulanan veri artırımı sonrasında elde edilen imgeler, bir ESA yardımı ile eğitime tabii tutulmuştur. Söz konusu ağın test işlemi sonucuna göre, verinin dağılımını doğru bir şekilde yansıtan çekişmeli üretici ağlar yardımıyla artırılan örneklerin, yüksek başarımla sergilediği tespit edilmiştir.

Tez kapsamında gerçekleştirilen çalışmalar, YİT uygulamaları için, ifade ile ilgili alanın seçilimi, ağ seçilimi ve öznitelik uzayı temelli örnek artırımı için yeni yaklaşımlar sunmaktadır. Uygulamalar sonucunda elde edilen başarımlar, literatürdeki mevcut yöntemlerle kıyaslanmış ve önerilen yöntemlerle elde edilen başarımların daha yüksek olduğu gözlemlenmiştir. Bu sonuçlar, tez kapsamında önermiş olduğumuz ilgili alan seçimi ve öznitelik uzayı temelli örnek artırımının, YİT'in başarımını arttırabileceği hipotezini desteklemektedir.

KAYNAKLAR

- [1] Y. Zhang, C. Hua, Driver fatigue recognition based on facial expression analysis using local binary patterns, *Optik (Stuttg)*. 126 (2015) 4501–4505. <https://doi.org/10.1016/j.ijleo.2015.08.185>.
- [2] A.D. Torres, H. Yan, A.H. Aboutaleb, A. Das, L. Duan, P. Rad, Patient facial emotion recognition and sentiment analysis using secure cloud with hardware acceleration, *Comput. Intell. Multimed. Big Data Cloud with Eng. Appl.* (2018) 61–89. <https://doi.org/10.1016/B978-0-12-813314-9.00003-7>.
- [3] K. Suzuki, F. Nakamura, J. Otsuka, K. Masai, Y. Itoh, Y. Sugiura, M. Sugimoto, Recognition and mapping of facial expressions to avatar by embedded photo reflective sensors in head mounted display, *Proc. - IEEE Virtual Real.* (2017) 177–185. <https://doi.org/10.1109/VR.2017.7892245>.
- [4] M.E. Barrera, D. Maurer, The perception of facial expressions by the three-month-old, *Child Dev.* 52 (1981) 203–206.
- [5] R.F. Caron, A.J. Caron, R.S. Myers, Do infants see emotional expressions in static faces?, *Child Dev.* 56 (1985) 1552–1560. <https://doi.org/10.1111/j.1467-8624.1985.tb00220.x>.
- [6] P. Ekman, Universals and Cultural Differences in Facial Expressions of Emotion BT - Nebraska Symposium on Motivation, in: *Nebraska Symp. Motiv.*, 1972: pp. 207–282. [papers3://publication/uuid/FDC5E29A-0E28-4DDF-B1A4-F53FEE0B4F70](https://doi.org/10.1016/j.jneuro.2016.07.026).
- [7] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, T. Aila, Analyzing and improving the image quality of StyleGAN, *ArXiv*. (2019).
- [8] A.T. Lopes, E. de Aguiar, A.F. De Souza, T. Oliveira-Santos, Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order, *Pattern Recognit.* 61 (2017) 610–628. <https://doi.org/10.1016/j.patcog.2016.07.026>.
- [9] F. Lin, R. Hong, W. Zhou, H. Li, Facial Expression Recognition with Data Augmentation and Compact Feature Learning, *Proc. - Int. Conf. Image Process. ICIP.* (2018) 1957–1961. <https://doi.org/10.1109/ICIP.2018.8451039>.
- [10] D.A. Pitaloka, A. Wulandari, T. Basaruddin, D.Y. Liliana, Enhancing CNN with Preprocessing Stage in Automatic Emotion Recognition, *Procedia Comput. Sci.* 116 (2017) 523–529. <https://doi.org/10.1016/j.procs.2017.10.038>.
- [11] S. Zafeiriou, C. Zhang, Z. Zhang, A survey on face detection in the wild: Past, present and future, *Comput. Vis. Image Underst.* 138 (2015) 1–24. <https://doi.org/10.1016/j.cviu.2015.03.015>.
- [12] W. Zhao, R. Chellappa, P.J. Phillips, A. Rosenfeld, Face Recognition: A Literature Survey, *ACM Comput. Surv.* 35 (2003) 399–458.
- [13] Z. Kalal, K. Mikolajczyk, J. Matas, Face-TLD: Tracking-learning-detection applied to faces, *Proc. - Int. Conf. Image Process. ICIP.* (2010) 3789–3792. <https://doi.org/10.1109/ICIP.2010.5653525>.
- [14] S. Li, W. Deng, Deep Facial Expression Recognition: A Survey, *IEEE Trans. Affect. Comput.* (2020) 1–22. <https://doi.org/10.1109/TAFFC.2020.2981446>.
- [15] J. Kumari, R. Rajesh, K.M. Pooja, Facial Expression Recognition: A Survey, *Procedia Comput. Sci.* 58 (2015) 486–491. <https://doi.org/10.1016/j.procs.2015.08.011>.
- [16] Y. Fu, G. Guo, T.S. Huang, Age synthesis and estimation via faces: A survey, *IEEE Trans. Pattern Anal. Mach. Intell.* 32 (2010) 1955–1976. <https://doi.org/10.1109/TPAMI.2010.36>.
- [17] A. Laurentini, A. Bottino, Computer analysis of face beauty: A survey, *Comput. Vis. Image Underst.* 125 (2014) 184–199. <https://doi.org/10.1016/j.cviu.2014.04.006>.
- [18] V. Blanz, T. Vetter, A morphable model for the synthesis of 3D faces, *Proc. 26th Annu. Conf. Comput. Graph. Interact. Tech. - SIGGRAPH '99.* (1999) 187–194. <https://doi.org/10.1145/311535.311556>.
- [19] E. Borovikov, S. Vajda, Facematch: Real-world face image retrieval, *Commun. Comput. Inf. Sci.* 709 (2017) 405–419. https://doi.org/10.1007/978-981-10-4859-3_36.
- [20] I. Kemelmacher-Shlizerman, E. Shechtman, R. Garg, S.M. Seitz, Exploring photobios, *ACM Trans. Graph.* 30 (2011) 1. <https://doi.org/10.1145/2010324.1964956>.

- [21] W.W. Bledsoe, H. Chan, A man-machine facial recognition system—some preliminary results, Panor. Res. Inc, Palo Alto, California., Tech. Rep. PRI A. 19 (1965) 1965.
- [22] W.W. Bledsoe, Man-Machine Facial Recognition, Groningen Image Process. Syst. GIPSY. (1966).
- [23] S. Toshiyuki, M. Nagao, T. Kanade, Computer analysis and classification of photographs of human faces, First USA—Japan Comput. Conf. (1972) 55–62.
- [24] M.A. Fischler, R.A. Elschlager, The Representation and Matching of Pictorial Structures Representation, IEEE Trans. Comput. C–22 (1973) 67–92. <https://doi.org/10.1109/T-C.1973.223602>.
- [25] M.H. Yang, D.J. Kriegman, N. Ahuja, Detecting faces in images: A survey, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 34–58. <https://doi.org/10.1109/34.982883>.
- [26] R. Chellappa, S. Sirohey, C.L. Wilson, Human and Machine Recognition of Faces: A Survey, Proc. IEEE. 83 (1995) 705–741. <https://doi.org/10.1109/5.381842>.
- [27] E. Hjelmås, B.K. Low, Face detection: A survey, Comput. Vis. Image Underst. 83 (2001) 236–274. <https://doi.org/10.1006/cviu.2001.0921>.
- [28] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, Proc. 2001 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition. CVPR 2001. 1 (n.d.) I-511-I-518. <https://doi.org/10.1109/CVPR.2001.990517>.
- [29] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2004) 91–110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>.
- [30] N. Dalal, B. Triggs, Histogram of Oriented Gradients for Human Detection, 2 (2005) 886–893.
- [31] B. Jun, I. Choi, D. Kim, Local transform features and hybridization for accurate face and human detection, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 1423–1436. <https://doi.org/10.1109/TPAMI.2012.219>.
- [32] T. Ahonen, A. Hadid, M. Pietikäinen, Face Recognition with Local Binary Patterns, (2004) 469–481. https://doi.org/10.1007/978-3-540-24670-1_36.
- [33] T. Ojala, M. Pietikainen, T. Maenpää, Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, {sc Ieee} Trans. Pattern Anal. Mach. Intell. 24 (2002) 971–987.
- [34] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-Up Robust Features (SURF), Comput. Vis. Image Underst. 110 (2008) 346–359. <https://doi.org/10.1016/j.cviu.2007.09.014>.
- [35] E. Tola, V. Lepetit, P. Fua, DAISY: An efficient dense descriptor applied to wide-baseline stereo, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 815–830. <https://doi.org/10.1109/TPAMI.2009.77>.
- [36] P. Dollar, Z. Tu, P. Perona, S. Belongie, Integral Channel Features, in: Proceedings Br. Mach. Vis. Conf. 2009, British Machine Vision Association, 2009: pp. 91.1-91.11. <https://doi.org/10.5244/C.23.91>.
- [37] M. Mathias, R. Benenson, M. Pedersoli, L. Van Gool, Face detection without bells and whistles, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 8692 LNCS (2014) 720–735. https://doi.org/10.1007/978-3-319-10593-2_47.
- [38] M. Everingham, S.M.A. Eslami, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The Pascal Visual Object Classes Challenge: A Retrospective, Int. J. Comput. Vis. 111 (2014) 98–136. <https://doi.org/10.1007/s11263-014-0733-5>.
- [39] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2010) 303–338. <https://doi.org/10.1007/s11263-009-0275-4>.
- [40] G.B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, Univ. Massachusetts Amherst Tech. Rep. 1 (2007) 07–49. <https://doi.org/10.1.1.122.8268>.
- [41] V. Jain, E. Learned-Miller, FDDB : A Benchmark for Face Detection in Unconstrained Settings, -. (2010) UM-CS-2010-009.

- [42] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, (1995) 23–37. https://doi.org/10.1007/3-540-59119-2_166.
- [43] C. Cortes, V. Vapnik, Support-Vector Networks, *Mach. Learn.* 20 (1995) 273–297. <https://doi.org/10.1023/A:1022627411411>.
- [44] I. Tsouchantaris, T. Hofmann, T. Joachims, Y. Altun, Support vector machine learning for interdependent and structured output spaces, *Twenty-First Int. Conf. Mach. Learn. - ICML '04.* (2004) 104. <https://doi.org/10.1145/1015330.1015341>.
- [45] A. Krizhevsky, G.E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, in: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (Eds.), *Neural Inf. Process. Syst.*, Curran Associates, Inc., 2012: pp. 1–9. <https://doi.org/http://dx.doi.org/10.1016/j.protcy.2014.09.007>.
- [46] P.F. Felzenszwalb, R.B. Girshick, D. Mcallester, D. Ramanan, Object Detection with Discriminatively Trained Part Based Models, [1] P. F. Felzenszwalb, R. B. Girshick, D. Mcallester, D. Ramanan, “Object Detect. with Discrim. Trained Part Based Model. *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 32, No. 9, Pp. 1–20, 2009. *IEEE Trans. Pattern Anal. Ma.* 32 (2009) 1–20. <https://doi.org/10.1109/TPAMI.2009.167>.
- [47] G. Bradski, A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly Media Inc. 1 (2008). <https://doi.org/10.1109/MRA.2009.933612>.
- [48] R.B. Girshick, P.F. Felzenszwalb, D. McAllester, Discriminatively trained deformable part models, release 5, (n.d.). <http://people.cs.uchicago.edu/rbg/latent-release5/>.
- [49] Y. Jia, Caffe: An Open Source Convolutional Architecture for Fast Feature Embedding, <Http://Caffe.Berkeleyvision.Org/>. (2013). <http://caffe.berkeleyvision.org/>.
- [50] P. Viola, M. Jones, Robust real-time face detection, *Int. J. Comput. Vis.* 57 (2004) 137–154. <https://doi.org/10.1023/B:VISI.0000013087.49260.fb>.
- [51] S. Ren, K. He, R. Girshick, J. Sun, J. Donahue, T. Darrell, J. Malik, W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, R. Girshick, D. Impiombato, S. Giarrusso, T. Mineo, O. Catalano, C. Gargano, G. La Rosa, F. Russo, G. Sottile, S. Billotta, G. Bonanno, S. Garozzo, A. Grillo, D. Marano, G. Romeo, D. Erhan, C. Szegedy, A. Toshev, D. Anguelov, Rich feature hierarchies for accurate object detection and semantic segmentation, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* 794 (2015) 1–15. <https://doi.org/10.1016/j.nima.2015.05.028>.
- [52] Z. Cha, Z. Zhengyou, Improving multiview face detection with multi-task deep convolutional neural networks, *Appl. Comput. Vis. (WACV)*, 2014 *IEEE Winter Conf.* (2014) 1036–1041. <https://doi.org/10.1109/WACV.2014.6835990>.
- [53] H. Li, Z. Lin, J. Brandt, X. Shen, G. Hua, Efficient boosted exemplar-based face detection, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* (2014) 1843–1850. <https://doi.org/10.1109/CVPR.2014.238>.
- [54] X. Shen, Z. Lin, J. Brandt, Y. Wu, Detecting and aligning faces by image retrieval, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* (2013) 3460–3467. <https://doi.org/10.1109/CVPR.2013.444>.
- [55] P.F. Felzenszwalb, D.P. Huttenlocher, Pictorial structures for object recognition, *Int. J. Comput. Vis.* 61 (2005) 55–79. <https://doi.org/10.1023/B:VISI.0000042934.15159.49>.
- [56] X. Zhu, D. Ramanan, Face detection, pose estimation, and landmark localization in the wild, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* (2012) 2879–2886. <https://doi.org/10.1109/CVPR.2012.6248014>.
- [57] D. Chen, S. Ren, Y. Wei, X. Cao, J. Sun, Joint cascade face detection and alignment, *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*. 8694 LNCS (2014) 109–122. https://doi.org/10.1007/978-3-319-10599-4_8.
- [58] K. Mikolajczyk, C. Schmid, A. Zisserman, Human detection based on a probabilistic assembly of robust part detectors, *Proc. Eur. Conf. Comput. Vis.* 3021 (2004) 69–82. https://doi.org/10.1007/978-3-540-24670-1_6.
- [59] H. Schneiderman, T. Kanade, Object detection using the statistics of parts, *Int J Comput Vis.* 56 (2004)

- [60] F.C. Crow, Summed-Area Tables for Texture Mapping., *Comput. Graph.* 18 (1984) 207–212. <https://doi.org/10.1145/964965.808600>.
- [61] S.Z. Li, L. Zhu, Z. Zhang, A. Blake, H. Zhang, H. Shum, Statistical learning of multi-view face detection, *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*. 2353 (2002) 67–81. https://doi.org/10.1007/3-540-47979-1_5.
- [62] T. Mita, T. Kaneko, O. Hori, Joint Haar-like features for face detection, *Proc. IEEE Int. Conf. Comput. Vis. II* (2005) 1619–1626. <https://doi.org/10.1109/ICCV.2005.129>.
- [63] R. Xiao, L. Zhu, H.J. Zhang, Boosting chain learning for object detection, *Proc. IEEE Int. Conf. Comput. Vis.* 1 (2003) 709–715. <https://doi.org/10.1109/iccv.2003.1238417>.
- [64] C. Darwin, *The expression of the emotions in man and animals.*, John Murray, London, 1872. <https://doi.org/10.1037/10001-000>.
- [65] P. Ekman, W. V. Friesen, Constants across cultures in the face and emotion, *J. Pers. Soc. Psychol.* 17 (1971) 124–129. <https://doi.org/10.1037/h0030377>.
- [66] R.E. Jack, O.G.B. Garrod, H. Yu, R. Caldara, P.G. Schyns, Facial expressions of emotion are not culturally universal, *Proc. Natl. Acad. Sci. U. S. A.* 109 (2012) 7241–7244. <https://doi.org/10.1073/pnas.1200155109>.
- [67] A. Mehrabian, J.A. Russell, *An Approach to Environmental Psychology*, (1974) 266.
- [68] P. Lucey, J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, I. Matthews, The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression, 2010 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. - Work. CVPRW 2010. (2010) 94–101. <https://doi.org/10.1109/CVPRW.2010.5543262>.
- [69] M. Kamachi, M. Lyons, J. Gyoba, The japanese female facial expression (jaffe) database, URL <Http://Www.Kasrl.Org/Jaffe.Html>. 21 (1998) 32.
- [70] M.F. Valstar, M. Pantic, Induced Disgust, Happiness and Surprise: an Addition to the MMI Facial Expression Database, *Proc. Int'l Conf. Lang. Resour. Eval. Work. Emot.* (2010) 65–70. http://www.cs.nott.ac.uk/~mfv/Documents/MMI_spontaneous.pdf.
- [71] A. Mollahosseini, B. Hasani, M.H. Mahoor, AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild, *IEEE Trans. Affect. Comput.* 10 (2019) 18–31. <https://doi.org/10.1109/TAFFC.2017.2740923>.
- [72] D. Kollias, P. Tzirakis, M.A. Nicolaou, A. Papaioannou, G. Zhao, B. Schuller, I. Kotsia, S. Zafeiriou, Deep Affect Prediction in-the-Wild: Aff-Wild Database and Challenge, *Deep Architectures, and Beyond, Int. J. Comput. Vis.* 127 (2019) 907–929. <https://doi.org/10.1007/s11263-019-01158-4>.
- [73] I.J. Goodfellow, D. Erhan, P. Luc Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.H. Lee, Y. Zhou, C. Ramaiah, F. Feng, R. Li, X. Wang, D. Athanasakis, J. Shawe-Taylor, M. Milakov, J. Park, R. Ionescu, M. Popescu, C. Grozea, J. Bergstra, J. Xie, L. Romaszko, B. Xu, Z. Chuang, Y. Bengio, Challenges in representation learning: A report on three machine learning contests, *Neural Networks*. 64 (2015) 59–63. <https://doi.org/10.1016/j.neunet.2014.09.005>.
- [74] C.A. Corneanu, M.O. Simón, J.F. Cohn, S.E. Guerrero, Survey on RGB, 3D, Thermal, and Multimodal Approaches for Facial Expression Recognition: History, Trends, and Affect-Related Applications, *IEEE Trans. Pattern Anal. Mach. Intell.* 38 (2016) 1548–1568. <https://doi.org/10.1109/TPAMI.2016.2515606>.
- [75] C. Shan, S. Gong, P.W. McOwan, Facial expression recognition based on Local Binary Patterns: A comprehensive study, *Image Vis. Comput.* 27 (2009) 803–816. <https://doi.org/10.1016/j.imavis.2008.08.005>.
- [76] G. Zhao, M. Pietikäinen, Dynamic texture recognition using local binary patterns with an application to facial expressions, *IEEE Trans. Pattern Anal. Mach. Intell.* 29 (2007) 915–928. <https://doi.org/10.1109/TPAMI.2007.1110>.
- [77] R. Zhi, M. Flierl, Q. Ruan, W.B. Kleijn, Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition, *IEEE Trans. Syst. Man, Cybern. Part B Cybern.* 41

- (2011) 38–52. <https://doi.org/10.1109/TSMCB.2010.2044788>.
- [78] P. Ekman, W. V. Friesen, J.C. Hager, Facial Action Coding System: The Manual on CD ROM, (2002).
 - [79] L.I. Reed, M.A. Sayette, J.F. Cohn, Impact of depression on response to comedy: A dynamic facial coding analysis, *J. Abnorm. Psychol.* 116 (2007) 804–809. <https://doi.org/10.1037/0021-843X.116.4.804>.
 - [80] A.C. Lints-Martindale, T. Hadjistavropoulos, B. Barber, S.J. Gibson, A psychophysical investigation of the facial action coding system as an index of pain variability among older adults with and without Alzheimer’s disease, *Pain Med.* 8 (2007) 678–689. <https://doi.org/10.1111/j.1526-4637.2007.00358.x>.
 - [81] W.Y. Chang, S.H. Hsu, J.H. Chien, FATAUVA-Net: An Integrated Deep Learning Framework for Facial Attribute Recognition, Action Unit Detection, and Valence-Arousal Estimation, *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.* 2017-July (2017) 1963–1971. <https://doi.org/10.1109/CVPRW.2017.246>.
 - [82] P. Ekman, W. V. Friesen, J.C. Hager, Facial Action Coding System - Investigator’s Guide, *Facs.* (2002) 1–197. papers://66d17d8d-d8b0-4a45-a383-99034da5c2f1/Paper/p245.
 - [83] S. Du, Y. Tao, A.M. Martinez, Compound facial expressions of emotion, *Proc. Natl. Acad. Sci. U. S. A.* 111 (2014). <https://doi.org/10.1073/pnas.1322355111>.
 - [84] P. Ekman, Darwin, Deception, and Facial Expression, *Ann. N. Y. Acad. Sci.* 1000 (2003) 205–221. <https://doi.org/10.1196/annals.1280.010>.
 - [85] Y. Wu, Q. Ji, Facial Landmark Detection: A Literature Survey, *Int. J. Comput. Vis.* 127 (2019) 115–142. <https://doi.org/10.1007/s11263-018-1097-z>.
 - [86] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, M. Pantic, 300 Faces In-The-Wild Challenge: database and results, *Image Vis. Comput.* 47 (2016) 3–18. <https://doi.org/10.1016/j.imavis.2016.01.002>.
 - [87] T.F. Cootes, G.J. Edwards, C.J. Taylor, Active appearance models, *IEEE Trans. Pattern Anal. Mach. Intell.* 23 (2001). <https://doi.org/https://doi.org/10.1109/34.927467>.
 - [88] J.M. Saragih, S. Lucey, J.F. Cohn, Deformable model fitting by regularized landmark mean-shift, *Int. J. Comput. Vis.* 91 (2011) 200–215. <https://doi.org/10.1007/s11263-010-0380-4>.
 - [89] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, The MIT Press, 2016.
 - [90] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M.P. Reyes, M.L. Shyu, S.C. Chen, S.S. Iyengar, A survey on deep learning: Algorithms, techniques, and applications, *ACM Comput. Surv.* 51 (2018) 1–36. <https://doi.org/10.1145/3234150>.
 - [91] M.M. Najafabadi, F. Villanustre, T.M. Khoshgoftaar, N. Seliya, R. Wald, E. Muharemagic, Deep learning applications and challenges in big data analytics, *J. Big Data.* 2 (2015). <https://doi.org/10.1186/s40537-014-0007-7>.
 - [92] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE.* 86 (1998) 2278–2323. <https://doi.org/10.1109/5.726791>.
 - [93] Z. Li, W. Yang, S. Peng, F. Liu, A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects, *ArXiv.* (2020). <http://arxiv.org/abs/2004.02806>.
 - [94] Y. Lecun, Y. Bengio, G. Hinton, Deep learning, *Nature.* 521 (2015) 436–444. <https://doi.org/10.1038/nature14539>.
 - [95] C.E. Nwankpa, W. Ijomah, A. Gachagan, S. Marshall, Activation functions: Comparison of trends in practice and research for deep learning, *ArXiv.* (2018).
 - [96] V. Nair, G. E. Hinton, Rectified Linear Units Improve Restricted Boltzmann Machines Vinod Nair, *Proc. ICML.* 27 (2010) 807–814.
 - [97] P. Ramachandran, B. Zoph, Q. V. Le, Searching for activation functions, *ArXiv.* (2017).
 - [98] M.D. Zeiler, M. Ranzato, R. Monga, M. Mao, K. Yang, Q. V. Le, P. Nguyen, A. Senior, V. Vanhoucke, J. Dean, G.E. Hinton, On rectified linear units for speech processing, *ICASSP, IEEE Int.*

- Conf. Acoust. Speech Signal Process. - Proc. (2013) 3517–3521. <https://doi.org/10.1109/ICASSP.2013.6638312>.
- [99] X.Z. Kaiming He Shaoqing Ren, and Jian Sun, Surpassing Human-Level Performance on ImageNet Classification, ArXiv Prepr. ArXiv. (2015) 1502.01852.
 - [100] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc. (2015).
 - [101] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Adv. Neural Inf. Process. Syst., MIT Press, Cambridge, MA, USA, 2014: pp. 2672–2680. https://doi.org/10.3156/jsoft.29.5_177_2.
 - [102] Z. Wang, Q. She, T.E. Ward, Generative adversarial networks: A survey and taxonomy, ArXiv. (2019).
 - [103] I.G. Openai, NIPS 2016 Tutorial: Generative Adversarial Networks, ArXiv. (2017). <http://www.iangoodfellow.com/slides/2016-12-04-NIPS.pdf>.
 - [104] C. Doersch, Tutorial on Variational Autoencoders, (2016). <http://arxiv.org/abs/1606.05908>.
 - [105] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, 4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc. (2016).
 - [106] J. Susskind, A. Anderson, G.E. Hinton, The Toronto face dataset, U. Toronto, Tech. Rep. UTML TR. 1 (2010).
 - [107] T. Karras, T. Aila, S. Laine, J. Lehtinen, Progressive growing of gans for improved quality, stability, and variation, ArXiv. (2017).
 - [108] T. Karras, S. Laine, T. Aila, A style-based generator architecture for generative adversarial networks, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2019-June (2019) 4396–4405. <https://doi.org/10.1109/CVPR.2019.00453>.
 - [109] A. Karnewar, O. Wang, MSG-GAN: Multi-Scale Gradients for Generative Adversarial Networks, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (2020) 7796–7805. <https://doi.org/10.1109/CVPR42600.2020.00782>.
 - [110] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, GANs trained by a two time-scale update rule converge to a local Nash equilibrium, Adv. Neural Inf. Process. Syst. 2017-Decem (2017) 6627–6638.
 - [111] R. Abdal, Y. Qin, P. Wonka, Image2StyleGAN++: How to edit the embedded images?, ArXiv. (2019).
 - [112] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, D. Cohen-Or, Encoding in Style: a StyleGAN Encoder for Image-to-Image Translation, ArXiv. (2020).
 - [113] J. Zhu, Y. Shen, D. Zhao, B. Zhou, In-Domain GAN Inversion for Real Image Editing, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 12362 LNCS (2020) 592–608. https://doi.org/10.1007/978-3-030-58520-4_35.
 - [114] K. Li, Y. Jin, M.W. Akram, R. Han, J. Chen, Facial expression recognition with convolutional neural networks via a new face cropping and rotation strategy, Vis. Comput. 36 (2020) 391–404. <https://doi.org/10.1007/s00371-019-01627-4>.
 - [115] Y.F. Wen, H.M. Wong, R. Lin, G. Yin, C. McGrath, Inter-ethnic/racial facial variations: A systematic review and Bayesian meta-analysis of photogrammetric studies, PLoS One. 10 (2015). <https://doi.org/10.1371/journal.pone.0134525>.
 - [116] J.M. Girard, J.F. Cohn, L.A. Jeni, S. Lucey, F. De La Torre, How much training data for facial action unit detection?, 2015 11th IEEE Int. Conf. Work. Autom. Face Gesture Recognition, FG 2015. (2015). <https://doi.org/10.1109/FG.2015.7163106>.
 - [117] X. Fan, T. Tjahjadi, A spatial-temporal framework based on histogram of gradients and optical flow for facial expression recognition in video sequences, Pattern Recognit. 48 (2015) 3407–3416. <https://doi.org/10.1016/j.patcog.2015.04.025>.
 - [118] T.H.H. Zavaschi, A.S. Britto, L.E.S. Oliveira, A.L. Koerich, Fusion of feature sets and classifiers for

- facial expression recognition, *Expert Syst. Appl.* 40 (2013) 646–655. <https://doi.org/10.1016/j.eswa.2012.07.074>.
- [119] A.R. Rivera, J.R. Castillo, O. Chae, Local directional number pattern for face analysis: face and expression recognition., *IEEE Trans. Image Process.* 22 (2013) 1740–1752. <https://doi.org/10.1109/TIP.2012.2235848>.
- [120] W. Gu, C. Xiang, Y. V. Venkatesh, D. Huang, H. Lin, Facial expression recognition using radial encoding of local Gabor features and classifier synthesis, *Pattern Recognit.* 45 (2012) 80–91. <https://doi.org/10.1016/j.patcog.2011.05.006>.
- [121] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, D.N. Metaxas, Learning active facial patches for expression analysis, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* (2012) 2562–2569. <https://doi.org/10.1109/CVPR.2012.6247974>.
- [122] M. Liu, S. Li, S. Shan, X. Chen, AU-inspired Deep Networks for Facial Expression Feature Learning, *Neurocomputing.* 159 (2015) 126–136. <https://doi.org/10.1016/j.neucom.2015.02.011>.
- [123] M. Liu, S. Shan, R. Wang, X. Chen, Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* (2014) 1749–1756. <https://doi.org/10.1109/CVPR.2014.226>.
- [124] S.H. Lee, W.J. Baddar, Y.M. Ro, Collaborative expression representation using peak expression and intra class variation face images for practical subject-independent emotion recognition in videos, *Pattern Recognit.* 54 (2016) 52–67. <https://doi.org/10.1016/j.patcog.2015.12.016>.
- [125] F. De La Torre, W.S. Chu, X. Xiong, F. Vicente, X. Ding, J. Cohn, IntraFace, 2015 11th IEEE Int. Conf. Work. Autom. Face Gesture Recognition, FG 2015. (2015). <https://doi.org/10.1109/FG.2015.7163082>.
- [126] P. Liu, S. Han, Z. Meng, Y. Tong, Facial expression recognition via a boosted deep belief network, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* (2014) 1805–1812. <https://doi.org/10.1109/CVPR.2014.233>.
- [127] N. Jain, S. Kumar, A. Kumar, P. Shamsolmoali, M. Zareapoor, Hybrid deep neural networks for face emotion recognition, *Pattern Recognit. Lett.* 115 (2018) 101–106. <https://doi.org/10.1016/j.patrec.2018.04.010>.
- [128] D.K. Jain, P. Shamsolmoali, P. Sehdev, Extended deep neural network for facial emotion recognition, *Pattern Recognit. Lett.* 120 (2019) 69–74. <https://doi.org/10.1016/j.patrec.2019.01.008>.
- [129] H. Yang, Z. Zhang, L. Yin, Identity-adaptive facial expression recognition through expression regeneration using conditional generative adversarial networks, *Proc. - 13th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2018.* (2018) 294–301. <https://doi.org/10.1109/FG.2018.00050>.
- [130] Y. Li, J. Zeng, S. Shan, X. Chen, Occlusion Aware Facial Expression Recognition Using CNN With Attention Mechanism, *IEEE Trans. Image Process.* 28 (2019) 2439–2450. <https://doi.org/10.1109/TIP.2018.2886767>.

ÖZGEÇMİŞ

Ömer Faruk SÖYLEMEZ

[illegible]

[REDACTED]
[REDACTED] [REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]

ARAŞTIRMA DENEYİMİ

- ✓ Programlama Dilleri : Python, Matlab, C#
- ✓ Kullanılan Frameworkler: Tensorflow, Pytorch, OpenCV

AKADEMİK FAALİYETLER

Makaleler:

1. O.F. Soylemez, B. Ergen, Facial Landmark Based Region of Interest Localization for Deep Facial Expression Recognition, Teh. Vjesn. 29 (2022). <https://doi.org/10.17559/TV-20200423145443>.
2. Ö.F. Söylemez, B. Ergen, Farklı Evrişimsel Sinir Ağı Mimarilerinin Yüz İfade Analizi Alanındaki Başarımlarının İncelenmesi, DÜMF Mühendislik Derg. 11 (2020) 123–133. <https://doi.org/10.24012/dumf.679793>.

Bildiriler:

1. V. Tümen, Ö.F. Söylemez, B. Ergen, Facial emotion recognition on a dataset using Convolutional Neural Network, IDAP 2017 - Int. Artif. Intell. Data Process. Symp. (2017). <https://doi.org/10.1109/IDAP.2017.8090281>.