

T.C.
GÜMÜŞHANE ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

BİYOTEKNOLOJİ ANA BİLİM DALI

TERMAL GÖRÜNTÜLERDE ATEŞ TESPİTİNDE KLASİK/MODERN
METOTLAR VE GENİŞ DİL MODELİ YAKLAŞIMI

YÜKSEK LİSANS

Adem ÖZTÜRK

ŞUBAT-2025
GÜMÜŞHANE



**T.C.
GÜMÜŞHANE ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

BİYOTEKNOLOJİ ANA BİLİM DALI

**TERMAL GÖRÜNTÜLERDE ATEŞ TESPİTİNDE KLASİK/MODERN
METOTLAR VE GENİŞ DİL MODELİ YAKLAŞIMI**

**CLASSICAL/MODERN METHODS AND LARGE LANGUAGE MODEL
APPROACH IN FEVER DETECTION FROM THERMAL IMAGES**

YÜKSEK LİSANS

Adem ÖZTÜRK

**ŞUBAT-2025
GÜMÜŞHANE**



T.C.

**GÜMÜŞHANE ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

BİYOTEKNOLOJİ ANA BİLİM DALI

**TERMAL GÖRÜNTÜLERDE ATEŞ TESPİTİNDE KLASİK/MODERN
METOTLAR VE GENİŞ DİL MODELİ YAKLAŞIMI**

**CLASSICAL/MODERN METHODS AND LARGE LANGUAGE MODEL
APPROACH IN FEVER DETECTION FROM THERMAL IMAGES**

YÜKSEK LİSANS

Adem ÖZTÜRK

Danışman: Dr. Öğr. Üyesi Emrah KAPLAN

**ŞUBAT-2025
GÜMÜŞHANE**

BİLİMSEL ETİĞE UYGUNLUK BEYANI

Yüksek Lisans Tezi olarak hazırlamış olduğum "**Termal Görüntülerde Ateş Tespitinde Klasik/Modern Metotlar ve Geniş Dil Modeli Yaklaşımı**" isimli bu tezimin, tamamen kendi çalışmam olduğunu, her alıntıya kaynak gösterdiğimi, alıntı yaptığım tüm çalışmaları kaynakçada belirttiğimi ve Gümüşhane Üniversitesi'nin lisanslı kullanıcısı olduğu intihal yazılım programı ile Lisansüstü Eğitim Enstitüsünün belirlediği kıstaslara uygun olarak raporladığımı taahhüt ederim. Tezimin kâğıt ve elektronik kopyalarının Gümüşhane Üniversitesi Lisansüstü Eğitim Enstitüsü arşivinde saklanmasına izin verdiğimi onaylarım.

Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili maddeleri uyarınca gereğinin yapılmasını arz ederim.

05.02.2025

Adem ÖZTÜRK

TEŞEKKÜR

Bu tez çalışması, ateşli hastalıkların tespiti amacıyla, termal görüntüler üzerinde farklı yapay zeka ve makine öğrenimi yöntemlerinin karşılaştırılması üzerine yapılmıştır. Bu süreç boyunca, bana yol gösteren ve desteğini esirgemeyen danışman hocam Dr. Öğr. Üyesi Emrah KAPLAN'a en içten teşekkürlerimi sunarım. Ayrıca, bu çalışmada bana destek veren Dr. Öğr. Üyesi Muhammed Taha KÖROĞLU hocama ve değerli aileme, göstermiş oldukları sabır ve destekleri için minnettarım. Bu tezde emeği geçen ve her aşamada yanımda olan herkese teşekkür ederim.

Adem ÖZTÜRK

Gümüşhane-2025

ÖZET

Bu tez çalışmasında, termal görüntü ile COVID-19 ve benzeri ateşli hastalıkların tespiti amacıyla farklı yöntemler karşılaştırılmıştır. Haar Cascade (HC), Derin Öğrenme (Deep Learning) ve Büyük Dil Modeli (LLM) ile gerçekleştirilen analizler, yöntemlerin termal görüntüleme, yüz tanıma ve veri analizi alanındaki performanslarını değerlendirilmiştir. Burada, LLM doğrudan bir görüntü işleme yöntemi olmamış, arka planda belirlediği algoritma ve parametrelerle analiz sürecine katkı sağlamıştır. HC yöntemi, tekil yüzlerin bulunduğu termal görüntülerde %75 doğruluk oranı ile başarılı bir performans sergilerken, çoklu yüz içeren görüntülerde bu oran %50'nin altına düşmüştür. Bu yöntem, RGB görüntülerde daha yüksek doğruluk oranları sergilemesine rağmen, karmaşık termal görüntülerde sınırlamalar göstermiştir. HC yönteminin özellikle düşük kontrast ve düşük çözünürlüklü görüntülerde kullanım potansiyeli dikkat çekmiştir. Derin öğrenme modelleri, termal görüntü işleme alanında yüksek doğruluk oranları ile öne çıkmıştır. Özellikle tekil yüzlerin bulunduğu termal görüntülerde %100 doğruluk oranına yaklaşılmıştır. Bununla birlikte, çoklu yüz içeren görüntülerde başarı oranı %0'a düşmüştür. Model, normal ve yan açıdan alınan görüntülerde etkili sonuçlar sunarken, karmaşık senaryolar için daha fazla iyileştirme gerekliliği ortaya koymuştur. Veri setinin genişletilmesi ve transfer öğrenme tekniklerinin uygulanması, bu modelin performansını artırabilecektir. LLM, belirlediği algoritma ve parametrelerle özellikle yüz tespitinden sonra sıcaklık analizinin anlamlandırılmasında önemli bir rol oynamıştır. Üretilen model, normal yüz tespitinde %62.5 doğruluk oranı ile orta düzeyde bir performans sergilemiştir. Ancak, farklı açılardan çekilmiş yüzlerde ve çoklu yüz içeren görüntülerde başarısı önemli ölçüde düşmüştür. LLM'nin doğal dil işleme yetenekleri, termal görüntülerden elde edilen verilerin daha iyi analiz edilmesine olanak sağlasa da yüz tespiti konusunda Haar Cascade veya Derin Öğrenme yöntemleriyle desteklenmesi gerekmektedir.

Anahtar Kelimeler: Termal görüntüleme, Derin öğrenme, Haar Cascade, Büyük Dil Modeli (LLM).

SUMMARY

In this thesis, various methods were compared for detecting COVID-19 and similar febrile diseases using thermal imaging. The analyses conducted with Haar Cascade (HC), Deep Learning (DL), and Large Language Model (LLM) evaluated the performance of these methods in the fields of thermal imaging, face recognition, and data analysis. The results provided significant findings, particularly in single face detection and the use of thermal images in healthcare applications. The HC method showed successful performance with a 75% accuracy rate in thermal images containing single faces, while this rate dropped below 50% in images with multiple faces. This method demonstrated higher accuracy rates in RGB images, but exhibited limitations in complex thermal images. The potential of the HC method, especially in low contrast and low-resolution images, was noteworthy. Deep learning models stood out in thermal image processing with high accuracy rates. Specifically, a 100% accuracy rate was achieved in thermal images containing single faces. However, the success rate dropped to 0% in images with multiple faces. The model provided effective results in both normal and side-angle images, but revealed the need for further improvements in more complex scenarios. Expanding the dataset and applying transfer learning techniques could enhance the performance of this model. LLM proved to be a powerful tool in interpreting temperature analysis after face detection. The model demonstrated moderate performance with a 62.5% accuracy rate in normal face detection. However, its performance significantly decreased in images with faces captured from different angles or containing multiple faces. Although the natural language processing capabilities of LLM allowed for better analysis of the data obtained from thermal images, face detection still needs to be supported by Haar Cascade or Deep Learning methods.

Keywords: Thermal imaging, Deep learning, Haar Cascade, Large Language Model (LLM).

İÇİNDEKİLER

KABUL VE ONAY	III
BİLİMSEL ETİĞE UYGUNLUK BEYANI.....	IV
TEŞEKKÜR.....	V
ÖZET.....	VI
İÇİNDEKİLER	VIII
TABLolar DİZİNİ	XII
ŞEKİLLER DİZİNİ.....	XIII
EKLER DİZİNİ.....	XVI
SİMGELER VE KISALTMALAR DİZİNİ	XVII
1. GİRİŞ	1
1.1. Amaç	3
1.2. Tezin Organizasyonu	5
2. TEMEL KAVRAMLAR.....	6
2.1. Termal Görüntüleme	6
2.1.1. Termal Kamera Kullanımını Sınırlayan Faktörler	7
2.2. Bilgisayarlı Görü.....	8
2.2. COVID-19.....	9
2.3. Yapay Zeka	11
2.4. Yapay Sinir Ağları	12
2.5. Derin Öğrenme.....	15
2.5.1. Derin Sinir Ağlar.....	17
2.5.2. Konvolüsyonel (Evrişimsel) Sinir Ağlar.....	18
2.7. Literatür Araştırması	21
2.7.1. Yüz Tanıma Konusunda Yapılan Araştırmalar.....	21
2.7.3. Isı Tespiti Konusunda Yapılan Çalışmalar	25
3. MATERYAL VE YÖNTEM	28
3.1. Haar Cascade Algoritması	29
3.4. Derin Öğrenme Uygulamasında Kullanılan CNN ve Programlama Dili.....	31
3.4.1. Python	31
3.4.1.1. Python Yükleme.....	31
3.4.1.1.2. PowerShell	34
3.4.2. OpenCV	35
3.4.2.1. Sanal Ortam Oluşturulması	35
3.4.2.2. OpenCV ve Bağımlı Paketlerin Sanal Ortama Kurulumu	35

3.4.3. Visual Studio Code	36
3.4.4. Termal resimlerin RGB Formata Dönüştürülmesi Ve Sıcaklık Haritasının Oluşturulması	36
3.5. Kullanılan Veri Setleri	39
3.5.1. Termal Yüz Görüntüleri.....	39
3.5.2. Veri arttırma yöntemleri.....	40
3.6. Tasarlanan Modeller.....	41
3.6.1. Haar Cascade Yöntemi ile Yüz Tespiti.....	41
3.6.1.1. Python Kodunda Haar Cascade Uygulaması	41
3.6.1.1.1. Modelin Yüklenmesi.....	41
3.6.1.1.2. Görüntünün Hazırlanması	41
3.6.1.1.3. Parametre Ayarları ve Yüz Tespiti.....	41
3.6.1.1.4. Tespit Edilen Yüzlerde Sıcaklık Analizi.....	42
3.6.1.1.5. Görselleştirme	42
3.6.2 Derin Öğrenme ve Konvolüsyonel Sinir Ağları Tabanlı Derin Öğrenme Yöntemi ile Yüz Tespiti	42
3.6.2.1. Derin Öğrenme.....	42
3.6.2.2. Konvolüsyonel Sinir Ağları	44
3.6.2.3. CNN'nin katmanları.....	47
3.6.2.3.1. Giriş katmanı.....	47
3.6.2.3.2. Konvolüsyon katmanı	47
3.6.2.3.3. Düzeltmeli doğrusal birim (ReLU)	49
3.6.2.3.4. Havuzlama Katmanı.....	50
3.6.2.3.5. Tam bağlantılı katman.....	52
3.6.2.3.6. Düğüm seyreltme katmanı	52
3.6.2.3.7. Sınıflandırma katmanı.....	53
3.6.2.2. Nesne sınıflandırmada kullanılan CNN modelleri.....	53
3.6.2.2.1. İstifleme (Stacked) blok içeren modeller	54
3.6.2.2.2. Kalıntı (Residual) blok içeren modeller	55
3.6.2.2.3. Başlangıç (Inception) blok içeren modeller	55
3.6.2.2.4. Yoğun (Dense) blok içeren modeller	56
3.6.2.2.5. Nesne tanımda kullanılan CNN modelleri	58
3.6.2.3. CNN modellerinin eğitimi.....	58
3.6.2.3.1. Topoloji parametreleri.....	59
3.6.2.3.2. Eğitim parametreleri.....	60
3.6.3. Büyük Dil Modelleri (LLM) ile Termal Görüntülerde Yüz Tanıma ve Sıcaklık Analizi	61

4. TERMAL GÖRÜNTÜLERDE ATEŞ TESPİTİNDE KLASİK/MODERN METOTLAR VE GENİŞ DİL MODELİ YAKLAŞIMI	64
4.1. Haar Cascade Yaklaşımı	64
4.1.1. Çalışmanın özeti.....	64
4.1.2. Önerilen Haar Cascade modeli.....	65
4.1.2.1. Modelin Yüklenmesi.....	67
4.1.2.2. Görüntünün Hazırlanması	68
4.1.2.3. Parametre Ayarları ve Yüz Tespiti.....	68
4.1.2.4. Tespit Edilen Yüzlerde Sıcaklık Analizi.....	68
4.1.2.5. Görselleştirme	68
4.1.3.6. Deneysel Çalışmalar ve Sonuçlar.....	70
4.1.3.6.1. scaleFactor = 1.01 ve minNeighbors = 1	70
4.1.3.6.2. scaleFactor = 1.05, minNeighbors = 5	72
4.1.3.6.3. scaleFactor = 1.05, minNeighbors = 25	74
4.2. Derin Öğrenme ve Konvolüsyonel Sinir Ağları Tabanlı Derin Öğrenme Modeli...	77
4.2.1. Çalışmanın özeti.....	79
4.2.2. Önerilen Derin Öğrenme Modeli	79
4.2.2.1. Modelin Yüklenmesi.....	79
4.2.2.2. Görüntü İşleme.....	80
4.2.2.3. Yüz Tespiti ve Sıcaklık Analizi	80
4.2.2.4. Deneysel Çalışmalar ve Sonuçlar.....	80
4.3. Büyük Dil Modeli (LLM)	84
4.3.1. Çalışmanın özeti.....	84
4.3.2. Önerilen LLM Modeli.....	86
4.3.2.1. Görüntü İşleme.....	87
4.4. Deneysel Çalışmalar ve Sonuçlar.....	88
4.4.1. Yüz Tanıma Konusunda Elde edilen sonuçlar	88
4.4.1.1. Haar Cascade Modeli ile Elde edilen sonuçlar	88
4.4.1.2. Derin Öğrenme ve Konvolüsyonel Sinir Ağları Tabanlı Derin Öğrenme Modeli ile Elde Edilen Sonuçlar.....	91
4.4.1.3. Büyük Dil Modeli (LLM) ile Elde Edilen Sonuçlar	94
4.3. Önerilen Modellerin Karşılaştırılması.....	96
4.3.2. Yüz Tespiti Konusunda Elde edilen sonuçlar	96
4.3.2. Sıcaklık Analizi.....	101
5. SONUÇLAR VE ÖNERİLER	104
5.1. Sonuçlar	104
5.1.1. Haar Cascade Modeli Sonuçları.....	104

5.1.2. Deep Learning Modeli Sonuçları	105
5.1.2. Büyük Dil Modeli (LLM) Sonuçları	107
5.1.3. Çoklu yüz tespitinde yaşanan sorunlar	109
5.2. Tartışma	111
5.2.1 Haarcascade Yönteminin Performans Tartışması	111
5.2.2 Derin Öğrenme Yönteminin Performans Tartışması	112
5.2.3 Large Language Models (LLM) Performans Tartışması	112
5.2.4 Tekil ve Çoklu Yüzlerin Değerlendirilmesi	113
5.2.5 Genel Değerlendirme	113
5.3. Öneriler	114
KAYNAKÇA	116
EKLER	127
ETİK KURUL KARARI	133
ÖZGEÇMİŞ	134

TABLolar DİZİNİ

Tablo 1. Görüntü İşleme ve Bilgisayarlı Görü Arasındaki Temel Farklar (Forsyth & Ponce, 2012).	9
Tablo 2. İnsan beyni ve bilgisayarlar arasındaki işlem performansının karşılaştırılması (Bektaş, 2018).	13
Tablo 3. Yapay ve biyolojik sinir ağlarının karşılaştırılması (Öztürk & Şahin, 2018)..	14
Tablo 4. Python kodu (URL-1).	30
Tablo 5. Her termal çekim türüne ait görüntü sayıları	40
Tablo 6. Literatürde yaygın olarak kullanılan önceden eğitilmiş CNN mimarilerinin bazı özellikleri	54
Tablo 7. Haar Cascade Mimarisi ve Katmanları	65
Tablo 8. Önerilen Model Katmanları ile İlgili Uygulama Adımları Akış Şeması	67
Tablo 9. Haar Cascade modeli sonucu elde edilen karmaşıklık matrisleri	88
Tablo 10. Her sınıf için temel istatistiksel performans metrik hesap sonuçları	90
Tablo 11. Derin Öğrenme Modeli Sonucu Elde Edilen Karmaşıklık Matrisi	92
Tablo 12. Her sınıf için temel istatistiksel performans metrik hesap sonuçları	92
Tablo 13. Büyük Dil Modeli (LLM) Sonucu Elde Edilen Karmaşıklık Matrisi	94
Tablo 14. Her sınıf için temel istatistiksel performans metrik hesap sonuçları	95
Tablo 15. Haar Cascade modeli ile elde edilen karmaşıklık matrisleri	97
Tablo 16. Her sınıf için temel istatistiksel performans metrik hesap sonuçları	98
Tablo 17. Sıcaklık Tespitinde Model Performans Analizi	103

ŞEKİLLER DİZİNİ

Şekil 1. Virüs yapısı (Zhou, vd., 2019; Aydın, 2023).....	10
Şekil 2. Sürüntü testi uygulama örneği (Aydın, 2023).....	11
Şekil 3. Yapay zeka venn şeması gösterimi (Aydın, 2023).	12
Şekil 4. Basitleştirilmiş şekliyle a) biyolojik ve b) yapay sinir ağı (Bektaş, 2018).	14
Şekil 5. Üç katmanlı yapay sinir ağı sisteminde geri yayılım diyagramı.....	15
Şekil 6. Konvolüsyonel sinir ağı katmanları (Hu, 2015; Akın ve Şahin, 2024)	19
Şekil 7. Konvolüsyonel katmanı işlem basamakları (Akın ve Şahin, 2024).....	19
Şekil 8. Relu fonksiyonu kullanarak düzleştirme katmanı ve Maksimum havuzlama yöntemi (Akın ve Şahin, 2024).....	20
Şekil 9. Tam bağlantı katmanı	21
Şekil 10. Haar Cascade Özellikleri: Kenar (üstte), Çizgi (ortada), Dört kare (altta)	29
Şekil 11. Haar Cascade ile Yüz Okuma Örneği	30
Şekil 12. Python resmî web sitesi (Downloads sekmesi).....	32
Şekil 13. Python'ın uygun bir sürümünün indirilmesi	32
Şekil 14. Python yükleme ekranı	33
Şekil 15. Python yüklendikten sonra kurulumu sonlandırma	33
Şekil 16. PowerShell logosu	34
Şekil 17. PowerShell'de OpenCV klasörü oluşturmak	34
Şekil 18. Sanal ortamın aktif hâle getirilmesi	35
Şekil 19. Visual Studio Code logosu	36
Şekil 20. SmartView Programının indirme	37
Şekil 21. SmartView Programının indirilmesi	37
Şekil 22. SmartView Uygulamasında Resmin açılması.....	37
Şekil 23. SmartView Uygulamasında Resmin özellikleri.....	38
Şekil 24. SmartView Uygulamasında Sıcaklık haritasının oluşturulması	38
Şekil 25. SmartView Uygulamasında RGB Resim Oluşturmak	39
Şekil 26. Termal görüntü örnekleri	40
Şekil 27. YZ, MÖ ve DÖ'nün zaman içerisindeki gelişimi	44
Şekil 28. CNN mimarisinin genel yapısı.....	45
Şekil 29. CNN'nin üç önemli mekanizması (Yerel alıcı alan, ağırlık paylaşımı, boyut indirgeme) (Shen et al., 2017).....	46
Şekil 30. Katmanındaki filtrelerin konvolüsyonu	48
Şekil 31. Boyut hesabında kullanılan değişkenler	49
Şekil 32. ReLU katmanının çıktı verisine etkisi	50

Şekil 33. 4×4 giriş görüntüsüne maksimum havuzlama için bir ve iki adımlı kaydırmalarla 2×2 filtre uygulamanın matematiksel temsili	51
Şekil 34. 4×4 giriş görüntüsüne maksimum havuzlama için bir ve iki adımlı kaydırmalarla 2×2 filtre uygulamanın matematiksel temsili	51
Şekil 35. CNN ağında dropout katmanının uygulanması (Wang et al., 2020a-b)	52
Şekil 36. Blok İstif yapıları (VggNet mimari örneği) (Simonyan ve Zisserman, 2014). 54	
Şekil 37. Kalıntı blok yapısı (He vd., 2016)	55
Şekil 38. Başlangıç blok yapısı	56
Şekil 39. 1×1 konvolüsyon işleminin boyut düşürmek için kullanılması	56
Şekil 40. Büyüme Oranı $k = 4$ ve 5 Katmandan Oluşan Yoğun Bir Blok (Huang vd., 2017) (Her katman tüm önceki özellik haritalarını girdi olarak alır).....	57
Şekil 41. Ateşleme blok yapısı (SqueezeNet mimarisi örneği).....	57
Şekil 42. (a) Görüntü sınıflandırma, (b) yüz lokalizasyonu ve (c) yüz tespiti	58
Şekil 43. Haar eğitim süreci aşağıdaki	66
Şekil 48. Haar Cascade çoklu yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)	70
Şekil 49. Haar Cascade tekli yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)	71
Şekil 50. Haar Cascade yatay bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)	71
Şekil 51. Haar Cascade ile yatay ve dikey bakan insan yüzü tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)	72
Şekil 52. Haar Cascade yana bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)	72
Şekil 53. Haar Cascade çoklu yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)	73
Şekil 54. Haar Cascade ile yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)	73
Şekil 55. Haar Cascade yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)	73
Şekil 56. Haar Cascade ile yatay ve dikey bakan insan yüzü tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)	74
Şekil 57. Haar Cascade ile yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)	74
Şekil 58. Haar Cascade çoklu yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)	75
Şekil 59. Haar Cascade ile yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)	75
Şekil 60. Haar Cascade ile hafif yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)	75

Şekil 61. Haar Cascade ile yatay ve dikey bakan insan yüzü tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)	76
Şekil 62. Haar Cascade ile yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)	76
Şekil 63. Ivankhnenko A. tarafından eğitilen bilinen ilk derin ağ mimarisi (Şeker vd., 2017).	77
Şekil 64. Deep Learning yana bakan yüz tespiti ve sıcaklık analizi	80
Şekil 65. Deep Learning hafif yana bakan yüz tespiti ve sıcaklık analizi.....	81
Şekil 66. Deep Learning hafif gülen yüz tespiti ve sıcaklık analizi.....	81
Şekil 67. Deep Learning yüz tespiti ve sıcaklık analizi	82
Şekil 68. Deep Learning yüz tespiti ve sıcaklık analizi	82
Şekil 69. Deep Learning çoklu yüz tespiti ve sıcaklık analizi	83
Şekil 70. Deep Learning hafif yana bakan ve ileri bakan yüz tespiti ve sıcaklık analizi	83
Şekil 71. Deep Learning yana bakan yüz tespiti ve sıcaklık analizi	84
Şekil 72. Deep Learning hafif yana ve ileri bakan yüz tespiti ve sıcaklık analizi	84
Şekil 73. Chat GPT üzerinden görüntü öğretimi.....	87
Şekil 74. Chat GPT üzerinden görüntü işleme.....	87
Şekil 75. Chat GPT üzerinden çoklu görüntü işleme.....	88
Şekil 76. Modelin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları .	91
Şekil 77. Termal görüntü çeşitlerine göre modelin tanıma başarısı.....	91
Şekil 78. Modelin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları .	93
Şekil 79. Termal görüntü çeşitlerine göre modelin tanıma başarısı.....	94
Şekil 80. Modelin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları .	96
Şekil 81. Termal görüntü çeşitlerine göre modelin tanıma başarısı.....	96
Şekil 82. Önerilerin modellerin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları.....	99
Şekil 83. Termal görüntü çeşitlerine göre modelin tanıma başarısı.....	100
Şekil 84. Modellerin eğitim sırasındaki minibatch doğruluğu.....	100
Şekil 85. Modellerin eğitim sırasındaki eğitim sürecindeki RMSE değerleri	100

EKLER DİZİNİ

Ek 1: Haar Cascade Yüz Tespiti.....	127
Ek 2: Deep Learning Yüz Tespiti	128
Ek 3: Deep Learning Infrared Resim Yüz ve Sıcaklık Tespiti	129
Ek 4: Haar Cascade Infrared Resim Yüz ve Sıcaklık Tespiti.....	132



SİMGELER VE KISALTMALAR DİZİNİ

CV	: Computer Vision
AI	: Artificial Intelligence
JS	: JavaScript
Go	: Golang
OpenCV	: Open Computer Vision
API	: Application Programming Interface
pdf	: Probability Density Function



1. GİRİŞ

Pandemiler, tarih boyunca birçok örneği bulunan ve insanlığın en büyük zorluklarından biri olan olgulardır. Bu tür bulaşıcı hastalıklarla mücadelede en kritik adım, hastalığa sahip bireylerin mümkün olan en kısa sürede tespit edilmesi ve böylece bulaşmanın önlenmesidir (Hays, 2005).

Hemen hemen tüm pandemi hastalıklarının ortak bir belirtisi vardır: yüksek ateş. Bu nedenle, 38 °C veya daha yüksek vücut sıcaklığına sahip bireylerin tespit edilmesi, ki bu durum basitçe anormal vücut sıcaklığı olarak tanımlanır, bulaşma riskini ortadan kaldıracak ve salgının kontrol altına alınmasına yardımcı olacaktır. Ancak, yüzlerce kişinin giriş çıkış yaptığı ortamlarda bu tür bireylerin izlenmesi zordur. Bu süreç, insan gücü gerektirir, maliyet oluşturur ve zaman kaybına neden olur. Bu nedenle, günümüzün gelişmiş teknolojik dünyasında, insan müdahalesini ortadan kaldıracak şekilde bu süreci otomatikleştirmek gereklidir (Cai vd., 2020).

İnsanlık, tarih boyunca çeşitli seviyelerde pandemilerle karşılaşmıştır ve bunlardan bazıları insanlık için son derece yıkıcı olmuştur. 2020 yılında insanlık, benzer türden bir zorlukla, Yeni COVID-19 koronavirüsü şeklinde yeniden karşı karşıya kalmıştır. Bu hızlı bir şekilde yayılan, güçlü ve görünmez düşman ilk olarak Çin Halk Cumhuriyeti'nin Wuhan eyaletinde tanımlanmıştır. 31 Aralık 2019'dan itibaren Çin'de hızla ilerleyen bu zatürre salgını, birçok ülkeye de yayılmıştır. Pandemi, milyonlarca insanın hayatını köklü bir şekilde değiştirmiş; aktiviteler, seyahatler ve sosyal temaslar ciddi şekilde kısıtlanmıştır.

Bir virüs, yeni konaklar aramak üzere evrimleşmiş bir yaşam formudur; çünkü taşıyıcıları ölmekte ve varlığını sürdürebilmesi için her zaman ölümün bir adım önünde olması gerekmektedir. COVID-19, zatürre benzeri bir hastalık oluşturan bir virüstür. Bu hastalık, SARS-CoV-2 olarak adlandırılan yeni bir koronavirüs tarafından meydana gelmektedir ve Şiddetli Akut Solunum Yolu Sendromu (SARS) hastalığına neden olan virüsle benzerdir. Virüsün yayılmasının ardından bir yıl bile olmadan, 20 Ekim 2020

itibarıyla, dünya genelinde 40.785.821 koronavirüs vakası tespit edilmiş; 1.124.972 kişi hayatını kaybederken 30.450.704 kişi iyileşmiştir.

Bu virüsün en belirgin özelliklerinden birisi yüksek ateştir. Virüsün tespiti için öncelikle ateş kontrolü yapılmakta, daha sonrasında ise belirtiler var ise burun sürüntüsü alınmaktadır. Pandemi döneminde en belirgin özellik olan ateşin ölçülmesinde genel olarak sağlık kurumlarında temassız ateş ölçerlerle ve termal kameralarla ölçümler yapılırken kamusal alanların bir çoğunda termal kameralara başvurulmuştur. Termal kameralar, elektrik ekipmanlarının, süreç ekipmanlarının, sağlık sektörünün, savunmanın ve bina teşhislerinin incelenmesi için kullanılabilir. Sağlık sektöründe yukarıda belirttiğimiz gibi özellikle kamusal alanlarda bireylerin sıcaklıklarının tespitinde, bazı kanser hastalıklarının tespitinde (meme kanseri, cilt kanseri. vs), diyabetik ayak teşhisinde, Periferik Arter Hastalığının (PAH) teşhisinde, oftalmoloji alanında, flep (doku) canlılığı teşhisinde, Yara ve Cerrahi Alan Enfeksiyonlarında (CAE), ortopedi, jinekoloji ve romatoloji alanları gibi daha bir çok alanda kullanılmaktadır (İlçe ve Demir, 2024).

Literatür incelendiğinde de pandemi döneminde yayımlanan çeşitli araştırmalar, yüksek vücut sıcaklığına sahip bireylerin diğer bireyleri enfekte etme riskini önlemek amacıyla, özellikle enfeksiyon riskinin yüksek olduğu dönemlerde kamusal alanlarda termal kameraların ateş taraması için kullanılmasını ele almıştır (Kumar, vd., 2020; Mohammed, vd., 2020; Maghdid, vd., 2020; Huang; vd 2020; Bwire ve Paulo, 2020). Termal kameralar, canlı varlıkları vücutlarından yayılan ısı yoluyla tespit eder. Kısaca, bu kameralar, canlıların yaydığı vücut ısını kullanarak hareketlerini algılar ve konumlarını belirler. Termal kameralar, son derece küçük sıcaklık farklarını algılayabilir ve bunları gerçek zamanlı video görüntülerine dönüştürerek bir ekranda izlenebilir hale getirir. Tam gece görüşü ve benzer zorlu uygulamalar için en uygun ürünlerdir. Sıcak nesnelerin görüntülerini yakalamak için tasarlanmış iki tür termal kamera bulunmaktadır (Faundez-Zanuy ve Mekyska, 2011). Bu kameralar arasındaki fark, sensörlerinin yapısında gizlidir. İlk grup, uzun dalga kızılötesi aralığında çalışan soğutmasız sensörlere sahip kameralardan oluşur. İkinci grup ise kriyojenik soğutma tekniğiyle soğutulmuş sensörler kullanır. Bu kameralar, orta dalga ve uzun dalga bantlarında görüntü alabilir. Termal dalga yöntemleri, bir katı nesnenin termal

özellikleri ile bir akışkanın zaman veya mekân boyunca ısı akışı arasındaki ilişkiyi ortaya çıkarmak için kullanılır. Bir nesnenin termal görüntüleri hakkındaki bilgiler, zaman veya frekans alanındaki sıcaklık tepkisinden elde edilir (Kosikowski ve Suszynski, 2011; Çalışan ve Türkoğlu, 2011). Termal dalga yöntemleri kullanılarak, bir nesnenin yüzeyindeki termal düzensizlikler görselleştirilir ve dinamik sıcaklık alanlarının tanısı yapılır. Nesne üzerindeki sıcaklık farklarını ayarlamak için enerji uyarımının frekans spektrumu (aralığı) dikkate alınır (Şenalp ve Ceylan, 2023).

Yaygın kullanım alanına sahip olmasına karşın düşük maliyetli termal görüntüleme sistemleri detay bilgisi yeterli olmayan bulanık görüntüler meydana getirmektedirler. Bundan dolayı termal görüntülerin çözünürlüğünün iyileştirilmesine yönelik çalışmalar, görüntü işleme alanında odak noktası haline gelmiştir. Bunlar arasında en yaygın olanı ise derin öğrenmedir. Birçok araştırmacı derin öğrenme yöntemini kullanarak düşük çözünürlüklü ve bulanık görüntüler arasından bireylerin tespitini daha iyi sağlayabilmek için çalışmalar yapmışlardır. Bu konuda bir çok farklı evrişimli sinir ağları (CNN) tabanlı modeller geliştirilmiştir (Dong, vd., 2016; Kim vd., 2016; Lim, vd., 2017; Şenalp ve Ceylan, 2023). Bu çalışma da aynı odak üzerine yoğunlaşarak termal kameralarla elde edilen görüntülerde yüz tespitinin gerekliliği ön planda tutulmuştur.

Bu çalışma, termal kameralarda yüzlerin verimli ve kolay bir şekilde tespit edilmesi ve belirlenen yüzlerde sıcaklık ölçümlerinin yapılması için Haar Cascade Modeli, Derin Öğrenme Modeli ve Büyük Dil Modeli (LLM) yüz algılama sistemleri önermektedir. Bu sayede pandemilerin günlük yaşam akışı üzerindeki olumsuz etkileri hafifletilebilir ve kesintiler en aza indirilebilir.

1.1. Amaç

Konvolüsyonel Sinir Ağları (CNN), derin yapılarında güçlü temsil yetenekleri ile bilinir ve birçok araştırma alanında giderek artan bir popülerlik kazanmıştır. Özellikle Derin Öğrenme (DL) yaklaşımlarının görüntü tabanlı sorunları çözmede sağladığı hız ve performans artışları, çeşitli zorlukların üstesinden gelmede önemli katkılar sağlamıştır.

Görüntü analizinde sınıflandırma, nesne sınır değerlerini belirlemek için segmentasyon ve nesne tanıma gibi sorunların çözümüne yönelik çeşitli algoritmalar geliştirilmiştir.

Daha derin ve geniş CNN mimarilerinin oluşturulması her zaman daha yüksek bir performans garantisi sağlamaz. Literatürde yaygın olarak kullanılan önceden eğitilmiş CNN modellerine bakıldığında, Konvolüsyonel katmanların derinliği arttıkça, muazzam hesaplama karmaşıklığının sınırlı donanım kaynaklarına sahip sistemlerde çalışmayı zorlaştırdığı görülmektedir. Bu nedenle etkili modeller tasarlamak, başarılı mimarilerin geliştirilmesine bağlıdır. Bu bağlamda, ağ eğitimi stratejilerinin değiştirilmesi, etkili konvolüsyon filtrelerinin kullanılması, verimli CNN mimarilerinin tasarlanması ve ağ eğitimi sırasında öğrenilecek parametre sayısını azaltacak yaklaşımların geliştirilmesi, CNN’lerde hesaplama maliyetlerini düşürmekte, performansı artırmakta ve eğitim sürelerini kısaltmaktadır. Bu tez çalışması, farklı yaklaşımlar ve yeni oluşturulmuş termal görüntü veri setlerini içeren özgün bir çalışma sunmaktadır. Bu tezin amacı, CNN’lerde ağ eğitiminin iyileştirilmesi için çeşitli yaklaşımlar geliştirmek, bu yaklaşımları özgün olarak hazırlanmış veri setlerinde test etmek ve deneysel sonuçları analiz etmektir. CNN mimarilerinde ağ derinleştikçe öğrenilecek parametre sayısının arttığı göz önüne alındığında, bu çalışma, ağı derinleştirmeden performansı artıracak alternatif model tasarımları geliştirmeye odaklanmıştır.

Özellikle tıbbi alanda veri elde etme teknolojilerinin yaygınlaşmasıyla birlikte, elde edilen görüntülerin sayısı da artmıştır. Uzmanlar tarafından değerlendirilmesi zaman alan görüntü analiz işlemleri önemli bir yük oluşturmaktadır. Değerlendiren uzmanların bilgi seviyesine bağlı olarak öznel çıkarımlar yapılabileceği düşünüldüğünde, daha kısa sürede daha fazla görüntü analizini gerçekleştirebilecek uzman sistemlere ihtiyaç duyulmaktadır. CNN’lerde ağ eğitiminin iyileştirilmesine yönelik bu tezde önerilen yaklaşımlar, yalnızca performansı artırmakla kalmayıp daha hızlı sonuçlar da üretmektedir. Ayrıca tez kapsamında termal yüz görüntülerinden oluşan özgün veri setlerinin oluşturulması, literatüre katkı sağlamaktadır.

1.2. Tezin Organizasyonu

Bu tez çalışmasında, birinci bölümde tezin amacı ve önemi açıklanmıştır. İkinci bölümdeki literatür incelemesi kısmında temel kavramlar ele alınarak termal görüntüleme, bilgisayarlı görü, COVID-19 ve pandemi, yapay zeka, yapay sinir ağları, derin öğrenme, öğrenme aktarımı ve derin öğrenme yöntemiyle termal kameraların verimliliğini artıran çalışmalar incelenmiştir. Tezin üçüncü bölümünde kullanılan materyaller ve yöntemlere dair bilgiler verilmiş, CNN’lerde ağ eğitiminin iyileştirilmesine yönelik önerilen yaklaşımlar hakkında detaylar sunulmuş ve deneysel çalışmalar açıklanmıştır. Bu bölümde öncelikle performans ölçütlerine ilişkin gerekli açıklamalar yapılmış, ardından önerilen üç farklı yaklaşım için gerçekleştirilen deneysel çalışmalara ait temel bilgiler paylaşılmıştır. Tezin dördüncü bölümünde ise önerilen yaklaşımların sonuçları diğer yöntemlerin sonuçlarıyla karşılaştırılmış, tartışmalar yapılmış ve tez kapsamında elde edilen sonuçlar ile öneriler aktarılmıştır.

2. TEMEL KAVRAMLAR

2.1. Termal Görüntüleme

Bir maddenin ulaşabileceği varsayılan en düşük sıcaklığın mutlak sıfır (-273,15 °C) olduğu bilinmektedir. Mutlak sıfırın üzerinde sıcaklığa sahip tüm maddelerin molekülleri titreşim halindedir ve termal radyasyon yayar. İnsan gözü, belirli bir bandın içerisindeki radyasyonu algılayabilir; bu bant, görünür ışık olarak bilinir (dalga boyları 0,4 µm ile 0,7 µm arasında). Görünür ışık, doğada mevcut olan radyasyonun yalnızca küçük bir kısmını oluşturur (Djajakusumah vd., 2023). Öte yandan, kızılötesi radyasyon, dalga boyları 0,7 µm ile 1000 µm arasında değişen ve insan gözünün algılayamadığı bir radyasyon türüdür. Oda sıcaklığına yakın nesneler tarafından yayılan termal radyasyonun büyük bir kısmı kızılötesi bantta yer alır. Kızılötesi biyotermal radyasyon ise 8–14 µm dalga boylarında bulunur. Normal koşullarda, 36,6 °C vücut sıcaklığına sahip bir insan, 9,7 µm dalga boyunda termal radyasyon yayabilir (İlçe & Demir, 2024).

Termal görüntüleme sistemleri, donanım ve yazılım aracılığıyla, nesneler tarafından yayılan radyasyonu—özellikle insan gözünün göremediği frekansların ötesindeki kızılötesi ışınları—tespit eder ve bu ışınları insan gözünün algılayabileceği görüntülere dönüştürür (Çalışan & Türkoğlu, 2011). İlk olarak, gözlemlenen ortamdan toplanan kızılötesi ışınlar, lensler, aynalar ve prizmalardan oluşan bir optik sistemden geçerek, çok küçük sıcaklık farklarına karşı oldukça hassas olan bir kızılötesi sensöre yönlendirilir. Kızılötesi sensör, gelen kızılötesi ışınlar nedeniyle ısınarak ya da doğrudan kızılötesi fotonların yardımıyla elektrik sinyalleri üretir. Bu elektrik sinyalleri, amplifikatörler kullanılarak güçlendirilir ve ardından sistemin "beyni" olan elektronik bir sinyal işleme ünitesinde işlenir. Son olarak, bu sinyaller bir ekrana yansıtılarak insan gözüyle algılanabilir görüntülere dönüştürülür (Zhang vd., 2018).

Termal kameraların kızılötesi sensörleri, hücrelerden oluşan bir matris şeklinde tasarlanmıştır ve bu hücreler tarafından toplanan elektrik sinyalleri, nihai görüntünün üretilmesi için birçok işlemden geçirilir. Özetle, kızılötesi sensör matrisinin her bir

hücresi, ortaya çıkan görüntüdeki bir piksele karşılık gelir ve bu şekilde görüntü oluşur. Dijitalleştirilmiş görüntüler, pikseller adı verilen hücrelerden oluşur ve piksel yoğunluğu doğrudan görüntü kalitesini etkiler (İlçe & Demir, 2024).

2.1.1. Termal Kamera Kullanımını Sınırlayan Faktörler

Termal kamera kullanımında ölçüm hatalarını en aza indirmek için hava koşulları, görünürlük, mesafe ve hayvanların fiziksel aktivitesi gibi sınırlayıcı faktörlerin dikkate alınması önemlidir.

Hava Koşulları: Güneş ışığı, yağış, nem, sis, kar ve toz gibi etmenler, hayvanların vücut yüzeyi sıcaklıklarının artmasına veya azalmasına neden olabilir ve bu da ölçüm sonuçlarını olumsuz etkileyebilir. Termal görüntüleme kameraları bir ahıra sabitlenmişse, ortamda hava hareketine neden olan ekipmanların (klima, fan, vb.) veya ısıtıcıların bulunmaması gerekir. Araştırmalar, hayvanların gündüzleri daha aktif olduğunu ve bunun, üretilen ısı nedeniyle termal görüntüleme hatalarına yol açabileceğini göstermiştir. Bu nedenlerle, gece yapılan termal görüntülemenin, gündüz yapılan görüntülemeden daha doğru sonuçlar verdiği bulunmuştur (Cilulko vd., 2013). Golzarian vd. (2017), süt verimi ile SHS arasında negatif bir korelasyon ($r = -0,97$) bildirmiş ve sağlıklı ve sağlıklı meme uçları arasındaki ortalama sıcaklık farkının $0,44^{\circ}\text{C}$ olduğunu, termal görüntüleme doğruluğunun ise %57,3 olduğunu belirtmiştir. Düşük doğruluk oranının nedeni, meme dokusuna ulaşan ışığın, meme yüzeyindeki tüylerin veya cilt lezyonlarının küçük sıcaklık farklarını etkileyen etkisi olarak gösterilmiştir. Bu nedenle, bu faktörlere ve memenin karna temas ettiği bölgelere dikkat edilmesi gerekmektedir.

Mesafe: Hayvan ile termal görüntüleme kameraları arasında birkaç metre mesafe olduğunda doğru ölçümler yapılabilir. Güneş ışığından kaynaklanan hataları en aza indirmek için kameranın görüş açısı, incelenen alanda 50° ile 90° arasında ayarlanmalıdır (Minkina, 2004).

Fiziksel Aktivite: Görüntülemeden önce yapılan fiziksel aktiviteler (koşma veya egzersiz gibi) vücut yüzeyi sıcaklığını artırır. Hataları en aza indirmek için, hayvanlar

fiziksel aktivite yapmamalı ve ölçüm öncesinde görüntülenecek alanda tutulmalıdır (Cilulko vd., 2013).

Bitki Örtüsü: Eğer hayvan ile termal kamera arasında yoğun bitki örtüsü varsa, hayvan tarafından yayılan ısı termal kamera tarafından tespit edilemez ve bu da görüntüyü engeller. Bir araştırma, bitki örtüsünün yoğunluğu ve kalınlığının yanı sıra alandaki diğer hayvanların varlığının da hayvan ile termal kamera arasındaki görüşü engelleyebileceğini ve kameranın etkinliğini sınırlayabileceğini bulmuştur. Bu nedenle, seyrek bitki örtüsüne sahip alanlar termal görüntüleme analizi için daha uygun ortamlardır (Ditchkoff vd., 2005; Butler vd., 2006; Cilulko vd., 2013).

Tüy Yapısı: Tüy yapısı, vücut yüzeyi sıcaklığını önemli ölçüde etkiler ve ince tüy yapısına sahip hayvanlar, kalın tüylü hayvanlara göre termal görüntüleme için daha uygun kabul edilir. Ölçüm hatalarını en aza indirmek için, görüntülenecek yüzey temizlenmeli ve analizden önce kurumasına izin verilmelidir (Hilsberg-Merz, 2008; Cilulko vd., 2013).

Çevresel ve Deri Nemliliği: Çevre ve deri yüzeyinin nem seviyesi, hayvanın verimlilik seviyesi ve fizyolojik durumu ile sağım ve beslenme zamanları, meme yüzeyi sıcaklığını etkileyen faktörler olarak kabul edilmektedir (Colak vd., 2008).

2.2. Bilgisayarlı Görü

Bilgisayarla görme, mühendislik ve bilim alanlarında giderek artan öneme sahip olan Bilgisayarla görme, mühendislik ve bilim alanlarında giderek artan öneme sahip olan disiplinler arası bir alandır. Bu alan, kameralar veya diğer görüntüleme cihazları tarafından toplanan görsel verilerden anlamlı bilgiler çıkarmayı amaçlar. Yüz tespiti, nesne tanıma ve hareket analizi gibi uygulamalar, bilgisayarla görmenin önde gelen alanlarındandır (Forsyth & Ponce, 2012).

Bilgisayarla görme, sıklıkla başka bir odak alanı olan görüntü işleme ile karıştırılmaktadır. Görüntü işleme, genellikle bir görüntünün iyileştirilmesi veya

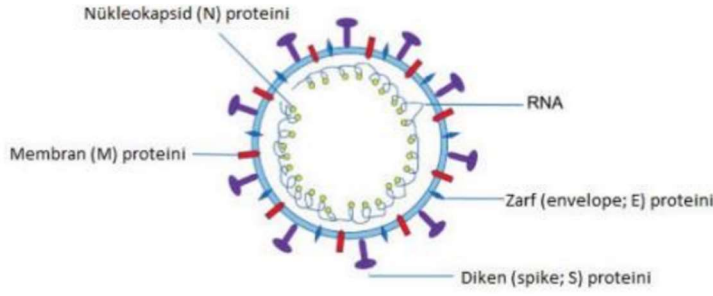
dönüştürülmesiyle ilgili sınırlı bir süreçtir. Buna karşılık, bilgisayarla görme, görüntülerden bilgi çıkararak daha ileri düzeyde analiz ve karar verme süreçlerine katkıda bulunur (Jain, 1989). Örneğin, görüntü işleme bir görüntüdeki gürültüyü azaltmaya odaklanırken, bilgisayarla görme aynı görüntüdeki yüzleri tanımlayıp sınıflandırabilir. Tablo 1, bu iki alan arasındaki temel farkları açıkça göstermektedir.

Tablo 1. Görüntü İşleme ve Bilgisayarlı Görü Arasındaki Temel Farklar (Forsyth & Ponce, 2012).

Görüntü İşleme	Bilgisayarlı Görü
Giriş: Görüntü	Giriş: Görüntü
Çıkış: Görüntü	Çıkış: Bilgi
Örnek: Gürültü giderme	Örnek: Yüz tespiti

2.2. COVID-19

Latince’de "virüs" terimi, zehir anlamına gelir ve bakterilerden farklı olarak yalnızca enfekte ettikleri hücrelerde yaşam belirtileri gösteren yapılara atıfta bulunur. Hücre dışında, virüsler herhangi bir yaşam belirtisi göstermez ve çoğalabilmek için canlı bir hücreye ihtiyaç duyar. Virüsler, yaşamı sona erdirebilecek kadar ciddi olumsuz etkiler yaratma potansiyeline sahiptir. Virüsler genellikle belirli konak hücrelere sahiptir; ancak, kuduz virüsü gibi bazı virüsler, hem insan hem de hayvan hücrelerini konak olarak kullanabilir (Aydın, 2023). Virüsler ile bakteriler arasındaki diğer bir fark ise, virüslerin yalnızca konak hücre içinde parazit olarak var olmalarıdır. Boyutları 20 ila 300 nanometre arasında değişen virüsler çok küçüktür ve yalnızca elektron mikroskobu kullanılarak gözlemlenebilir. Bakterilerin aksine, virüsler antibiyotiklere duyarlı değildir, bu da onları çok daha dirençli kılar. Antibiyotiklerle tedavi edilemeyen viral hastalıklar için aşılar kullanılarak tedavi yöntemleri geliştirilmiştir. Virüsler, konak hücreye aktarılabilecek bilgiyi depolayan RNA veya DNA içeren genetik materyale sahiptir. Virüslerin nükleik asitleri, kapsomer adı verilen protein yapıları ile çevrilidir ve bu yapı, kapsid olarak bilinen bir yapıyı oluşturur. Spike proteini, virüsün konak hücreye bağlanmasını sağlar. Çevreleyen yapı ise, zarf adı verilen ve proteinler, lipidler ve karbonhidratlardan oluşan bir yapıdır (Zhou vd., 2019). Şekil 1, bir virüsün yapısını göstermektedir.



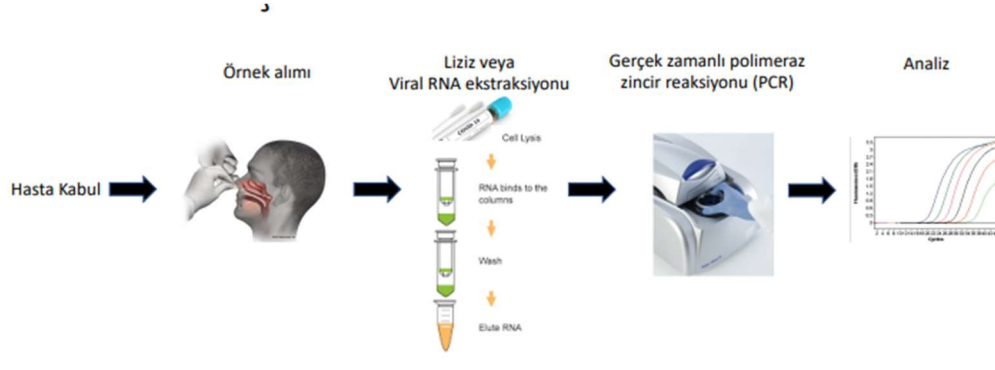
Şekil 1. Virüs yapısı (Zhou, vd., 2019; Aydın, 2023)

Tarih boyunca, birçok virüs türü keşfedilmiş ve bunların çoğu ölümcül olmamıştır. Koronavirüs ailesi, 1960'larda keşfedilmiş, 2003'te SARS virüsü, 2012'de MERS virüsü ve nihayetinde 2019'un sonunda Çin'in Wuhan şehrinde yeni bir koronavirüs hastalığı ortaya çıkmış, hızla dünyaya yayılmış ve pandemi olmuştur. Salgının başlangıçta bölgedeki deniz ürünleri ve hayvan pazarlarına dayandığı izlenmiş ve virüsün, solunum damlacıkları yoluyla insandan insana yayıldığı belirlenmiştir (Aydın, 2023).

Koronavirüs, tek iplikli, pozitif duyarlı, zarf kaplı RNA virüs ailesi olup, 80 ila 120 nm arasında bir boyuta sahip olup insan patojeni olabilme kapasitesine sahiptir. Bu virüslerin zoonotik olduğu ve yarasalar veya kemirgenler gibi hayvanlardan kaynaklandığı düşünülmektedir. Yapılan araştırmalar, COVID-19'un yarasalardan insanlara geçtiğini öne sürmektedir (Wu & McGoogan, 2020; Akkuş, 2020).

COVID-19 pandemisinin yayılmasını engellemek ve etkilerini azaltmak için, onun bulaşmasını önlemek hayati önem taşımaktadır. Enfekte olmuş bireylerin toplumdan izole edilmesi için hızlı tanı koyulması gereklidir. COVID-19 ile enfekte olan bireyleri teşhis etmek için bir PCR test yöntemi geliştirilmiştir. Test, COVID-19 belirtileri gösteren bir hastanın solunum yollarından bir sürüntü örneği almayı içerir. Test sırasında, hastanın başı 70 derece açıyla geriye doğru eğilir ve sürüntü, nazal geçişe yerleştirilir ve farengeal duvara kadar ulaşması sağlanır. Hedef noktaya ulaşıldığında, sürüntü birkaç saniye tutulur ve sonra yavaşça çıkarılır. Sürüntü örneği laboratuvarında test edilir ve eğer COVID-19 virüsünden RNA tespit edilirse, hasta

COVID-19 pozitif olarak teşhis edilir (Aydın, 2023). Şekil 2, sürüntü testi prosedürünün bir örneğini göstermektedir.



Şekil 2. Sürüntü testi uygulama örneği (Aydın, 2023).

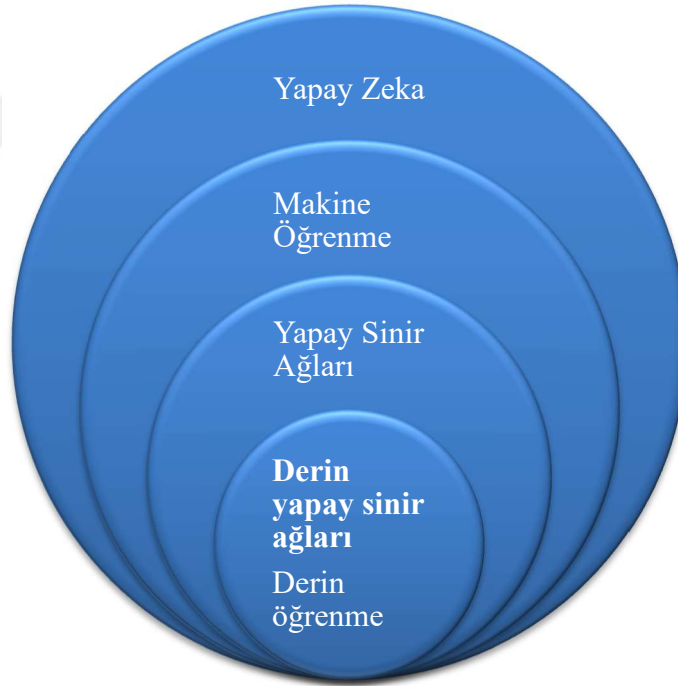
2.3. Yapay Zeka

Yapay zeka (YZ), bilgisayarların insan gibi düşünmesini sağlamak amacıyla geliştirilmiş bir sistemdir; böylece öğrenebilir, problem çözebilir ve karar verebilirler. YZ'nin temel amacı makineleri daha verimli ve etkili hale getirmektir. YZ, bilgisayar bilimi, matematik, istatistik, psikoloji ve sinirbilim gibi çeşitli disiplinlerin birleşimiyle oluşan geniş bir alandır. Modern anlamda YZ kavramı, Alan Turing'in 1950 tarihli makalesinde tanıtılmıştır. Bu makalede, zeki makinelerin nasıl oluşturulabileceği ve özellikle bu zekanın nasıl test edilebileceği incelenmiştir. Turing, bir makine insanların etkileşim sırasında insan benzeri algılanabiliyorsa, o makinenin zeki sayılabileceğini öne sürmüştür. Turing, bilgisayarlardaki zeki davranışı, insan seviyesinde bilişsel görevleri yerine getirme yeteneği olarak tanımlamış ve bu daha sonra Turing testi olarak bilinir hale gelmiştir (İnce vd., 2021).

Bugün, YZ birçok alanda kullanılan bir teknoloji haline gelmiştir. Örneğin, sesli asistanlar, görüntü tanıma sistemleri ve otonom sürüş teknolojileri, YZ uygulamalarına birkaç örnektir. YZ'nin, gelecekte daha karmaşık görevleri yerine getirebilmesi için gelişmeye devam etmesi beklenmektedir. 20. yüzyılın ortalarından bu yana, araştırmacılar YZ tekniklerinin tıbbi alandaki uygulamalarını araştırmaktadır. Tıbbi YZ'nin geliştirilmesi, kliniklerin tanı koymasına, tedavi kararları almasına ve sonuçları tahmin etmesine yardımcı olan YZ programlarının yaratılmasına odaklanmaktadır. Bu

sistemler, sađlık alıřanlarının gnlk iřlerinde destek sađlamak ve veri ve bilgi maniplasyonuna dayalı grevlerde yardımcı olmak amacıyla tasarlanmıřtır. YZ'nin, gelecekte farklı alanlarda daha karmařık grevleri yerine getirebilmesi beklenmektedir (nver & Altunok, 2020).

YZ'de kullanılan tekniklerden biri, veriler temelinde ğrenme ve karar verme yeteneđi sađlayan makine ğrenmesidir. Makine ğrenmesi, veri analizi ve istatistiksel yntemler kullanarak modeller oluřturur ve bu modelleri gelecekteki olayları tahmin etmek veya belirli grevleri yerine getirmek iin kullanır. YZ'nin bir dalı olan derin ğrenme, byk miktarda veriyi analiz etmek ve karmařık iliřkileri ğrenmek iin yapay sinir ađları olarak bilinen karmařık yapıları kullanır. Derin ğrenme algoritması, verileri yapay sinir ađları aracılıđıyla iřler ve karmařık desenleri tanıyıp anlar (Atalay & elik, 2017). řekil 3, yapay zekanın Venn diyagramını gstermektedir.



řekil 3. Yapay zekâ venn řeması gsterimi (Aydın, 2023).

2.4. Yapay Sinir Ađları

Yapay sinir ađları, insan beynini taklit ederek verileri yorumlayan, ğrenen ve ıktı reten veri iřleyicileri olarak tanımlanabilir. İlk kez, 1943 yılında McCulloch ve

Pitts tarafından bilgisayarlar kullanılarak formüle edilemeyen problemleri çözmede kullanılmıştır. Genellikle, bu bağlamda bilgisayarlar ve insan beyni karşılaştırılmıştır. İnsan beyni doğasında hesaplama yapabilme yeteneğine sahipken, bilgisayarlar, işlemciler, bellek birimleri ve algoritmik yapılar olmadan karmaşık hesaplamaları hızlı bir şekilde yapamazlar. Teorik olarak, bilgisayarların insan beyninden daha karmaşık bir yapıya sahip olması gerekir. Bilgisayarlar yaklaşık 10^9 transistöre ve 10^{-9} saniye işlem süresine sahipken, insan beyni yaklaşık 10^{11} nörondan oluşur ve işlem süresi 10^{-3} saniyedir. İnsan beyninin önemli bir kısmı sürekli çalışırken, bilgisayarlarda pasif veri deposu yalnızca gerektiğinde görevini yerine getirir (Bektaş, 2018). İşlem performansı ile saniye başına yapılan işlemlerin kalite seviyesi arasında doğrudan bir ilişki olmasa da, karşılaştırma amacıyla Tablo 2'de bu konu gösterilmiştir.

Tablo 2. İnsan beyni ve bilgisayarlar arasındaki işlem performansının karşılaştırılması (Bektaş, 2018).

Elemanlar	Bilgisayarlar	İnsan beyni
İşlemci sayısı	$\sim 10^9$	$\sim 10^{11}$
İşlemcinin tipi	İşlemci	Sinir ağları
Hesaplama tipi	Dizi halinde	Eş zamanlı
Veri depolama	Adres bazlı	İlişkisel
İşlem hızı	$\sim 10^9$	$\sim 10^{-3}$
Operasyonel çevrim hızı	$\sim 10^9$ 1/s	$\sim 10^{13}$ 1/s
Gerçek performans	$\sim 10^{10}$ 1/s	$\sim 10^{12}$ 1/s

Yapay sinir ağları verileri genelleştirir ve ilişkisel bir yapı oluşturur. Başarılı bir sinir ağı eğitiminin ardından, benzer problem çözme senaryoları için kullanılabilecek mantıklı çözümler üretir. Ayrıca, hata toleransı açısından biyolojik sinir ağı yapısıyla yakın bir ilişkiye sahiptir. Biyolojik sinir ağları, yıkıcı dış etkilere uyum sağlama yeteneğine sahip olup, kendini ayarlama için bir tolerans sunar. Örneğin, çok zor okunabilen bir el yazısını tanıyabilirler. Yapay sinir ağlarında, biyolojik sinir ağlarının aksine, her zaman temiz, net ve ölçülebilir veriler mevcut değildir. Bu nedenle, sistem için önceden tasarlanmış olan hata toleransı sisteme girilir ve ağ yapısı içindeki hesaplamalar bu tolerans kullanılarak tasarlanır, böylece çıktı elde edilir (Öztürk & Şahin, 2018).

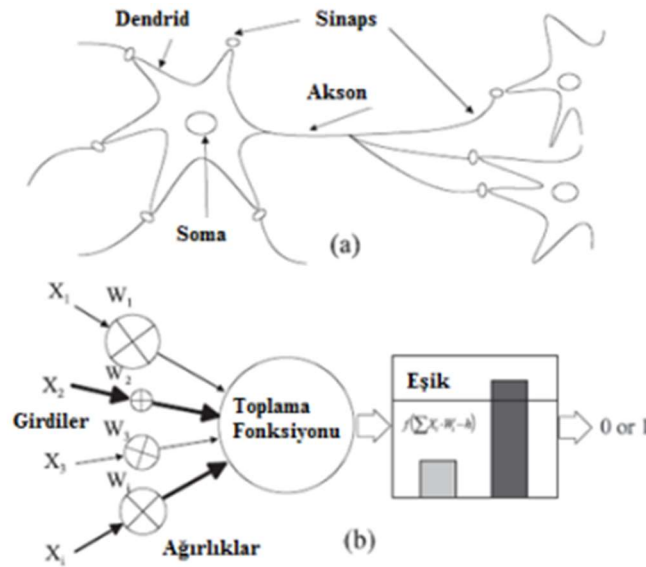
Yapay sinir ağlarının fonksiyonel öğeleri, insan beynini modelleyen yapılar, biyolojik sinir ağlarına benzerlik gösterir. Örneğin, biyolojik sinir ağlarındaki düğümler,

yapay sinir ağlarındaki işleme birimlerine benzer. Bu benzerlik, Tablo 3'te gösterilmiştir.

Tablo 3. Yapay ve biyolojik sinir ağlarının karşılaştırılması (Öztürk & Şahin, 2018).

Yapay sinir ağları	Biyolojik sinir ağları
Ağ hesaplama yöntemi	Sinir sistemi
İşlemci, düğüm noktası	Sinir hücreleri
Bağlantı ağırlıkları	Sinapslar
Toplama fonksiyonu	Dendridler
Aktivasyon fonksiyonu	Sinir gövdesi
Çıktı	Akson

Yapay sinir ağlarında, iletişim giriş sinyalleri aracılığıyla sağlanır. Şekil 4'te gösterildiği gibi, sistem, giriş sinyalini kabul ettiğinde başlar. Her giriş sinyali, ilişkili bağlantı ağırlık katsayısı ile çarpılır ve sonra toplam fonksiyona gönderilir. Eğer giriş veya sinyalin ağırlığı yeterince yüksekse, kritik eşiği aşarak aksonlar aracılığıyla sinapslara taşınır. Genellikle, eşik 1 veya 0 olarak değerlendirilir, yani eşik ya aşılmıştır ya da aşılmamıştır, ardından işleme başlar. Eşik aşıldığında, sistem içinde tetikleme başlar. Bu, sigmoid fonksiyonu ile belirtilir (Bektaş, 2018).

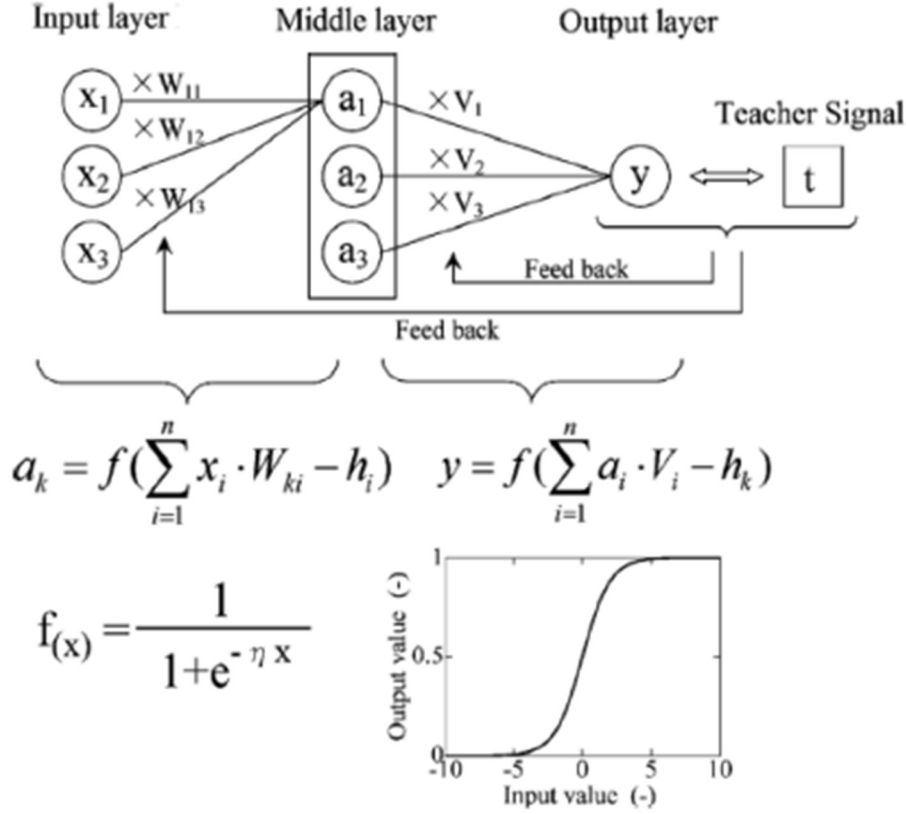


Şekil 4. Basitleştirilmiş şekliyle a) biyolojik ve b) yapay sinir ağı (Bektaş, 2018).

Eşik aşıldığında, sistem tetiklemeye başlar. Denklem (1), tetikleme mekanizmasını sigmoid fonksiyon eğrisiyle ifade eder:

$$y = f(x) = 1 / [1 + \exp(-\eta \cdot x)] \quad (1)$$

Burada, x giriş değerini, y ise çıkış değerini temsil eder. Denklemden, η (eta) sigmoid grafiğinin eğrisini belirleyen bir katsayıdır. Şekil 5, üç katmanlı bir yapay sinir ağı yapısında geri yayılım (backpropagation) ilkesini göstermektedir (Bektaş, 2018).



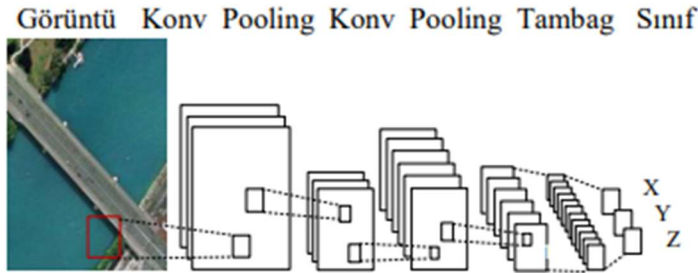
Şekil 5. Üç katmanlı yapay sinir ağı sisteminde geri yayılım diyagramı (Bektaş, 2018).

2.5. Derin Öğrenme

Hinton'ın çalışması, yapay sinir ağlarına yeni bir yaklaşım getirmiştir ve bu yaklaşımı yayımladığı makalesinde tanımlamıştır. Bu yaklaşım, derin öğrenme (Deep Convolutional Neural Network) olarak adlandırılmaktadır (Hinton et al., 2012). Konvolüsyonel sinir ağları, çok katmanlı sinir ağları olarak bilinir. Bu ağ sistemiyle yapılan önemli araştırmalar, yüksek performanslı sonuçlar elde edilmesini sağlamıştır. Derin konvolüsyonel sinir ağları, bu başarıları daha da ileriye taşımış ve önemli bir atılım gerçekleştirmiştir (Doğan ve Türkoğlu, 2018).

Konvolüsyonel sinir ağı, sinyal işleme, video analizi, görüntü analizi ve tespiti, sınıflandırma ve tıbbi görüntü işleme gibi çeşitli alanlarda önemli katkılarda bulunmuştur. Bu ağı kullanırken, ön işleme, özellik çıkarımı ve sınıflandırma-tespit gibi birkaç aşama yer alır. Her aşamada doğruluğu artırmak için özel yaklaşımlar uygulanır. Özellik çıkarımı süreci için birçok farklı yöntem önerilmiştir. Özellik çıkarımının amacı, tespit edilmesi gereken olaya ilişkin ayırt edici noktaları vurgulamaktır. Bir sonraki aşamada, belirlenen özellikler sınıflandırılmak için yapay sinir ağı kullanılarak işlenir (Kayalı ve İlhan Omurca, 2021).

Derin öğrenme ile daha önce ayrı ayrı yapılan birçok işlem artık birlikte gerçekleştirilir. Bu yaklaşımda, ön işleme ve özellik çıkarımı gibi aşamalar göz ardı edilir ve bu süreçler sinir ağı içinde otomatik olarak işlenir. Derin konvolüsyonel sinir ağlarında, özellik çıkarımı ağı içinde belirlenir ve tespit edilmesi gereken yapının özellikleri katmanlar içinde tanımlanır. Alt ve üst katmanlar arasında bağlı bir hiyerarşik yapı vardır. Özellik çıkarımı için ayrı bir aşama yoktur; nesneler veya olaylarla ilgili ayırt edici özellikler katmanlarda tanımlanır ve bir sonraki katmana aktarılır. Şekil 6, uydu görüntülerini sınıflandırmak için kullanılan bir konvolüsyonel sinir ağı modelini göstermektedir (Doğan ve Türkoğlu, 2018).



Şekil 6. Uydu görüntülerini sınıflandırmak için kullanılan konvolüsyonel sinir ağı modeli (Doğan ve Türkoğlu, 2018).

Yapay sinir ağı sınıflandırmasında kullanılan üç temel öğrenme yapısı vardır: denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme (Arı ve Berberler, 2017).

Denetimli öğrenme: Denetimli öğrenmede, giriş verileri $y(t)$ ve beklenen çıktı $d(t)$ sağlanır. Oluşturulan sinir ağında, istenen sonuca ulaşmak için ağırlıklar

tanımlanır. Bu ağırlıklar, verilen örneklere göre güncellenir ve giriş $y(t)$ 'nin çıktı $d(t)$ ile uyumlu hale gelmesi sağlanır. Ağırlıkların güncellenmesi süreci, öğrenme sürecinin gerçekleşmesini sağlamak için belirli bir iterasyon sayısı kadar devam eder (Alimovski, 2019).

Denetimsiz öğrenme: Denetimsiz öğrenmede, çıktı bilgisi sağlanmaz ve giriş görüntüleri ağda işlenir. Çıktı verileri ağ katmanlarında üretilir ve benzer değerlere sahip sonuçlar bir araya getirilir. Her grup bir sınıfı temsil eder (Doğan ve Türkoğlu, 2018).

Pekiştirmeli öğrenme: Pekiştirmeli öğrenmede, çıkışın ne olması gerektiğine dair bilgi verilmez. Ağdan, giriş verilerinden çıktı üretmesi beklenir. Bir öğretmen, sonucun doğru ya da yanlış olup olmadığını geri bildirir ve ağ, çıktı yanlışsa doğru sonucu üretmek için ağırlıklarını günceller (Arı ve Berberler, 2017).

Aynı yapay sinir ağı modeli, her problemi çözmek için kullanılamaz. Sinir ağları, nerede ve ne amaçla kullanıldığına göre farklılık gösterir. Seçilen model, problemin yapısına göre değişir. Bir sonraki bölümde, farklı yapay sinir ağı modelleri sunulmuştur. Sınıflandırma, tespit, teşhis, tahmin ve tanı gibi sonuçlar üretmek için farklı modeller seçilir. Verilerin yapısı (görüntü, ses, sinyal vb.) da sinir ağı modelinin seçiminde etkili olur.

2.5.1. Derin Sinir Ağlar

En popüler Derin Sinir Ağları (DNN), Yapay Sinir Ağları (ANN) temel alınarak oluşturulmuştur ve DNN'nin yapısı, normal bir sinir ağına kıyasla daha fazla gizli katmana sahip olacak şekilde ayarlanmıştır. Eklenmesi gereken gizli katman sayısı, ilgili probleme bağlıdır. Denetimli öğrenme ile eğitilen ve en az üç gizli katmana sahip olan Çok Katmanlı Algılayıcılar (MLP), DNN içinde en yaygın kullanılan sinir ağlarıdır (Alimovski, 2019).

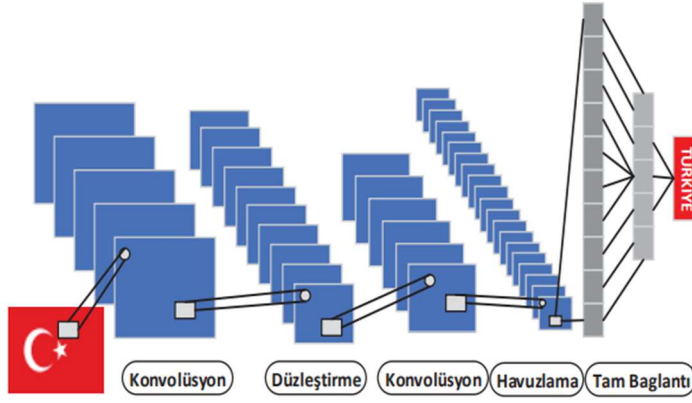
Bu yapılardaki gizli katmanların fazla sayıda olması, aynı zamanda ağırlıklar ve nöron sayısını da artırır. Bu karmaşık yapı ve düğümler nedeniyle, mevcut ağları eğitmek büyük miktarda veri ve çok yüksek hesaplama gücüne sahip bilgisayar

donanımı gerektirir. Bu, DNN'mizi sıfırdan eğitemememizin ana nedenidir; kendi ağımızı oluşturmak ve eğitmek için gerekli bilgisayar donanımına sahip değilizdir. ANN'ler ve geri yayılım yöntemi onlarca yıldır var olmasına rağmen (Gümüş ve Eyüpoğlu, 2023), 2000'lerden önce, internet ve sosyal medya kullanımında patlama yaşanmasından önce yeterli eğitim verisi almak zordu. Ayrıca, ANN'leri doğru bir şekilde eğitmek için yeterli güçlü bilgisayarlar yoktu. İyi haber 2006'da geldi (Tan et al., 2021). Hinton (2006), bir ANN'nin nasıl eğitilebileceğini her katmanı Sınırlı Boltzmann Makinesi olarak görerek göstermiştir. O yıl öncesinde ANN eğitimiyle ilgili çalışmalar olsa da, bu çalışma, ANN'lere ve uygulamalarına olan ilgiyi yeniden canlandırmıştır. Son yıllarda büyük veri ve Grafik İşlem Birimleri (GPU'lar) ile bu ağları eğitmenin daha ilginç yöntemlerinin geliştirilmesi, ANN'ler ve DNN'lerin insan seviyesinde performans gösterme yeteneğini artırarak araştırmacıların bu sistemlere olan ilgisini artırmıştır (Alimovski, 2019).

2.5.2. Konvolüsyonel (Evrişimsel) Sinir Ağlar

Konvolüsyonel Sinir Ağları (CNN'ler), esasen görüntüleri sınıflandırmak, bir fotoğraftaki benzerliklere göre nesneleri veya öğeleri kümelendirerek belirli özelliklere göre gruplamak için kullanılan çok katmanlı, ileri yönlü yapay sinir ağı modelleridir. Bu sistem ilk olarak 1968'de Hubel ve Wiesel tarafından hayvan görsel sistemlerinden ilham alınarak geliştirilmiştir (Hubel ve Wiesel, 1968). Yapay sinir ağı modelleri içinde en yaygın kullanılan modeldir.

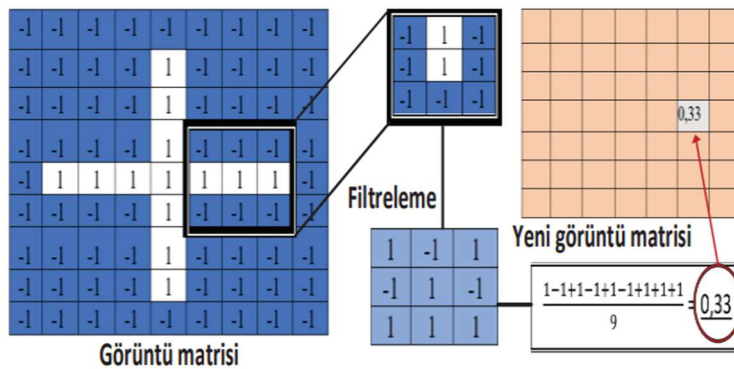
Konvolüsyonel sinir ağları, konvolüsyon, doğrusal aktivasyon birimi (ReLU), havuzlama ve tam bağlantılı katmanlardan oluşur. Bu işleme adımları, elde edilecek görüntünün netliğine göre farklı sayılarda ve boyutlarda tasarlanabilir. Bu adımların sıralaması Şekil 6'da gösterilmektedir (Akın ve Şahin, 2024).



Şekil 6. Konvolüsyonel sinir ağı katmanları (Hu, 2015; Akın ve Şahin, 2024)

Konvolüsyonel sinir ağı, görüntülerdeki farklı özelliklerin tespit edilmesini ve gruplanmasını sağlayan birçok filtreleme ve özellik çıkarımı işlemi gerçekleştirir. Görüntülerdeki nesnelerin (insan, araçlar, hayvanlar, bitkiler vb.) özelliklerini kolayca tespit etmek mümkündür (Akın ve Şahin, 2024).

Konvolüsyonel sinir ağlarında ilk adım konvolüsyonel katmandır. Şekil 6'da görüldüğü gibi, bu adımda, bir görüntünün 9x9x3 boyutlarında olan matrisine, her satır-sütun aralığına uygulanan 3x3 boyutlarında bir filtre ile işlem yapılır ve sonuçta 7x7x3 boyutlarında yeni bir görüntü matrisi elde edilir. Filtreleme matrisi ve ana matriste taranan alan, satır satır ve sütun sütun çarpılır. Ortaya çıkan matris toplanılır ve tüm sütun ve satır değerlerinin ortalaması alınır. Elde edilen değerler, Şekil 7'de gösterilen tabloda işlenir. Bu değerler, veri boyutuna, sayısına ve kullanılan filtreleme işlemine bağlı olarak değişebilir (Altan, 2019).

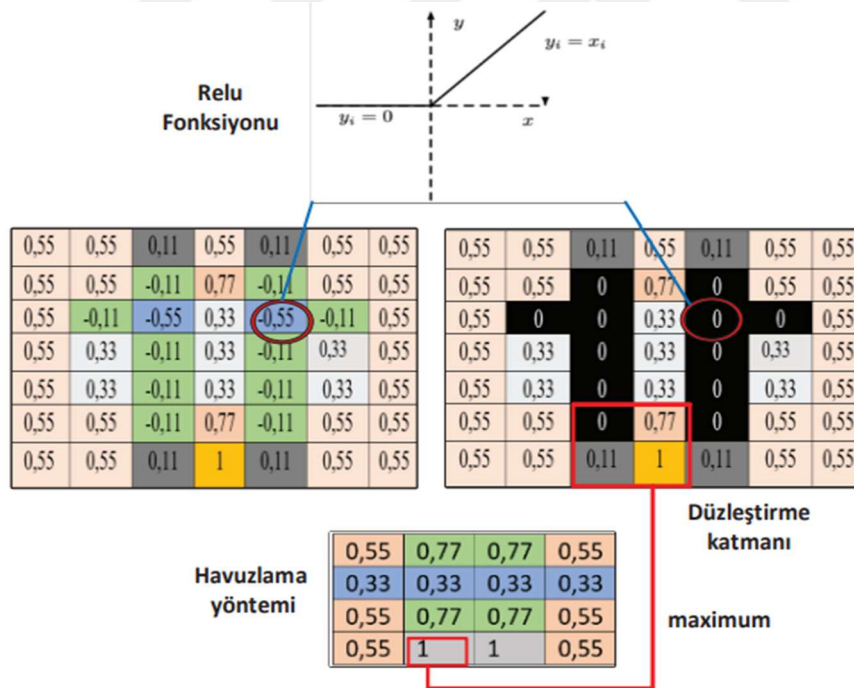


Şekil 7. Konvolüsyonel katmanı işlem basamakları (Akın ve Şahin, 2024)

İkinci adım olan düzleştirme işlemi, genellikle Şekil 6'da görüldüğü gibi ReLU (Düzleştirilmiş Doğrusal Birim) fonksiyonu kullanır. Bu işlemin amacı, sıfırdan küçük olan değerleri karanlık alanlar olarak yorumlamak ve onları sıfıra ayarlamaktır.

$$Relu = f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (2)$$

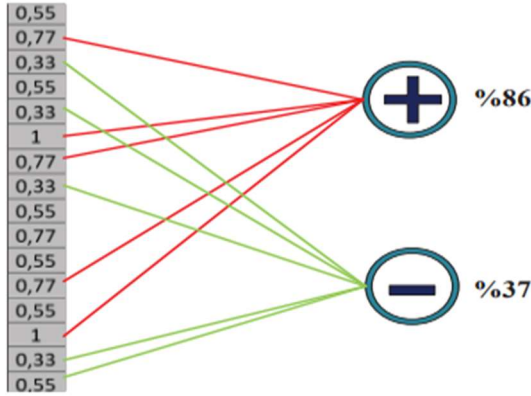
Pooling katmanının amacı, konvolüsyon işlemiyle elde edilen yeni görüntü matrisinin boyutunu azaltmaktır. Boyutun küçültülmesi, ağır işleme hızını artırır. İki tür pooling yöntemi vardır: ortalama pooling ve max pooling. En yaygın kullanılan yöntem max pooling'dir, burada NxN matrisleri seçilir ve içindeki en büyük değer seçilir. Genellikle 2x2 matris kullanılır. Şekil 8'de max pooling yöntemi gösterilmektedir. Bu yöntemin amacı, ReLU işlemi sonrasında oluşturulan matrisin karanlık noktalarını ortadan kaldırmaktır. Elde edilen görüntü matrisi istenilen netliği sağlamıyorsa, istenilen pooling yöntemi tekrar uygulanabilir (Akın ve Şahin, 2024).



Şekil 8. Relu fonksiyonu kullanarak düzleştirme katmanı ve Maksimum havuzlama yöntemi (Akın ve Şahin, 2024)

Konvolüsyonlar, düzleştirme ve pooling sayısı, kullanılan görüntüye veya kullanıcının amacına bağlı olarak değişebilir. Bu işlemlerin amacı, tamamen bağlı

katmandan önce tek boyutlu bir giriş verisi elde etmektir. Son adımda, tamamen bağlı katman, nesneyi tanımlayacak olan özellikleri ve bunların ilgili kategorilerini belirler. Şekil 9'da görüldüğü gibi, tamamen bağlı katmanda, doğruluk oranını elde etmek için sütundaki en yüksek değerlerin ortalaması alınır, hata oranını elde etmek için ise en düşük değerlerin ortalaması alınır (Altan, 2019).



Şekil 9. Tam bağlantı katmanı (Altan, 2019).

Konvolüsyonel sinir ağı modelleri günümüzde öncelikle görüntü işleme alanında kullanılmaktadır. Ayrıca, ses ve video işleme, hastalık tanısı için tıbbi görüntüleme, otonom cihazların kontrolü ve görüntülerde nesne tanımlama gibi alanlarda da tercih edilmektedir.

2.7. Literatür Araştırması

Literatürdeki çalışmalar incelendiğinde, yüz tespiti ve sıcaklık ölçümü ile ilgili çeşitli yaklaşımlar sunulmaktadır. Ancak çalışmaların ortak noktası çoğu zaman yüz tanıma ve sıcaklığın topografik gösterimi konusunda yaşanan sorunlardır. Aşağıda yüz tanıma ve sıcaklık tespiti ile ilgili literatür taramalarına yer verilmiştir.

2.7.1. Yüz Tanıma Konusunda Yapılan Araştırmalar

Zhang vd. (2014), yüzü tespiti konusunda etkili sonuçlar elde etmek için çoklu görev öğrenimi ile geliştirilmesi gerektiğini savunmuştur. Özellikle, yüz tespiti, heterojen ancak ince bir şekilde ilişkili görevlerle optimize edilmiştir. Uygulama sonucunda, önerilen görev kısıtlı öğrenimin mevcut yöntemlere kıyasla, özellikle ciddi

engellenme ve poz deęişikliklerine sahip yüzlerle başa çıkmada daha başarılı olduęu ve model karmaşıklığını büyük ölçüde azalttığı gözlemlenmiştir.

Li H. vd. (2015), gerçek yüzleri tanımda mekan ve açı sorununa değinerek (örneğin aydınlatma) ayırarak arka plandan doğru şekilde ayırt edebilen bir CNN kademeli modelini ortaya atmışlardır. Çalışmada gerçekleşen CPU’da tipik VGA görüntüleri için 14 FPS’ye, GPU’da ise 100 FPS’ye kadar hızlandırılabilen sensörler (dedektörler), giriş görüntüsünü düşük çözünürlükte değerlendirerek yüz olmayan alanları hızlıca reddeder ve daha zor alanları yüksek çözünürlükte dikkatlice işler. Çalışmada algılama hızını artırmak ve sınırlayıcı kutu kalitesini iyileştirmek için aşamalı olarak kalibrasyon ağırları eklenmiştir.

Liu vd. (2015), yüz tanıma tahminleri için yeni bir derin öğrenme çerçevesi kullanılmıştır. Ağ, iki farklı kademeli CNN modelinden oluşmaktadır. LNET ve ANET katmanları farklı şekilde önceden eğitilmiştir. LNET yüz tanıma tahmini için eğitilirken, ANET büyük yüz kimlikleriyle özellik tahmini için eğitilmiştir. Tasarlanmış ön eğitim stratejileri sayesinde model, arka plan karışıklıklarına karşı direnç göstermiş yüze odaklanmıştır. Gereksiz hesaplamayı önlemek için hızlı bir ileri besleme algoritması kullanılmıştır.

Yang vd. (2015), mevcut veri kümelerinden 10 kat daha büyük olan WIDER FACE veri kümesini sunmuştur. Veri kümesi, maskelemeler, pozlar, etkinlik kategorileri ve yüz sınırlayıcı kutuları gibi zengin açıklamalar içerir. Önerilen veri kümesindeki yüzler, ölçek, pozlama ve örtüşmedeki büyük farklılıklar nedeniyle oldukça zordur. Bu faktörler, birçok gerçek yüz tanıma sistemlerinde oldukça yaygındır. Bu nedenle, bu veri kümesiyle eğitilmiş modeller oldukça iyi performans göstermektedir.

Ranjan vd. (2017), derin evrişimli sinir ağılarını (CNN) kullanarak yüz algılama, işaretlerin yerleştirilmesi, poz tahmini ve cinsiyet tanıma işlemlerini eşzamanlı gerçekleştiren bir algoritma geliştirmiştir. HyperFace olarak adlandırılan yöntem, derin bir CNN’in katmanlarını ayrı bir CNN ile birleştirir ve ardından birden fazla görevi bir

arada öğrenen bir algoritma uygular. Kapsamlı deneyler, önerilen modellerin hem genel hem de yerel bilgileri yakalayabildiğini ve bu dört görevin her biri için birçok rakip algoritmaya üstünlük sağladığını göstermiştir.

Jiang ve Learned-Miller, Faster RCNN'i yüz algılama için uygulamayı önermiştir ve bu yöntem, çeşitli nesne algılama kriterlerinde etkileyici sonuçlar göstermiştir. Yazarlar, büyük ölçekli WIDER yüz veri kümesi üzerinde daha hızlı bir R- CNN modelini eğitmiş ve bunu WIDER test setindeki en son sonuçlarla karşılaştırmıştır. Çalışma sonucunda, Faster R-CNN'in genel nesne algılama için tasarlanmış olmasına rağmen, uygun bir yüz algılama eğitim setiyle yeniden eğitildiğinde etkileyici yüz algılama performansı gösterdiği sonucuna varılmıştır (Jiang & Learned-Miller, 2017).

Mundial ve arkadaşları (2020) tarafından yürütülen çalışmada, COVID-19 pandemisi sırasında yüz tanıma problemleri ele alınmıştır. Araştırmacılar, maske takan bireylerin doğru şekilde tanımlanmasında yüz tanıma teknolojilerinin karşılaştığı zorluklar hakkında bilgi sunmuştur. Makalede, maske kullanımının yüz tanıma sistemlerine etkisi incelenmiş ve maske takan bireylerin tanınması için kullanılan farklı yaklaşımlar, yöntemler ve teknikler tartışılmıştır. Ayrıca, pandemi sürecinde yüz tanıma teknolojilerinin artan kullanımıyla ortaya çıkan güvenlik, mahremiyet ve etik sorunlara da değinilmiştir. Makale, bu dönemde karşılaşılan mevcut zorluklar ve yüz tanıma teknolojilerinin gelecekteki potansiyeline dair önemli bilgiler sunmaktadır.

Karadağ (2020) tarafından yapılan çalışmada, yüz tanıma işlemi, Haar Cascades sınıflandırıcısıyla yüz tespiti yapıldıktan sonra LBPH (Yerel İkili Desen Histogramları) yöntemi kullanılarak gerçekleştirilmiştir. Van Natta ve arkadaşları (2020) ise COVID-19 pandemisi sırasında termal yüz tanıma teknolojisinin artışı ve düzenlenmesini incelemiştir. Araştırmada, bu teknolojinin güvenlik ve sağlık sektörlerindeki uygulamaları, etik ve mahremiyet endişeleri ile düzenleyici önlemler ele alınmıştır. Ayrıca, yüz tanıma teknolojisinin potansiyel faydaları ve riskleri üzerine de değerlendirmeler yapılmıştır.

Vansh ve arkadaşları (2020), "YCbCr" renk alanı ve "AdaBoost" algoritmaları kullanılarak geliştirilen bir yüz tanıma yöntemini tartışmıştır. Çalışmada, YCbCr renk alanının yüz tespit performansını artırdığı, AdaBoost algoritmasının ise sınıflandırma doğruluğunu iyileştirdiği belirtilmiştir. Bu çalışma, yüz tanıma sistemlerinin performansını artırmak ve daha doğru ve verimli sonuçlar elde etmek için yeni bir yaklaşım sunmaktadır.

Özdemir ve Koç (2019) tarafından yapılan çalışmada, internetten erişilebilen görüntüler kullanılarak oluşturulan bir veri seti (RidNet) üzerinden yedi farklı yüz ifadesinin tanınması için derin öğrenme yöntemleri kullanılmıştır. Daha sonra, RidNet üzerinde AlexNet, GoogLeNet ve ResNet101 gibi literatürde bilinen evrimsel sinir ağı (CNN) mimarileri ile transfer öğrenme gerçekleştirilmiştir. Çalışmada, Compound Facial Expressions of Emotion (CE) ve Static Facial Expressions in the Wild (SFEW) veri setleri test veri seti olarak belirlenmiştir. En iyi sınıflandırma performansını gösteren CNN mimarisi, deneysel çalışmalarla tespit edilmiştir. AffectNet, The Karolinska Directed Emotional Faces (KDEF) ve RidNet veri setleriyle eğitilen bu ağlar, kontrollü ortamda oluşturulan CE veri seti ile test edildiğinde benzer performanslar sergilemiştir. Ancak kontrolsüz ortamda bulunan SFEW veri setinde, RidNet ile eğitilen ağ diğer ağlara kıyasla önemli bir avantaj sağlamıştır.

Kekül ve arkadaşları (2018), yüz tanıma uygulamalarında Eigenfaces ve Yapay Sinir Ağlarını karşılaştırmıştır. Çalışmada, her iki yöntemin yüz tanıma performansındaki etkili yönleri ve avantajları vurgulanmıştır. Eigenfaces yöntemi hızlı çalışması ve düşük hesaplama gereksinimleriyle öne çıkarken, yapay sinir ağları daha karmaşık özellikleri algılama ve daha yüksek doğruluk sağlama yeteneğine sahiptir. Makale, bu yöntemlerin uygulamaları, avantajları ve dezavantajları açısından detaylı bir karşılaştırma sunmakta ve yüz tanıma sistemlerinde karar verme süreci için yol gösterici bir çerçeve sunmaktadır. Çalışma, farklı yöntemlerin bir arada kullanılmasıyla daha iyi sonuçlar elde edilebileceğini önermektedir.

Mamak ve arkadaşları (2020), gerçek zamanlı yüz tanıma tabanlı personel kontrol ve izleme sistemi üzerine bir araştırma yapmıştır. Çalışma, yüz tanıma teknolojisinin personel kontrolü ve izleme süreçlerinde nasıl kullanılabileceğini açıklarken, tasarlanan

sistemin işleyişine dair bilgiler sunmaktadır. Bu sistem, personel kayıtları, yoklama takibi ve erişim kontrolü gibi uygulamalarda gerçek zamanlı yüz tanıma için kullanılmıştır. Makale ayrıca, sistem tasarımında kullanılan teknikler ve algoritmalar hakkında bilgi vermektedir.

Atasoy ve Tabak (2018), Destek Vektör Makineleri (SVM) kullanarak bir yüz tanıma uygulaması geliştirmiştir. Çalışmada, SVM algoritmasının yüz tanıma alanında nasıl uygulanabileceği, çalışma prensipleri ve uygulama aşamaları ayrıntılı şekilde ele alınmıştır. Veri toplama, özellik çıkarımı, model eğitimi ve sınıflandırma adımları vurgulanmış ve farklı veri setleri üzerinde yapılan deneylerin sonuçları paylaşılmıştır. SVM'nin performansı, doğruluk oranı ve avantajları da makalede detaylı şekilde değerlendirilmiştir.

Terzopoulos ve Waters (1990), fiziksel ve anatomik modeller kullanarak yüz görüntülerinin analizini gerçekleştirmiştir. Çalışmada, yüz tanıma sistemlerinin daha doğru ve güvenilir sonuçlar elde edebilmek için fiziksel ve anatomik modellere dayanabileceği belirtilmiştir. Fiziksel modellerin yüzün yapısını ve hareketlerini matematiksel denklemlerle temsil ettiği, anatomik modellerin ise yüzün anatomik yapısını temsil eden veri tabanları ve özellik tabanlı analiz yöntemlerini içerdiği ifade edilmiştir. Bu modellerin yüz tanıma performansını artırmak için nasıl kullanılabileceği ve analiz sürecine etkileri araştırılmıştır.

Kirby ve Sirovich (1990), insan yüzlerini tanımlamak için Karhunen-Loeve prosedürünü kullanan bir yöntem geliştirmiştir. Rodriguez ve Marcel (2006), yerel ikili desen histogramları (LBPH) kullanarak yüz doğrulama üzerine bir çalışma gerçekleştirmiştir. Ngo ve arkadaşları (2009), gerçek zamanlı bir yüz algılama sistemi için modüler bir yaklaşım kullanarak FPGA tabanlı bir tasarım geliştirmiştir.

2.7.3. Isı Tespiti Konusunda Yapılan Çalışmalar

Sun vd., (20117) bulaşıcı hastalıkları olan hastaları taramak için IRTequipped CMOS kamera kullanan birleşik görünür ve termal görüntü işleme yaklaşımının uygulanmasını önermiştir. Görüntü, görünür ve termal görüntüler üreten bir IRT sistemi

kullanılarak elde edilmiştir. Deneklerin solunum hızları, termal görüntülerdeki burun çevresindeki sıcaklık değişiklikleri izlenerek ölçülmüştür; aynı anda yüz derisi sıcaklıkları da ölçülmüştür. Sonuç olarak, önerilen sistem, bulaşıcı hastalık şüphesi olan hastaları verimli bir şekilde tespit etmiştir (Sun vd., 2017).

Li vd. (2018), kızılötesi termografi kullanarak gerçek zamanlı termal algılaması yorumlaması için yeni bir çerçeve sunmuştur. Bu çalışmanın temel katkısı, yüz derisi sıcaklığı verilerini sürekli ve otomatik olarak toplamak, geri çağırmak ve analiz etmek ve her kullanıcısı için termal belirleme koşullarını gerçek operasyonel ortamlarda yorumlamak üzere önerilen veri toplama ve analiz çerçevesidir. Önerilen çerçeve, termoregülasyon teorisi, bilgisayarla görme ve makine öğrenimi dahil olmak üzere disiplinler arası teknikleri kullanmıştır. Sonuçlar, düşük maliyetli ve müdahale etmeyen kızılötesi termal kameralardan toplanan yüz derisi sıcaklıklarının, termal konforun gerçek zamanlı olarak sağlam bir tahminini sağlamaya yardımcı olabileceğini ve zemin üzerindeki minimum kesintilerle iç ortamların senkronize kontrolüne olanak tanıdığını göstermektedir.

Kumar vd. (2020), enfekte kişilerin kolayca tespit edilmesini, insan teması olmadan gıda ve hizmetlerin ulaştırılmasını, termal kameralarla ateş taramasını, halka açık alanların dezenfekte edilmesi için dron kullanımını ve daha fazlasını içeren bir model önermiştir. Çalışma, modern teknolojinin COVID-19'u kontrol altına alma rolüne odaklanmaktadır. Yapay zekâ kullanılarak radyoloji görüntüleriyle erken teşhis sağlanması, anormal solunum desenlerinin belirlenmesi, hastanın sağlık durumunun tahmin edilmesi gibi birçok teknolojik çözüm üzerinde durulmaktadır. Ayrıca, sosyal medya ve farkındalık kampanyalarının maske takma ve el yıkama gibi önemli bilgileri halka ulaştırmada oynadığı rol vurgulanmıştır.

Chan vd.'ın (2020) çalışmalarında pandemi döneminde hızlı ateş taraması, bir nesneden yayılan kızılötesi enerjiyi tespit edip bunu sıcaklığa dönüştüren kızılötesi termografi kullanılarak yapılmıştır. 20 günlük bir süre boyunca toplamda 176 kişi taranmıştır. Yüzün farklı bölgeleri 0,5 ila 1,5 metre mesafeden taranarak alınan IRT (kızılötesi termografi) ölçümleri, geleneksel cıvalı termometrelerle alınan vücut sıcaklıklarıyla karşılaştırılmıştır. Bu veriler korelasyon ve regresyon analizine tabi

tutulmuştur. Sonuç olarak, IRT ölçümlerinin en doğru sonucu yüzün yan taraflarından verdiği ve vücut sıcaklığını tutarlı bir şekilde belirlediği gözlemlenmiştir. Bu sistem, havaalanları ve sınır geçişlerinde yolcuların yüksek vücut sıcaklığını kontrol etmek için uygulanabilir.

Maghdid, vd (2020), çalışmalarında yalnızca akıllı telefon sensörlerini kullanarak COVID-19'u tespit etmek için yeni bir çerçeve önermiştir. Çözümün işleyişi, öncelikle ateş, yorgunluk ve kuru öksürük gibi semptomların tespit edilmesine dayanmaktadır. Çerçeve, normal ateş ile COVID-19'a bağlı ateş düzeyini yerleşik sensörler aracılığıyla ayırt etmeye çalışmaktadır. Bir dizi sensör, koronavirüs hastalığında mevcut semptomları tespit etmek için ayarlanmıştır. Ayrıca, insan sağlığını kontrol etmek için bu okumalar üzerine algoritmalar uygulanmıştır. Makalede dört farklı katman tartışılmaktadır: sensör verilerini okuma için bir giriş katmanı, sensör yapılandırma katmanı, semptomları analiz ederek hastalığı belirleme katmanı ve son olarak hastalığı tahmin etme katmanı. Farklı sensörler sıcaklık, yorgunluk ve yürüyüş desenlerini algılayarak hastalığın tespit edilmesine olanak tanır. Bu çözümler, herkesin kullandığı akıllı telefon ve sensörlerle maliyet açısından oldukça etkili olduğunu kanıtlamaktadır.

Toplu ateş taraması için anlamlı ve doğru bir vücut sıcaklığı ölçümü almak her zaman önemli bir konu olmuştur. Bu amaçla, insan vücudunun güvenilir bir bölgesinden sıcaklık ölçümü yapılması gerekmektedir. Ayrıca, deri sıcaklığının yalnızca çekirdek vücut sıcaklığına bağlı olmadığını, çevre gibi birçok diğer faktör tarafından etkilendiğini unutmamak gerekir. Kamera kullanımı sırasında yanlış-negatif oranının artmaması önemlidir; çünkü yanlış-pozitif oranının artması, yanlış bir güvenlik hissi yaratabilir. Toplu ateş taramaları sırasında termal kameraların verimliliği ve doğruluğu değerlendirilmiştir. Eddie vd. (2004) yaptıkları çalışmada regresyon ve ROC eğrisi ile biyostatistik yöntemine başvurarak termal kameraların toplu ateş taraması için doğru ve temel bir araç olduğunu açıkça ortaya koymuşlardır.

3. MATERYAL VE YÖNTEM

Bu çalışma, termal görüntüler kullanılarak COVID-19 ve benzeri ateşli hastalıkların tespitini hedefleyen, Haar Cascade, Derin Öğrenme ve Büyük Dil Modelleri (LLM) tekniklerinin performanslarının karşılaştırılmasına dayanmaktadır. Araştırmada hem görüntüleme hem de öğrenme algoritmalarının etkinliği çok yönlü olarak değerlendirilmiştir.

Çalışma için toplamda 20 adet termal görüntüden oluşan bir veri seti kullanılmıştır. Bu görüntüler, farklı açılardan ve çoklu kişi içeren senaryolardan elde edilerek çeşitlilik sağlanmıştır. Veri setinde, şu tür çekimler bulunmaktadır:

- **Normal:** Karşıdan çekilmiş bireysel görüntüler.
- **Farklı Açı:** Çeşitli açılardan çekilen bireysel görüntüler.
- **Çoklu:** Birden fazla bireyi içeren karmaşık sahneler.

Veri seti, Haar Cascade, Derin Öğrenme ve LLM algoritmalarının gereksinimlerine uygun olarak ön işleme sürecinden geçirilmiştir. Çalışmada görüntü segmentasyonu ile yüz bölgesinin ayrılması ve termal verinin optimize edilmesi sağlanmıştır. İşlem hızını artırmak için tüm görüntüler 240x320 boyutuna indirgenmiştir. Görüntüler, ait oldukları sınıf kategorilerine (Normal, Farklı Açı, Çoklu) etiketlenmiştir.

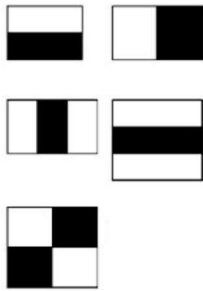
Çalışmamızda OpenCV kütüphanesi kullanılarak standart Haar Cascade algoritması uygulanmış ve yüz algılama performansı ölçülmüştür. Görüntüler üzerindeki algılama hızı ve kesinlik oranları hesaplanmıştır. Convolutional Neural Network (CNN) tabanlı bir model oluşturulmuş ve eğitilmiştir. Model, PyTorch kütüphanesi kullanılarak Öğrenme oranı: 0.001, Epoch sayısı: 50 ve Batch size: 16 olacak şekilde eğitilmiştir. Performans değerlendirmesi, karmaşıklık matrisi ve ROC eğrisi üzerinden yapılmıştır. OpenAI'nin GPT tabanlı modelleri kullanılmış, termal görüntülerden elde edilen verilerin metin olarak şifreleştirilmesi ve sınıflandırılması sağlanmıştır. Model, görüntüleri kategorize ederken termal verilerin ön işlenmiş

çıktılarına dayanmıştır. Tüm algoritmalar için performans, Doğruluk (Accuracy), Kesinlik (Precision), Duyarlılık (Recall), F1-Skoru ve Alan Altındaki Eğri (AUC) değeri ile değerlendirilmiştir. Sonuçlar, modellerin farklı termal görüntü çeşitlerini tespit edebilme kapasitesini ve ateşli hastalıkların özelliklerini doğru bir şekilde çıkarabilme yeteneğini karşılaştırmayı amaçlamaktadır.

3.1. Haar Cascade Algoritması

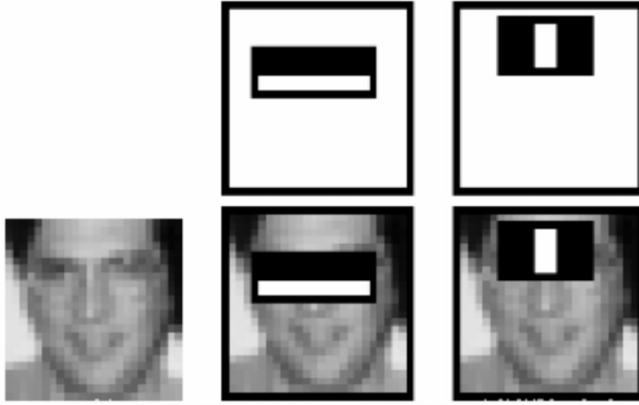
Haar Cascade algoritması, yüz okuma süreçleri için önceden eğitilmiş bir model üzerinde kullanılmıştır. Haar özelliklerini kullanan nesne algılama yaklaşımı, Haar dalgacık dönüşümünden uyarlanarak geliştirilmiştir. Nesneler, alt bileşenlerinin renk dağılımı ve yoğunluk bilgisi temel alınarak farklı alt parçalara ayrılır. Bu parçaların özellikleri, tüm nesneyi tanımlamak için farklı özellik setleriyle ifade edilir. Girdi görüntüsündeki yüz bölgelerini belirleme süreci, tüm görüntünün taranmasıyla gerçekleştirilir. Sınıflandırıcı, farklı boyutlardaki yüz özelliklerini algılayabilir. Girdi görüntüsünde farklı ölçeklerdeki yüz özelliklerini bulmak için aynı algılama işlemi, farklı ölçek parametreleriyle tekrarlanır.

Şekil 1, Haar Cascade algoritmasının üç temel özelliğini göstermektedir. Eğer bir görüntünün belirli bir alanı koyu renklerden, başka bir alanı ise açık renklerden oluşuyorsa, bu bir kenar özelliği olarak kabul edilir. Görüntü açık, koyu ve tekrar açık renkler arasında geçiş yapıyorsa, bu bir çizgi özelliğini belirtir. Kare bir alan içinde koyu ve açık tonlar çapraz bir düzenlemeyle yer alıyorsa, bu dört kare özelliğini ifade eder. Bu özellikler kullanılarak kenar, göz ve yüz gibi nesneler sağlanan görüntüde algılanabilir.



Şekil 10. Haar Cascade Özellikleri: Kenar (üstte), Çizgi (ortada), Dört kare (altta) (URL-1).

Şekil 11’de bir yüz okuma görüntüsü örneği verilmiştir. Haar Cascade yöntemi, insan yüzleri için kapsamlı bir şekilde eğitilmiş olup, yüz özelliklerinin yapısını doğru bir şekilde anlamaktadır. Yüz algılama sistemi, ilk olarak gözleri arar. Gözlerin girdi görüntüsünde mevcut olması durumunda, burun tespit edilir. Burun algılandıktan sonra kaşlar kontrol edilir ve böylece istenilen sonuçları verebilecek bir yapı sağlanır.



Şekil 11. Haar Cascade ile Yüz Okuma Örneği (URL-1).

Haar Cascade kullanılarak sağlanan görüntüdeki nesne tanımlama işlemleri kolayca gerçekleştirilebilir. Bu işlem için örnek bir Python kodu şu şekildedir:

Tablo 4. Python kodu (URL-1).

```
import cv2
face_cascade =
cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
cap = cv2.VideoCapture(0)
while 1:
    ret, img = cap.read()
    gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray,
1.3, 5)
    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h),
(255, 0, 0), 2)
        roi_gray = gray[y:y + h, x:x + w]
        roi_color = img[y:y + h, x:x + w]
        #show the frame window
        cv2.imshow('img', img)
        k = cv2.waitKey(30) & 0xff
    cap.release()
cv2.destroyAllWindows()
```

Bu çalışmada yüz tespiti için hibrit bir yöntem kullanılmıştır; RGB görüntülerde yüz tespiti yapıldıktan sonra, tespit edilen yüzlerin koordinatları IR (termal) görüntülerle eşleştirilmiştir. Bunun nedeni, termal görüntülerde yüz tespit performansının RGB'ye kıyasla düşük olmasıdır. RGB görüntülerdeki yüz bölgelerinin belirgin olması, tespit doğruluğunu artırmış ve IR görüntülerde bu bölgelerin sıcaklık analizine uygun şekilde işaretlenmesini sağlamıştır.

3.4. Derin Öğrenme Uygulamasında Kullanılan CNN ve Programlama Dili

Derin öğrenme, yapay zeka alanında karmaşık veri yapılarını modellemek için kullanılan bir tekniktir. Bu yöntemin etkili bir şekilde uygulanabilmesi için güçlü programlama dillerine ve özel olarak tasarlanmış kütüphanelere ihtiyaç duyulur. Bir CNN modeli olarak en yaygın kullanılan programlama dili Python'dur. Python, geniş kütüphane desteği, kullanım kolaylığı ve topluluk desteği nedeniyle derin öğrenmede en çok tercih edilen programlama dilidir. TensorFlow, PyTorch ve Keras gibi popüler çerçeveler Python ile uyumludur.

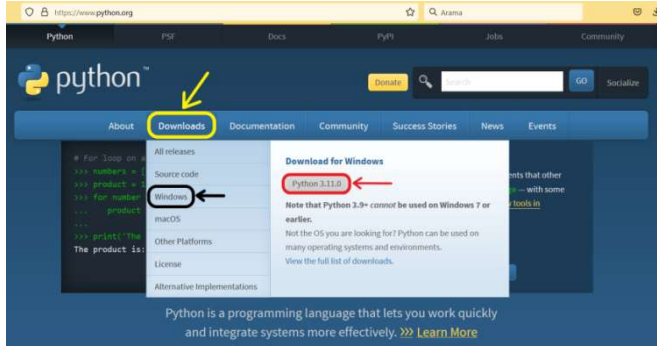
3.4.1. Python

3.4.1.1. Python Yükleme

Python genel olarak Anaconda¹ paketi ile beraber yüklenen bir programdır. Program direk olarak resmî web sitesinden² indirilmiş ve manuel olarak kurulmuştur. Şekil 12'de gösterildiği gibi siteden sarı renkli ok ile gösterilen “Downloads” sekmesinin üzerine gelince seçenekler görüntülenmektedir. Sayfanın ziyaret edildiği tarihe bağlı olarak kırmızı renkli ok ile gösterilen en güncel Python sürümü olan “Python 3.11.0” değişkenlik gösterecektir.

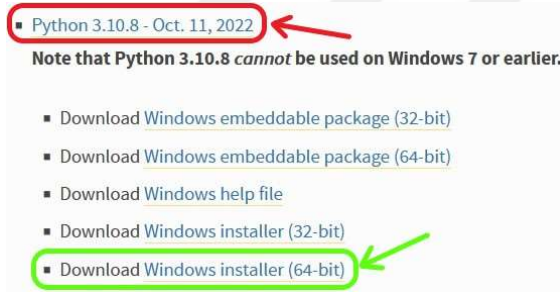
¹ <https://www.anaconda.com/>

² <https://python.org>



Şekil 12. Python resmî web sitesi (Downloads sekmesi) (URL-2)

Burada siyah renkli ok ile gösterilen “Windows” sekmesine tıklandığında gelen sayfada Windows için Python sürümleri yer almaktadır.³ Sol taraftaki kararlı sürümlerden⁴ Şekil 13’te (kırmızı renkli ok ile) gösterilen Python 3.10.8’in Windows 64-bit için (yeşil renkli ok ile) gösterilen yükleyicisine tıklayarak **python-3.10.8-amd64.exe** dosyası indirilir.



Şekil 13. Python’ın uygun bir sürümünün indirilmesi (URL-2)

Yukarıda Şekil 4’de dikkat edilmesi gereken önemli bir nokta kırmızı renkle gösterilen Python sürümünün altında koyu renkle yazılan notta belirtilen Windows versiyon uyumluluğu. Şekil 5’deki yazınlara göre, eğer bilgisayarda işletim sistemi olarak Windows 7 veya daha eski sürümler yüklüyse (e.g., Windows XP) o zaman mecburen Python 3.10.8’den daha eski bir sürüm yüklemek gerekecektir. Şekil 4’ün görüntülediği sayfada aşağı doğru bakıldığında eski sürümler yer almaktadır.

³ Mac veya Linux kullanıyorsa doğal olarak ilgili sekmeler seçilmeli ancak bu platformlarda Python gibi yazılımlar genellikle konsoldan yüklenmektedir.

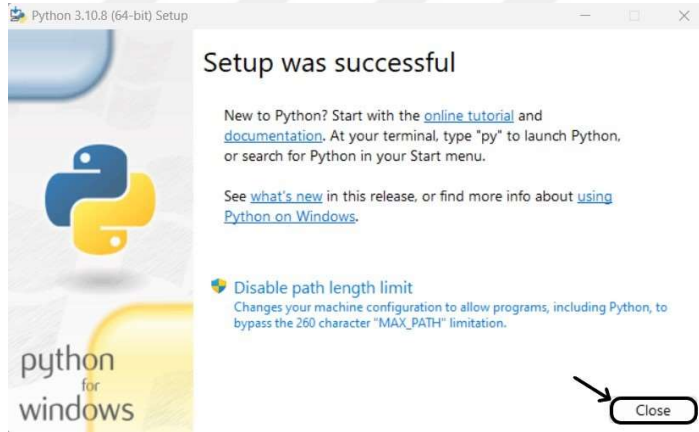
⁴ İng. stable releases

İndirilen **python-3.10.8-amd64.exe** dosyası yüklenir. Şekil 14’de gösterildiği gibi gelen ekranda özellikle en alt tarafta okla gösterilen **python.exe**’yi **PATH** denilen yere ekleme seçeneği çok önemli. Bu seçenek seçilmezse yükledikten sonra Python direk olarak çalışmayacağından dolayı mutlaka “Add python.exe to PATH” seçeneğini işaretlenmelidir⁵. Aşağıda okla gösterilen **Install Now** sekmesine tıkladığınızda ekrana gelecek mesaja onay verip Python’ı yüklenir.



Şekil 14. Python yükleme ekranı

Python yüklenmesi bittikten sonra Şekil 15’de okla gösterilen **Close** sekmesine tıklayıp kurulum tamamlanır.



Şekil 15. Python yüklendikten sonra kurulumu sonlandırma

Bilgisayarımıza Python yüklendiğini Denetim Masası → Program Ekle Kaldır’a gidip kontrol edilebilir ama bir sonraki bölümde tanıtılacak olan Windows PowerShell komut satırını kullanarak bu işlemi gerçekleştirilir.

⁵ Add **python.exe** to **PATH** seçeneği seçilmezse bilgisayardaki **ortam değişkenlerine** manuel olarak ekleme yapılması gerekir

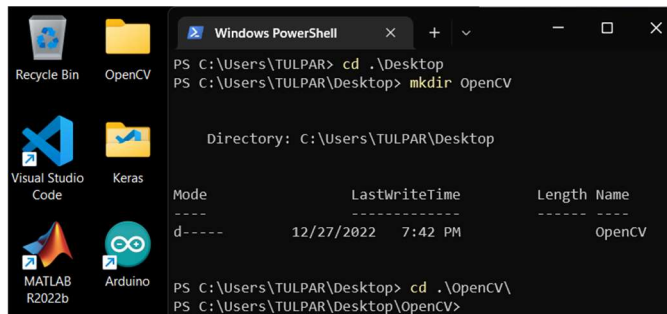
3.4.1.1.2. PowerShell

Komut İstemi'nin (Command Prompt) pek çok geliştirici tarafından yetersiz bulunması nedeniyle bu çalışmada, OpenCV yükleme işlemleri için PowerShell (PS) tercih edilmiştir. Ancak, belirtilen komutlar Komut İstemi kullanılarak da uygulanabilir. PowerShell'in tercih edilmesinin temel nedeni, Linux sistemlerinde kullanılan birçok komutun (örneğin, **cd**, **mkdir**) PowerShell üzerinde de çalışabilir olmasıdır. Şekil 16'da PowerShell logosu gösterilmektedir.



Şekil 16. PowerShell logosu (URL-3)

Bu çalışmada kullanılan bilgisayarda kullanıcı adı TULPAR olduğundan, PowerShell başlatıldığında varsayılan çalışma dizini C:\Users\TULPAR olarak görüntülenmektedir. OpenCV adlı bir klasör oluşturmak için iki yöntem kullanılabilir: grafik kullanıcı arayüzü (GUI) üzerinden fare ve klavye yardımıyla işlemi manuel olarak gerçekleştirmek ya da PowerShell komut satırında ilgili komutları kullanmak. Şekil 17'de görüldüğü üzere, öncelikle PowerShell üzerinde cd komutu ile Masaüstü dizinine geçiş yapılır, ardından mkdir OpenCV komutu kullanılarak yeni bir klasör oluşturulur. Bu işlem sonrasında cd OpenCV komutuyla oluşturulan klasörün içerisine geçiş yapılır. Komut yazımı sırasında, klavyeden TAB tuşu kullanılarak otomatik tamamlama özelliğinden faydalanılabilir.



Şekil 17. PowerShell'de OpenCV klasörü oluşturmak

3.4.2. OpenCV

Python kullanıcıları, projelerinde ihtiyaç duydukları paket ve kütüphaneleri (örneğin, OpenCV, Keras, PyTorch) bilgisayara global olarak kurmak yerine, her proje için ayrı sanal ortamlar oluşturarak bu ortamlar içinde ilgili paketleri yüklemeyi tercih etmektedirler. Bu yaklaşım, Python'ın yoğun olarak kullanıldığı veri bilimi ve yapay zeka alanlarında yaygın ve etkili bir uygulamadır. İlgili alandaki en iyi uygulamalar göz önünde bulundurulduğunda, burada da aynı yöntemi kullanarak yükleme işlemi gerçekleştireceğiz. Sanal ortam kullanmanın en önemli avantajı, zamanla yeni sürümleri yayımlanan paket ve kütüphaneleri, başka projelerde çalışırken yeni sanal ortamlara yükleyebilmek ve böylece versiyon uyumu sorunlarından kaçınmaktır. Bu sayede, tek bir bilgisayarda birden fazla farklı proje üzerinde sorunsuz bir şekilde çalışabilmek mümkün olacaktır.

3.4.2.1. Sanal Ortam Oluşturulması

Yukarıda Şekil 24'deden devam edilecek olursa; aşağıdaki komutu girerek **opencv-env** isimli sanal ortam oluşturulur:.

```
py -m venv opencv-env
```

Ardından hiçbir yere gitmeden

```
.\opencv-env\Scripts\activate
```

yazarak **opencv-env** isimli sanal ortamı Şekil 18'de yeşil renkli gözüktüğü gibi aktif hâle getirilir.

```
PS C:\Users\TULPAR\Desktop\OpenCV> py -m venv opencv-env
PS C:\Users\TULPAR\Desktop\OpenCV> .\opencv-env\Scripts\activate
(opencv-env) PS C:\Users\TULPAR\Desktop\OpenCV>
```

Şekil 18. Sanal ortamın aktif hâle getirilmesi

3.4.2.2. OpenCV ve Bağımlı Paketlerin Sanal Ortama Kurulumu

Benzetme yaparak konuşacak olursak artık sanal ortamın içerisine adım atılmıştır. Artık hem yüklemeleri bu ortama yapılacak hem de ilerleyen zamanlarda yapılacak

projeleri bu ortamın içindeyken çalışacaktır. OpenCV ve bağımlı olduğu paketleri tek adımda yüklemek yerine iki adımda aşağıdaki komutlarla yüklenir⁶:

- pip install opencv-contrib-python streamlit jupyter moviepy ipykernel
- pip install matplotlib pyautogui mediapipe mime

3.4.3. Visual Studio Code

Python kullanıcıları genel olarak PyCharm'ı da sıklıkla kullanmaktadırlar. Ancak bugün Linux, MacOS ve Windows desteğiyle Visual Studio Code (VSC) dünyada en çok tercih edilen IDE konumundadır. Aşağıda Şekil 19'da VSC logosu görülebilir.



Şekil 19. Visual Studio Code logosu (URL-4)

Yazılacak kodları VSC'de yazıp Ctrl+Shft+P kısayolu ile **Python: Yorumlayıcı Seç**⁷ isimli menüyü açıp:

```
C:\Users\User\Desktop\OpenCV\opencv-env\Scripts
```

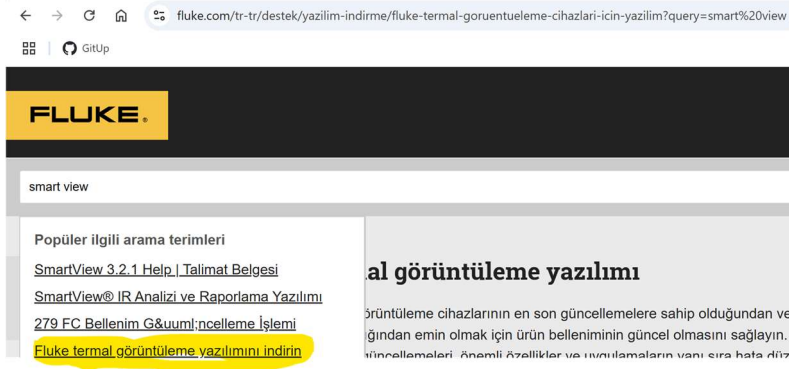
dizisinde yer alan **python.exe** dosyası seçildiğinde VSC kodu çalıştırmamız için **opencv-env** isimli sanal ortamı kullanım için aktif hâle getirmiş olacak. Böylece kodları PS'den çalıştırmak yerine görsel olarak daha dostça bir ortam olan VS Code'da koşturarak konsoldan bağımsız hale gelir. Bu da bize büyük projeler için kolaylık sağlayacaktır.

3.4.4. Termal resimlerin RGB Formata Dönüştürülmesi Ve Sıcaklık Haritasının Oluşturulması

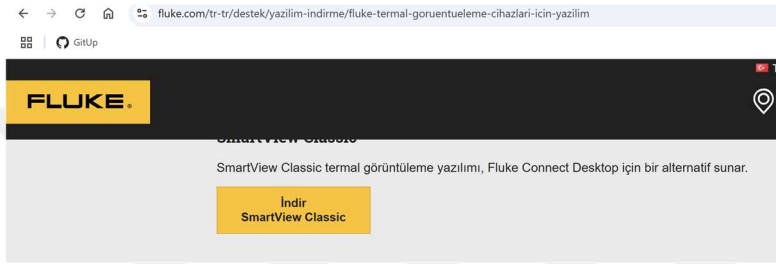
Fluke kamera ile çekilen resimler .IS2 formatındaki resimlerdir. Bunları açabilmek için Fluke kameranın SmartView uygulamasının indirilmesi gerekmektedir.

⁶ Bilgisayarınızın donanımına bağlı olarak yüklemeler esnasında bazı paketlerde uyarı ve hatalar karşınıza çıkabilir. Burada anahtar kelimeleri Google'da aratıp ilgili StackoverFlow sayfalarında çözümleri bulmayı denemek en makul hareket.

⁷ İng. Python: Select Interpreter

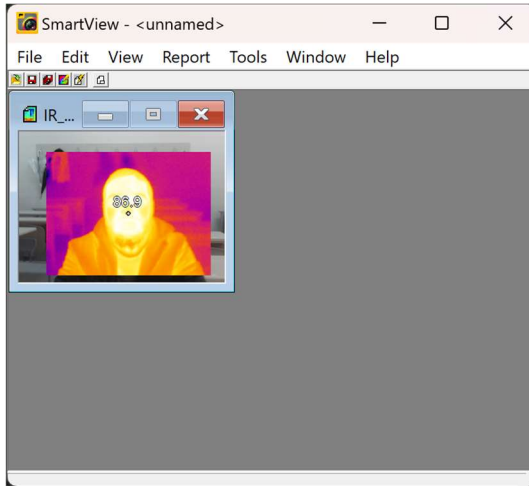


Şekil 20. SmartView Programının indirme (URL-5)



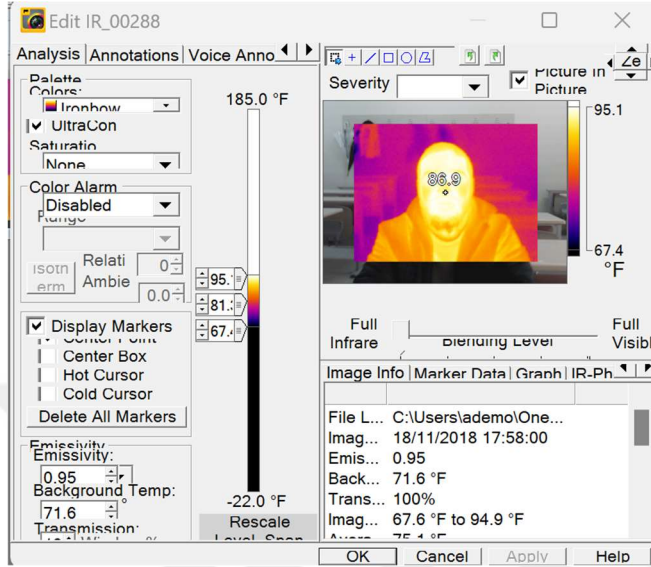
Şekil 21. SmartView Programının indirilmesi (URL-5)

Programı indirdikten sonra .IS2 uzantılı resme tıklanarak SmartView uygulamasında açılır



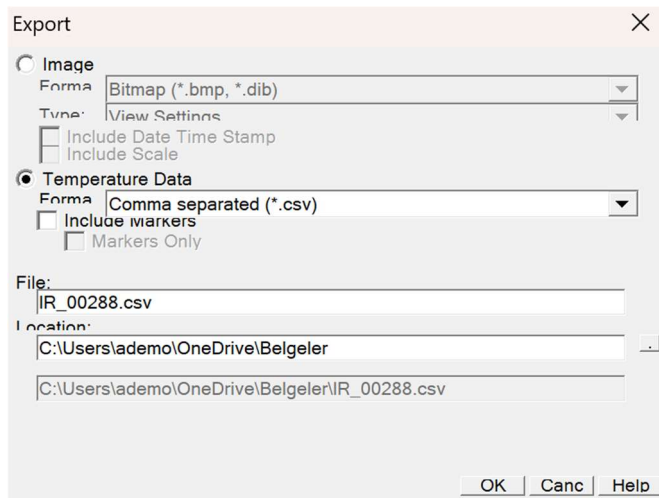
Şekil 22. SmartView Uygulamasında Resmin açılması

Resmin üstüne çift tıklanarak aşağıdaki ekrana girilir. Ve Display Markers Tık'ine tıklanarak resim üzerindeki sıcaklığın gitmesi sağlanır. Burada gösterilen değerler Fahrenheit cinsinden değerlerdir.



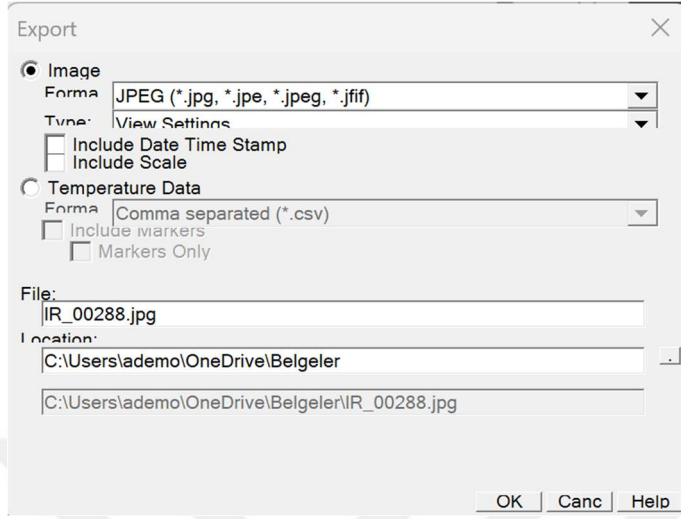
Şekil 23. SmartView Uygulamasında Resmin özellikleri

Burada sıcaklık haritasının çıkarılması için File/ Export yaparak ok butonuna basılır ve “.csv” uzantılı dosya elde edilir. Bu bir matrixtir ve ilk 5 karakter okutulmadan 6. Karakter itibarı ile sıcaklık matrixini bize verecektir.



Şekil 24. SmartView Uygulamasında Sıcaklık haritasının oluşturulması

RGB resim oluşturmak içinse aşağıda resimde olduğu gibi Image kısmı işaretlenir ve ok tuşuna basılır.



Şekil 25. SmartView Uygulamasında RGB Resim Oluşturmak

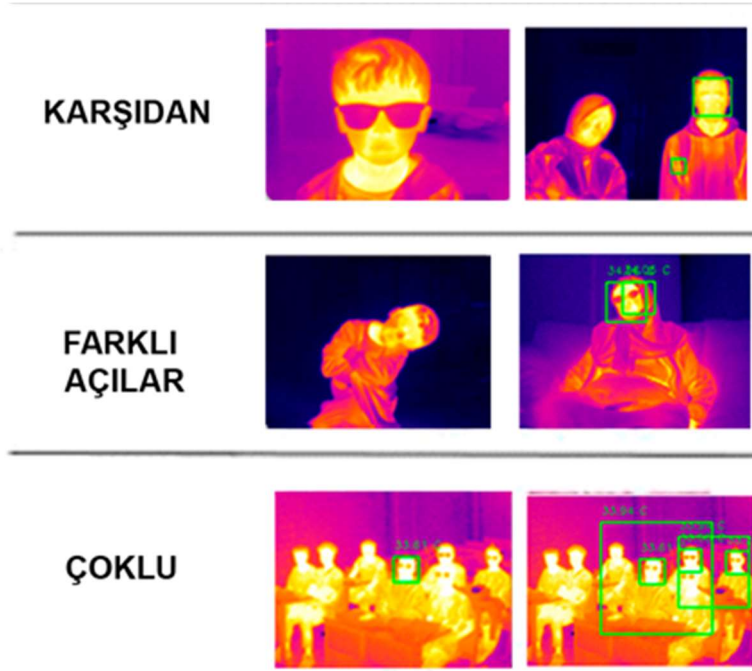
3.5. Kullanılan Veri Setleri

Tez çalışması kapsamında önerilen yaklaşımların test edilebilmesi için özgün veri setleri oluşturulmuştur. Bu veri setleri “termal yüz görüntüleri veri seti” dir. Termal görüntü veri seti termal çekimler ile elde edilmiş elde edilmiş termal yüz görüntülerini içermektedir.

Çalışma kapsamında kullanılan bu veri setleri için gerekli etik kurul raporu Gümüşhane Üniversitesi, Bilimsel Araştırma ve Yayın Etiği Kurulu, Biyoteknoloji Bölümünden ana bilim dallarından 27.06.gün ve 2024/6 sayılı yazıyla alınmıştır. Etik kurul raporu Ek-1’de sunulmuştur.

3.5.1. Termal Yüz Görüntüleri

Bu tez çalışması kapsamında oluşturulmuş olan veri seti termal görüntü içeren veri setidir. Termal görüntüler, termal kamera kullanılarak farklı açılardan farklı ortamlarda ve farklı kişi sayılarında fotoğraflar dijital ortama aktarılmıştır. Şekil 26’da örnek termal yüz görüntüleri gösterilmiştir.



Şekil 26. Termal görüntü örnekleri

Oluşturulan termal yüz görüntüsü veri setinde her bir çekim türüne ait görüntü sayıları Tablo 5’de verilmiştir. Orijinal görüntü boyutu 240x320 olup tasarlanan DÖ mimarisine bağlı olarak giriş görüntü boyutu yeniden boyutlandırılmıştır. Veri artırma yöntemleri kullanılarak eğitim ve test verisi sayısı artmıştır.

Tablo 5. Her termal çekim türüne ait görüntü sayıları

Görüntü Türü	Karşıdan	Farklı açılar	Çoklu
Adet	8	7	5

3.5.2. Veri artırma yöntemleri

Derin Öğrenme (DL) modelleri, eğitim için büyük veri setlerine ihtiyaç duyar. Ancak, büyük veri setlerini manuel olarak elde etmek genellikle mümkün değildir. Gerekli veriyi artırmak için veri artırma yöntemleri kullanılır. Veri artırma, derin ağlar için gerekli veriyi sağlamanın yanı sıra, DL ağlarının performansı üzerinde de olumlu bir etkiye sahiptir (Wong vd., 2016; Xu vd., 2016; Perez ve Wang, 2017; Uyar, 2022). Veri setleri gerçek hayatta çeşitli bozulma etkilerine maruz kaldığından, veri artırma yöntemleri modelin farklı koşullar altında öğrenmesine yardımcı olur ve sınıflandırma başarısını artırır.

Simetrik dönüşüm, renk oranlarını değiştirme, gürültü ekleme, görüntüleri belirli bir dereceye kadar döndürme, görüntüleri belirli eksenler boyunca çevirmeme ve görüntülerin belirli kısımlarını kesme gibi çeşitli veri artırma yöntemleri, yeni veri örnekleri oluşturmak için kullanılabilir. Bu yaklaşım ayrıca modelin aşırı uyumdan (overfitting) kaçınmasına yardımcı olur. Veri artırma yöntemlerine örnekler arasında yansıma, döndürme, gürültü ekleme, histogram eşitleme, kaydırma ve yeniden ölçeklendirme yer alır. Bu tezde, veri örneklerinin sayısını artırmak için döndürme, yansıma ve parlaklık ayarları kullanılmıştır.

3.6. Tasarlanan Modeller

3.6.1. Haar Cascade Yöntemi ile Yüz Tespiti

3.6.1.1. Python Kodunda Haar Cascade Uygulaması

3.6.1.1.1. Modelin Yüklenmesi

Haar Cascade algoritması, OpenCV kütüphanesinden alınan önceden eğitilmiş bir model (haarcascade_frontalface_default.xml) ile çalıştırılmıştır. Bu model, yüz özelliklerini tanımlayabilmek için optimize edilmiştir.

3.6.1.1.2. Görüntünün Hazırlanması

RGB görüntüler ve termal görüntüler, algoritmanın çalışabilmesi için gri tonlamalı forma dönüştürülmüştür. Gri tonlama, Haar Cascade algoritmasının temel gereksinimlerinden biridir. RGB görüntülerde tespit edilen yüzler, aynı çözünürlükteki termal görüntülere işaretlenmiştir. Böylece, termal görüntülerde doğrudan yüz tespiti yapmanın sınırlamaları aşılmıştır.

3.6.1.1.3. Parametre Ayarları ve Yüz Tespiti

Algoritmanın başarımı, **scaleFactor** ve **minNeighbors** parametrelerine bağlıdır:

- **scaleFactor:** Farklı boyutlardaki yüzleri tespit etmek için kullanılan ölçeklendirme faktörü. Bu çalışmada, scaleFactor değeri 1.05 olarak belirlenmiştir.

- **minNeighbors:** Bir alanın yüz olarak kabul edilmesi için gerekli olan komşu bölgelerin sayısı. Çalışmada bu parametre 25 olarak ayarlanmıştır.

RGB görüntülerde yapılan tespit, termal görüntülerdeki yüz koordinatlarının belirlenmesinde referans alınmıştır.

3.6.1.1.4. Tespit Edilen Yüzlerde Sıcaklık Analizi

Tespit edilen yüz bölgelerinde, maksimum sıcaklık değerleri analiz edilmiştir. Termal görüntülerdeki sıcaklıklar, Fahrenheit'tan Celsius birimine dönüştürülerek görüntü üzerine metin olarak işlenmiştir.

3.6.1.1.5. Görselleştirme

Algoritmanın tespit sonuçları, yüzlerin etrafına çizilen dikdörtgenlerle ve sıcaklık değerlerinin görüntü üzerine yazılmasıyla görselleştirilmiştir.

3.6.2 Derin Öğrenme ve Konvolüsyonel Sinir Ağları Tabanlı Derin Öğrenme Yöntemi ile Yüz Tespiti

3.6.2.1. Derin Öğrenme

Yapay Zeka (YZ), en geniş tanımıyla, makinelerin insan gibi düşünmesini ifade eder. Diğer bir deyişle, bilgisayar sistemlerinin, insan zekasına bağlı olarak öğrenme, akıl yürütme ve kendini düzeltme gibi süreçleri ve faaliyetleri gerçekleştirme yeteneğidir.

Derin öğrenme (DL), 1980'lerde ortaya çıkan ve sistemlerin verilerden otomatik olarak öğrenmesini, doğru tahminlerde bulunmasını ve deneyime dayalı olarak gelişmesini sağlayan bir YZ uygulamasıdır. Derin öğrenme algoritmalarının verimliliği ve başarısı, büyük ölçüde giriş verilerinin ne kadar iyi temsil edildiğine bağlıdır. Ham verilerden özellik çıkarımı, alan özeldir ve önemli bir insan çabası gerektirir. Derin öğrenme algoritmaları, denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme olarak kategorize edilir. Denetimli öğrenmede, öğrenme için giriş verileri ile etiketli çıktı verileri arasında ilişkiler kurulur. Bu öğrenme türünün temel uygulamaları sınıflandırma ve regresyon problemleridir. Denetimsiz öğrenmede, yalnızca giriş verileri kullanılarak öğrenme yapılır ve etiketli çıktı verisi kullanılmaz. Diğer bir

deyişle, önceden tanımlanmış etiketler olmadan, bir veri kümesinde kendi kendine organizasyon yoluyla bilinmeyen yapılar bulunur. Temel bileşen analizi ve kümeleme, bu öğrenme türünün başlıca uygulamalarıdır. Pekiştirmeli öğrenme, çevresini algılayabilen ve kendi başına kararlar verebilen bir sistemin, ödül-ceza yöntemlerini kullanarak amacına ulaşmak için doğru kararlar vermeyi nasıl öğrendiğini gösterir. Sistem, doğru kararlar için ödüllendirilir ve yanlış kararlar için cezalandırılır. Robotik ve oyun programlama problemleri, bu öğrenme türünün ana uygulamalarıdır (Uyar, 2022).

Derin öğrenmenin modern tarihi, 1943'te McCulloch-Pitts (MCP) modelinin tanıtılmasıyla başlamıştır (McCulloch ve Pitts, 1943). Bu model, yapay sinir ağı (YSA) modellerinin prototipi olarak bilinir. İnsan beyninin neokorteksini fonksiyonel olarak taklit eden, sinir ağlarına dayalı bir bilgisayar modeli oluşturulmuştur (Schmidhuber, 2015). Tasarlanan modelde, insan düşünme sürecini taklit etmek için değil, öğrenmek için kullanılan "eşik mantığı" adı verilen algoritmalar ve matematik kombinasyonu kullanılmıştır. MCP modelinden sonra, biyolojik sistemler için doğal ortamda kullanılan Hebbian teorisi uygulanmıştır. Daha sonra, 1958'de "perceptron" adı verilen ilk elektronik cihaz, bilişsel sistem çerçevesi altında piyasaya sürülmüştür. Werbos, derin öğrenme modellerini eğitirken hataları kullanan geri yayılım algoritmasını tanıtmış ve sinir ağlarına kapı açmıştır. 1980'de "neokognitron" tanıtılmış, bu model konvolüsyonel sinir ağlarını (CNN'ler) ilham kaynağı olmuştur (Fukushima, 1980). Ancak model, eğitim sürecinde yetersiz donanım performansı nedeniyle büyük ilgi görmemiş ve yaygınlaşmamıştır. 1990'larda, LeNet derin ağ mimarisi (LeCun vd., 1998) geliştirilmiş ve bu mimari, CNN mimarisini kullanarak el yazısı rakamlarını başarıyla sınıflandırmıştır. 2006'da derin inanç ağları geliştirilmiş ve araştırmacılar, önceki denemelerden çok daha derin sinir ağlarını eğitmek için bu tekniği kullanmışlardır. Bu gelişme, derin öğrenmede sinir ağlarının yeniden önem kazanmasına yol açmıştır.

Özellikle 2010'larda, derin öğrenme önemli bir popülerlik kazanmış ve YSA mimarisine dayalı ve çok katmanlı sinir ağlarının hesaplanmasını sağlayan algoritmalar haline gelmiştir. Geleneksel makine öğrenme yöntemlerinde sınıflandırma için özelliklerin manuel olarak girilmesi gerekirken, derin öğrenme sistemlerinde özellikler sistem tarafından otomatik olarak bulunur. Tipik bir sinir ağı modeli birkaç katmandan

oluşurken, derin öğrenmede gizli katmanların sayısı çok daha büyüktür. Derin öğrenmedeki "derin" terimi, çok sayıda gizli katmana sahip olmayı ifade eder. Özellikle Grafik İşlem Birimleri (GPU'lar) gibi donanım altyapısındaki gelişmeler sayesinde, veri analiz süresi kısalmış ve birçok gizli katmanı olan derin YSA'ların neden olduğu hesaplama maliyeti azalmıştır (Uyar, 2022). Şekil 27, yapay zeka, makine öğrenimi ve derin öğrenme kavramlarının zaman içinde gelişimini göstermektedir.



Şekil 27. YZ, MÖ ve DÖ'nün zaman içerisindeki gelişimi (URL-6)

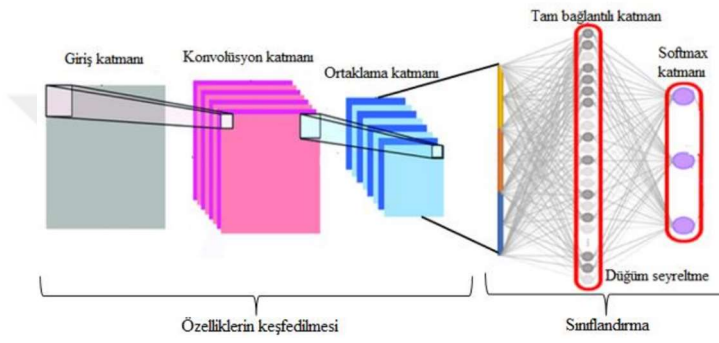
Literatürde sıklıkla kullanılan yaygın derin öğrenme yöntemleri arasında Konvolüsyonel Sinir Ağları (CNN), Tekrarlayan Sinir Ağları (RNN), Uzun Kısa Süreli Bellek (LSTM) Ağları, Kısıtlı Boltzmann Makineleri (RBM), Derin Otomatik Kodlayıcılar (DAE) ve Derin İnanç Ağları (DBN) bulunmaktadır.

Özetle, derin öğrenme, özellikle son zamanlarda dikkat çekici bir ilerleme kaydeden makine öğrenme tekniklerinden biri olup, bilgisayar mühendisliği alanında yaygın olarak kullanılmaktadır. Derin öğrenme kavramı, YZ ve makine öğrenmesinde yeni bir kavram olarak ortaya çıkmıştır. Amacı, düşük seviyeli özellikleri daha yüksek seviyedeki özelliklerle birleştirerek bir özellik hiyerarşisini öğrenmektir (Bengio, 2009).

3.6.2.2. Konvolüsyonel Sinir Ağları

Derin öğrenmenin temel mimarisi olarak kabul edilen Konvolüsyonel Sinir Ağları (CNN), görüntü sınıflandırma, görüntü konumlandırma, nesne tespiti ve görüntü semantik segmentasyonu gibi görüntü analiz görevlerinde başarıyla uygulanmaktadır. CNN'ler, özellikle büyük ölçekli görüntü tanıma yarışması olarak bilinen ImageNet

Large Scale Visual Recognition Challenge (ILSVRC) ile görüntü işleme alanında önemli bir uygulama alanı bulmuştur. Geleneksel görüntü sınıflandırma yöntemlerinde, ön işleme, özellik çıkarımı ve özellik seçimi adımları ortadan kaldırılmış ve bu işlemler CNN mimarisi kullanılarak yapılmaktadır. CNN'lerin yüksek performansı, büyük ölçüde hiyerarşik özellik çıkarım yeteneklerine dayanmaktadır. Görüntüler ağ içine verilir ve öğrenme, herhangi bir özellik çıkarım yöntemi kullanılmadan oluşturulan bir özellik hiyerarşisi aracılığıyla otomatik olarak yapılır. Genel bir CNN yapısı Şekil 28'de gösterilmektedir (Shen et al., 2017)

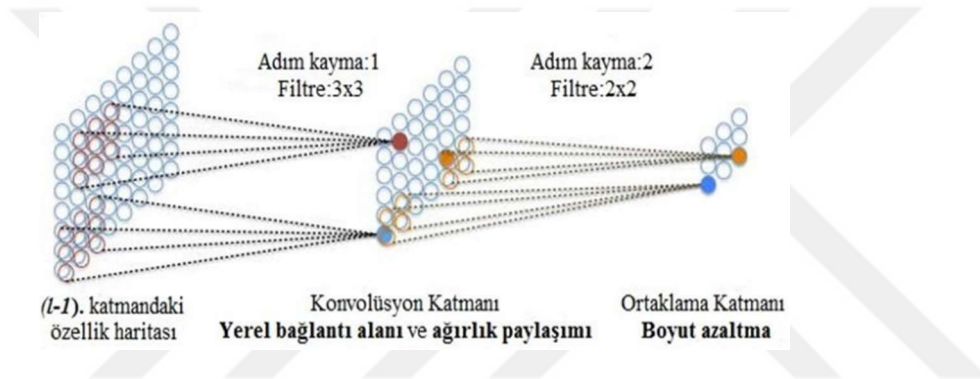


Şekil 28. CNN mimarisinin genel yapısı (Shen et al., 2017)

Şekil 19'da görüldüğü gibi, CNN yapısı genellikle iki bölüme ayrılır. İlk olarak, giriş görüntüsünün belirgin özellikleri, konvolüsyon, ReLU ve havuzlama katmanlarından oluşan bloklarda keşfedilir. Sonrasında ise sınıflandırma, tamamen bağlı katmanlar, ReLU ve dropout katmanları kullanılarak yapılır.

İlk bölümde, belirli bir boyutta giriş görüntüsü ağ içine verilir. Bu katman, giriş katmanı olarak bilinir ve ya 3 kanallı RGB görüntü ya da tek kanallı bir görüntü kullanabilir. Bu katmandan sonra, konvolüsyon katmanı gelir. Bu katman, özellik keşfi için en önemli katman olarak kabul edilir ve özellikler, 2×2 , 3×3 , 5×5 , 7×7 gibi farklı boyutlarda filtreler (çekirdekler, ağırlık matrisleri) kullanılarak çıkarılır. Her konvolüsyon filtresi, eğitim sırasında öğrenilen parametreleri içerir. Bu filtreler, belirli bir adım boyutunda giriş görüntüsünün üzerine konvolüve edilir ve çıktı görüntüsünü üretir. Konvolüsyon katmanından sonra, bu çıkış görüntüleri, filtre sayısına karşılık gelen özellik haritaları olarak adlandırılır. Bir sonraki katman olan ReLU katmanında, özellik haritalarındaki negatif değerler sıfıra ayarlanırken, diğer değerler doğrudan

çıkıya geçer. ReLU katmanından sonra, havuzlama katmanı gelir. Bu katmanın amacı, görüntü boyutunu küçültmek ve dolayısıyla hesaplama maliyetlerini azaltmaktır. İlk bölümdeki katmanların farklı kombinasyonları, çeşitli CNN mimarileri oluşturmak için kullanılabilir. İlk bölümdeki özellik çıkarımının ardından, sınıflandırmanın yapıldığı ikinci bölüm gelir. Tamamen bağlı katman, önceki katmanın tüm alanlarıyla bağlanır. Bu katmandan sonra, ağız ezberlemesini engellemek için dropout katmanı kullanılır. Bu katmanda bazı nöronlar ve bağlantılar rastgele devre dışı bırakılır. Bunun ardından, softmax katmanı, nesnenin 0-1 aralığında olasılık değerini hesaplar ve uygun sınıfa atar. Softmax sınıflandırıcısı yerine başka sınıflandırma algoritmaları da kullanılabilir. Şekil 29, CNN tarafından sunulan üç önemli mekanizmayı görsel olarak sunmaktadır.



Şekil 29. CNN'nin üç önemli mekanizması (Yerel alıcı alan, ağırlık paylaşımı, boyut indirgeme) (Shen et al., 2017)

CNN yapısındaki konvolüsyon ve havuzlama katmanları sayesinde, görüntüden daha fazla bilgi çıkarılabilir ve ağda eğitilmesi gereken parametre sayısı büyük ölçüde azaltılır. Örneğin, $300 \times 300 \times 3$ boyutlarında bir görüntüyle çalıştığımızı varsayalım, bu görüntü bir vektöre dönüştürülürse 27.000 parametre içerir. Ayrıca, gizli katmanların sayısı ve her gizli katmandaki nöron sayısı göz önüne alındığında, milyonlarca parametrenin eğitilmesi gerekir. Ağırlık paylaşımı özelliğiyle, bir filtre, görüntüdeki tüm pikselleri tarar, bu nedenle aynı ağırlık değerleri farklı nöronlara bağlanır. Yerel bağlantı özelliğiyle, filtre tarafından işlenen her parça, bir sonraki katmandaki sadece bir nöronla bağlanır. Bu nedenle, her nöron yalnızca görüntünün bir kısmından sorumludur ve diğer pikseller bu çıktıyı etkilemez.

3.6.2.3. CNN'nin katmanları

3.6.2.3.1. Giriş katmanı

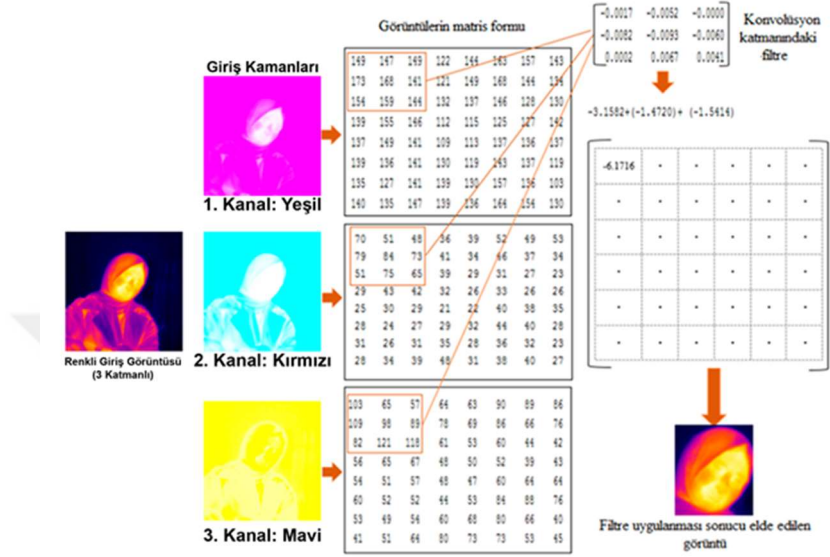
Bu, modelin ilk katmanıdır ve ham veriler ağı beslenir. Bu katmandaki verilerin boyutu, tasarlanan modelin başarısı için önemli bir parametredir. Büyük bir giriş görüntü boyutu, yüksek bellek gerektirir, eğitim süresini artırır ve her görüntü için test süresini uzatır. Öte yandan, küçük bir giriş görüntü boyutu seçmek, bellek gereksinimlerini azaltır ve eğitim süresini kısaltır, ancak ağın performansını düşürebilir. Görüntü analizinde, uygun bir giriş görüntü boyutu, ağ derinliği, donanım hesaplama maliyeti ve ağ performansı göz önünde bulundurularak seçilmelidir (Özkan ve Ülker, 2017).

3.6.2.3.2. Konvolüsyon katmanı

Konvolüsyon katmanı, CNN mimarisinin temelini oluşturan ve dönüşüm katmanı olarak da bilinen katmandır, görüntü üzerinde filtreler kullanarak konvolüsyon işlemi uygular. Dönüşüm işlemi, belirli bir filtreyi tüm görüntü üzerinde konvolüsyon yapma ilkesine dayanır. Bu katmanda, önceki katmandan gelen görüntüler üzerine filtreler kullanılarak konvolüsyon uygulanır ve çıktı verisi üretilir. Bu çıktı verisi, özellik haritası olarak adlandırılır. Bir özellik haritası, her filtreye özgü özelliklerin keşfedildiği bölgelere sahiptir. Görüntünün özellikleri, eğitim aşamasının sonunda ortaya çıkar. Tasarlanan CNN modelindeki konvolüsyon katmanlarının sayısına bağlı olarak temel nitelikler, temel desenler ve ileri düzey desenler elde edilir. Konvolüsyon katmanlarında kullanılan filtreler, görüntüde en yüksek yanıtları veren bölgeleri ortaya çıkarır. Bu nedenle, görüntüdeki aranan nesne, yönü ve boyutu ne olursa olsun tespit edilebilir.

Bir 3×3 filtresinin bir görüntü üzerindeki uygulaması Şekil 21'de gösterilmiştir. Termal görüntüler, 3 kanaldan oluşan renkli görüntülerdir. Bu kanallardan bir kısmı matris biçiminde ifade edilmiştir. Tasarlanan CNN modelleri, konvolüsyon katmanlarında farklı sayı ve boyutlarda filtreler içerir. Filtre matrisinin, görüntünün karşılık gelen piksel değerleriyle çarpılması ve toplamın elde edilmesi sağlanır. Bu işlem her görüntü için uygulandıktan sonra, elde edilen değerlerin ortalaması, konvolüsyon sonucunun karşılık gelen pikseline yazılır. Bu filtreleme işlemi, belirli bir adım boyutuna göre görüntüye uygulanır. Bir örnek olarak, Haar Cascade modeli kullanılır ve ilk konvolüsyon katmanında bir filtre uygulandıktan sonra oluşan özellik

haritasının örnek gösterimi Şekil 30’da verilmiştir. Haar Cascade modelinin ilk konvolüsyon katmanında 7×7 boyutunda 64 filtre bulunmaktadır, böylece her bir filtre, giriş görüntüsüne uygulandığında farklı görüntüler elde edilir. Şekil 30’da görüldüğü gibi, her filtrenin giriş görüntüsü üzerindeki etkisi farklıdır.

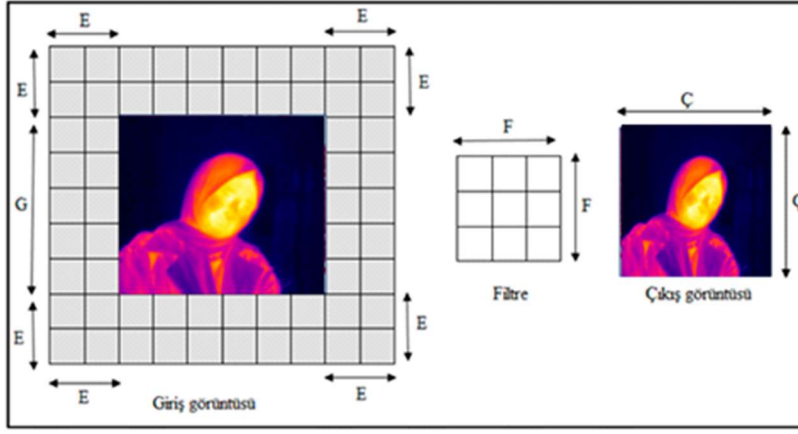


Şekil 30. Katmanındaki filtrelerin konvolüsyonu

Konvolüsyon katmanında, filtrelerin görüntüye uygulanması, çıktı görüntüsünün boyutunu değiştirir. Çıktı görüntüsünün boyutunu hesaplamak için kullanılan denklem, Denklem 3’te verilmiştir. Denklem 3’deki değişkenler Şekil 8’de gösterilmektedir.

$$Ç = \frac{G-F+2 \cdot E}{A} + 1 \quad (3)$$

Bu denklemde, G giriş görüntüsünü, F uygulanan filtrenin boyutunu, E giriş görüntüsü matrisine eklenecek satır ve sütun sayısını ve A ise filtreye konvolüsyon yapılacak adım boyutunu temsil eder. Örneğin, giriş görüntüsü 16×16 ise, filtre boyutu 3×3, padding 1 ve adım boyutu 1 ise, çıktı görüntü boyutu Denklem 4’te olduğu gibi 16 olarak hesaplanır.



Şekil 31. Boyut hesabında kullanılan değişkenler

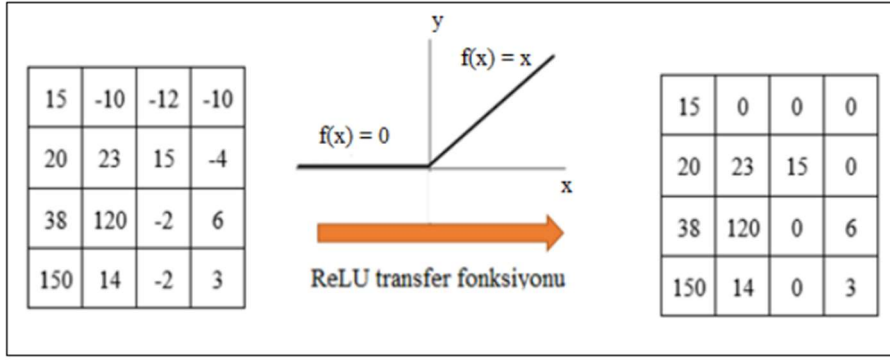
$$\Ç = \frac{16-3+ *1}{1} + 1 \quad (4)$$

Görüntüdeki kenarlar, giriş verilerinin yüksek frekanslı bölgelerini temsil eder ve kenar bilgisi, bu katmanda çıkarılan en önemli özelliklerden biridir. Konvolüsyon katmanındaki özellik çıkarıcıları olarak bilinen konvolüsyon filtreleri aracılığıyla derin ağlar, nesnenin belirli özelliklerini bulmaya çalışır ve ağ eğitimi boyunca bu özellikler ilgili kategori ile ilişkilendirilir.

3.6.2.3.3. Düzeltmeli doğrusal birim (ReLU)

Aktivasyon katmanı, genellikle konvolüsyon katmanı tarafından oluşturulan özelliklerin ardından kullanılan, aynı zamanda ReLU (Rectified Linear Unit) katmanı olarak da bilinen katmandır. Matematiksel olarak ReLU, Denklem 5’de gösterildiği gibi tanımlanır ve Şekil 32’de görüldüğü gibi, gelen değeri sıfırdan sonsuza kadar bir çıkışa dönüştürür. ReLU, doygun olmayan bir fonksiyon olarak bilinir ve hesaplamaları çok hızlı bir şekilde gerçekleştirebilir. Bu katmanın giriş verisi üzerindeki etkisi, negatif değerleri sıfıra ayarlarken pozitif değerleri olduğu gibi çıkışa iletmektir. Bu katmandan önce bir konvolüsyon katmanı belirli matematiksel işlemler gerçekleştirir, bu nedenle ağ doğrusal bir yapıya sahiptir. Bu katman, ağı doğrusal olmayan hale getirmek için uygulanır.

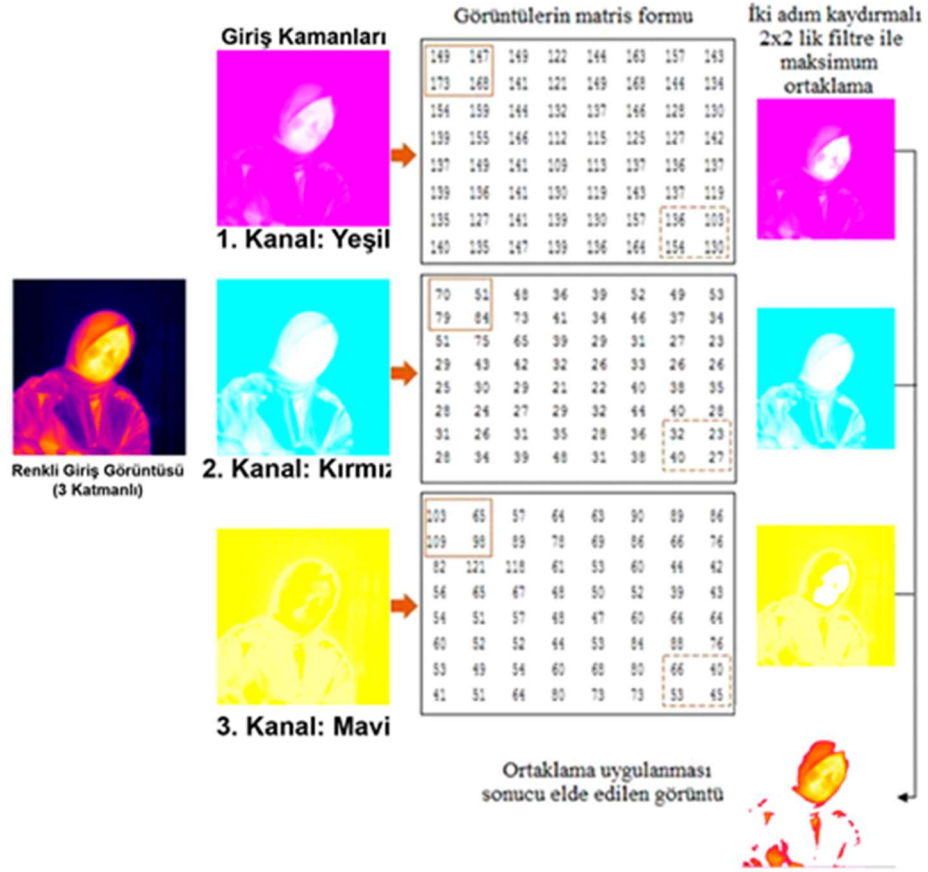
$$f(x) = f(x) = \begin{cases} 0, & \text{eğer } x < 0 \\ x, & \text{eğer } x \geq 0 \end{cases} \quad (5)$$



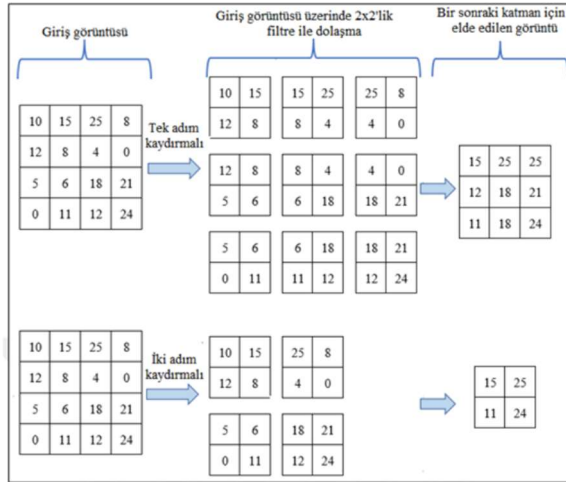
Şekil 32. ReLU katmanının çıktı verisine etkisi

3.6.2.3.4. Havuzlama Katmanı

Bu katmanda, ReLU katmanı tarafından düzeltilmiş olan özellik matrisinde alt örnekleme yapılır. Alt örnekleme, verilen adım boyutuna göre filtre boyutunun düzeltilmiş özellik matrisi üzerinde kaydırılması ve matrisi alt bölgelerine ayırma işlemini içerir. Hesaplama karmaşıklığını azaltmak için kullanılan bu katman, giriş matrisindeki kanal sayısını sabit tutarken yükseklik ve genişlik bilgisini azaltır. Böylece, hesaplama ve parametre sayısı azaldıkça, verinin hacimsel boyutu da sürekli olarak azalır. Bu katmanda öğrenilebilir parametreler yoktur, ancak verideki önemli bilgilerin kaybolmasına yol açabileceği için performansı düşürebilir. Maksimum, minimum ve ortalama havuzlama, kullanılan yöntemler arasındadır. Her havuzlama işlemi, mevcut matrisin maksimum, minimum veya ortalama değerini alır. Havuzlama katmanının gerçek görüntü uygulaması Şekil 33’de gösterilmiştir ve matematiksel uygulaması Şekil 34’de verilmiştir.



Şekil 33. 4×4 giriş görüntüsüne maksimum havuzlama için bir ve iki adımlı kaydırmalarla 2×2 filtre uygulamanın matematiksel temsili



Şekil 34. 4×4 giriş görüntüsüne maksimum havuzlama için bir ve iki adımlı kaydırmalarla 2×2 filtre uygulamanın matematiksel temsili

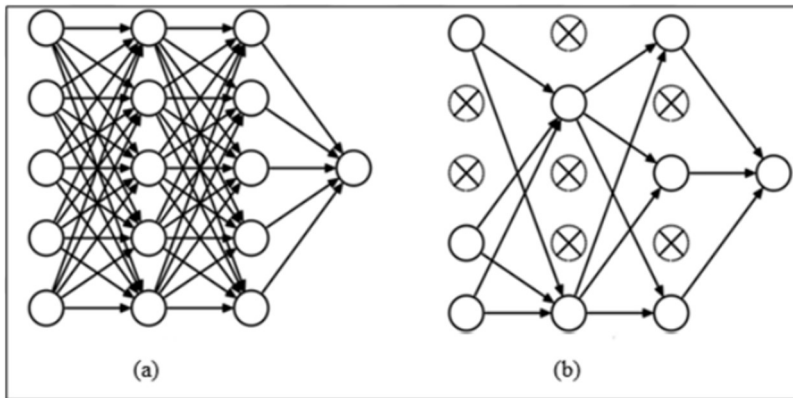
3.6.2.3.5. Tam bağlantılı katman

CNN modellerinde ardışık konvolüsyon, ReLU ve havuzlama katmanları kullanılarak yüksek seviyede özellikler elde edildikten sonra, ağın sonuna tam bağlantılı katman eklenir. Bu katman, önceki katmanın tüm bölgeleriyle bağlantılıdır. İlk tam bağlantılı katmandan önceki katman çok boyutludur ve tam bağlantılı katmana geçiş sırasında düzleştirme işlemi yapılır. Modelde kullanılan tam bağlantılı katmanların sayısı, yapılan işe bağlı olarak değişebilir. Tam bağlantılı katmana geçiş alanı, CNN modellerinde en fazla parametrenin kullanıldığı alandır ve bu alanda kullanılan parametre sayısı, tasarlanan CNN modelinin hesaplama karmaşıklığı göz önünde bulundurularak dikkatlice seçilmelidir.

Örneğin, CNN mimarisinin son katmanından üretilen matrisin boyutu $50 \times 50 \times 256$ ise, tam bağlantılı katmandaki matris boyutu 2048×1 olarak seçilirse, toplam matris boyutu 640.000×2048 olur. Bu nedenle, 640.000 nöronun her biri, 2048 nörona bağlı olacaktır. Bu nedenle, bu katman tam bağlantılı katman olarak adlandırılır.

3.6.2.3.6. Düşüm seyreltme katmanı

Dropout, sistemin ezberleme sorununu çözmek için kullanılan bir yaklaşımdır. Bu katmanın ana fonksiyonu, ağın bazı düğümlerini ve bağlantılarını rastgele kaldırmaktır. Bazı nöronlar ve bağlantılar rastgele kaldırılarak daha ince bir ağ yapısı oluşturulur ve ezberleme sorununu önlemek için daha küçük ağırlıklara sahip bir ağ elde edilir. Şekil 35’de ağın orijinal yapısı gösterilmiştir.



Şekil 35. CNN ağında dropout katmanının uygulanması (Wang et al., 2020a-b)

3.6.2.3.7. Sınıflandırma katmanı

Tam bağlantılı katmandan sonra, bu katmanda sınıflandırma işlemi gerçekleştirilir. Bu katmanın çıkış değeri, sınıflandırılacak nesne sayısına bağlıdır. Örneğin, 5 farklı doku örneğinin sınıflandırıldığı histoloji veri setinde çıkış değeri 5 olarak seçilir. Bu katmanda farklı sınıflandırıcılar kullanılabilir, ancak sınıflandırma başarısı nedeniyle genellikle softmax sınıflandırıcısı tercih edilir. Softmax sınıflandırıcısı, giriş vektörünün mevcut sınıflara ait olma olasılığını hesaplar. Bu sınıflandırıcı kullanıldığında, her nesne için aralığında bir değer üretilir. Sınıflandırma katmanı, giriş vektörünün sınıfını belirler. Nesne, en yüksek çıkış değerine karşılık gelen sınıf ile etiketlenir.

3.6.2.2. Nesne sınıflandırmada kullanılan CNN modelleri

1998 yılında, LeCun vd. (1998) LeNet adı verilen basit bir CNN mimarisi tanıttılar ve bu mimari rakam sınıflandırmasında kullanıldı. O zamandan bu yana birçok CNN mimarisi önerilmiştir. Grafik işleme birimleri (GPU'lar) ve bilgisayar donanımının gelişimiyle birlikte, CNN mimarileri zamanla daha derin hale gelmiş ve farklı yapıları olan mimarilerin oluşturulmasına yol açmıştır.

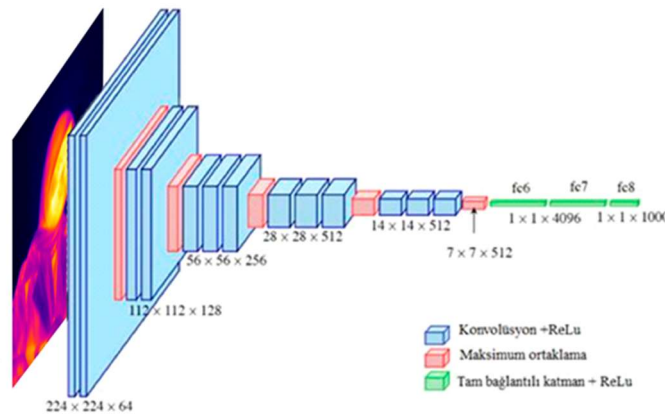
Bu tezde, AlexNet (Krizhevsky vd., 2012), DenseNet201 (Huang vd., 2017), EfficientNet (Tan ve Le, 2019), GoogleNet (Szegedy vd., 2015), InceptionResNetv2 (Szegedy vd., 2017), Inceptionv3 (Szegedy vd., 2016), MobileNetv2 (Sandler vd., 2018), ResNet18 (He vd., 2016), ResNet50 (He vd., 2016), ResNet101 (He vd., 2016), ShuffleNet (Zhang vd., 2018), SqueezeNet (Iandola vd., 2016), Vgg16 (Simonyan ve Zisserman, 2014) ve Vgg19 (Simonyan ve Zisserman, 2014) gibi CNN modelleri kullanılarak performans karşılaştırmaları yapılmıştır. Bu sıkça kullanılan önceden eğitilmiş CNN mimarilerinin bazı özellikleri Tablo 6'da verilmiştir. Kullanılan CNN modelleri, yığılmış, artımlı, inception, yoğun ve fire blokları gibi farklı yapılara sahiptir. Bu blok yapılarının özellikleri, bir sonraki bölümde detaylı olarak açıklanacaktır.

Tablo 6. Literatürde yaygın olarak kullanılan önceden eğitilmiş CNN mimarilerinin bazı özellikleri

CNN modeli	İçerdiği blok	Derinlik	Parametre sayısı (milyon)	Görüntü çözünürlüğü
1. AlexNet	İstifleme	8	61.0	227x227
2. DenseNet201	Yoğun	201	20.0	224x224
3. EfficientNet	Kalıntı	82	5.30	224x224
4. GoogleNet	Başlangıç	22	7.00	224x224
5. InceptionResNetv2	Başlangıç + Kalıntı	164	55.9	299x299
6. Inceptionv3	Başlangıç	48	23.9	299x299
7. MobileNetv2	Kalıntı	53	3.50	224x224
8. ResNet18	Kalıntı	18	11.7	224x224
9. ResNet50	Kalıntı	50	25.6	224x224
10. ResNet101	Kalıntı	101	44.6	224x224
11. ShuffleNet	Kalıntı	50	1.40	224x224
12. SqueezeNet	Ateşleme	18	1.24	227x227
13. Vgg19	İstifleme	19	144	224x224

3.6.2.2.1. İstifleme (Stacked) blok içeren modeller

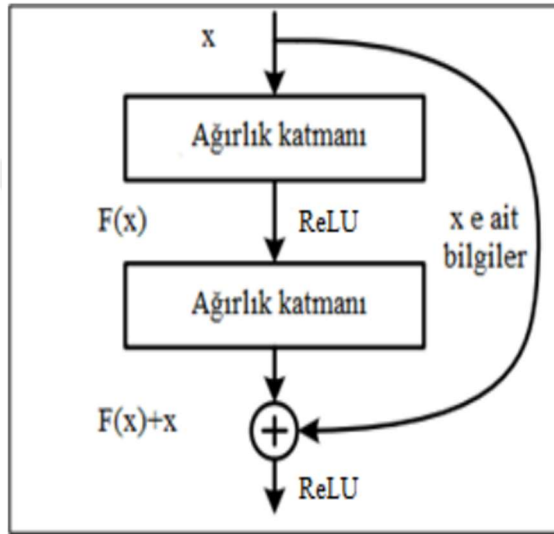
AlexNet, Vgg16 ve Vgg19 gibi ilk geliştirilen CNN mimarileri, yığılmış konvolüsyon ve havuzlama katmanlarından oluşmaktadır. Zamanla, katmanların yığılması ve çok sayıda filtre eklenmesinin, bu mimarilerin hesaplama ve bellek maliyetlerini artırdığı ve aşırı uyum sağlama olasılığını yükselttiği gözlemlenmiştir. Daha sonra geliştirilen mimarilerde, yaklaşım değiştirilmiş ve farklı yapıya sahip bloklar oluşturulmuştur. Yığılmış bir bloğa örnek, Şekil 36'da VggNet mimarisıyla gösterilmiştir.



Şekil 36. Blok İstif yapıları (VggNet mimari örneği) (Simonyan ve Zisserman, 2014)

3.6.2.2.2. Kalıntı (Residual) blok içeren modeller

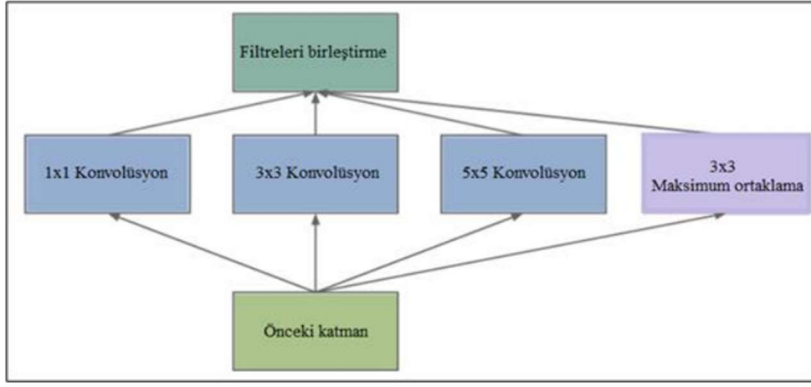
Teorik olarak, bir modelin katman sayısı arttıkça performansının artacağı varsayılmıştır; ancak pratikte durum böyle olmamıştır. Bunun üzerine artımlı bloklar içeren CNN modelleri geliştirilmiştir. Artımlı blokların yapısı, Şekil 37’de gösterilmiştir. Bir artımlı blokta, konvolüsyon-ReLU-konvolüsyon serisinin ardından bir sonuç $F(x)$ elde edilir. Bu sonuç, giriş x ile toplanır ve çıkış değeri $H(x) = F(x) + x$ olarak ifade edilir.



Şekil 37. Kalıntı blok yapısı (He vd., 2016)

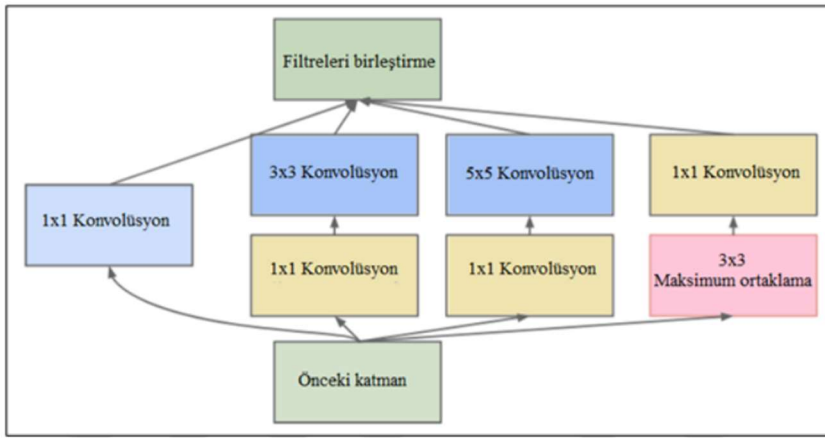
3.6.2.2.3. Başlangıç (Inception) blok içeren modeller

Paralel bağlanan inception blokları kullanılmıştır. Bu model, önceki ağ modellerinden farklıdır ve anlaşılması zordur. Ancak, hesaplama karmaşıklığı ve boyut için bulunan çözüm, hız ve performans iyileştirmeleri sağlamaktadır. Şekil 30’da görüldüğü gibi, inception bloğu 3×3 maksimum havuzlama ve 1×1 , 3×3 , 5×5 konvolüsyon katmanlarından oluşur.



Şekil 38. Başlangıç blok yapısı

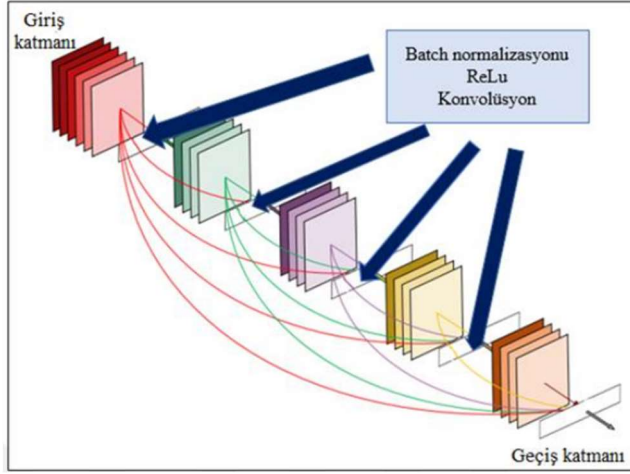
İncepion bloğu, bir önceki katmandan gelen verilere uygulandıktan sonra, filtreler çıkışta birleştirilir ve yeni veriler bir sonraki katmana iletilir. Bu bloklar, farklı özellikler keşfetmeye çalışır. Bu yapıda yapılan konvolüsyon işlemleri sırasında parametre sayısı artar, bu da hesaplama maliyetini artırır. Bu sorunu çözmek için, Şekil 39'da gösterildiği gibi, boyut indirgeme işlemi 1×1 konvolüsyon uygulanarak yapılır.



Şekil 39. 1×1 konvolüsyon işleminin boyut düşürmek için kullanılması

3.6.2.2.4. Yoğun (Dense) blok içeren modeller

Yoğun bloklar içeren mimarilerde, her katman tüm önceki katmanları girdi olarak alır. Bu, her katmanın daha çeşitli özelliklere ve daha zengin desenlere sahip olmasını sağlar. DenseNet201 modeli, yoğun bloklar içeren derin bir CNN mimarisidir. Yoğun blokların yapısı, Şekil 40'da gösterilmiştir.



Şekil 40. Büyüme Oranı $k = 4$ ve 5 Katmandan Oluşan Yoğun Bir Blok (Huang vd., 2017) (Her katman tüm önceki özellik haritalarını girdi olarak alır)

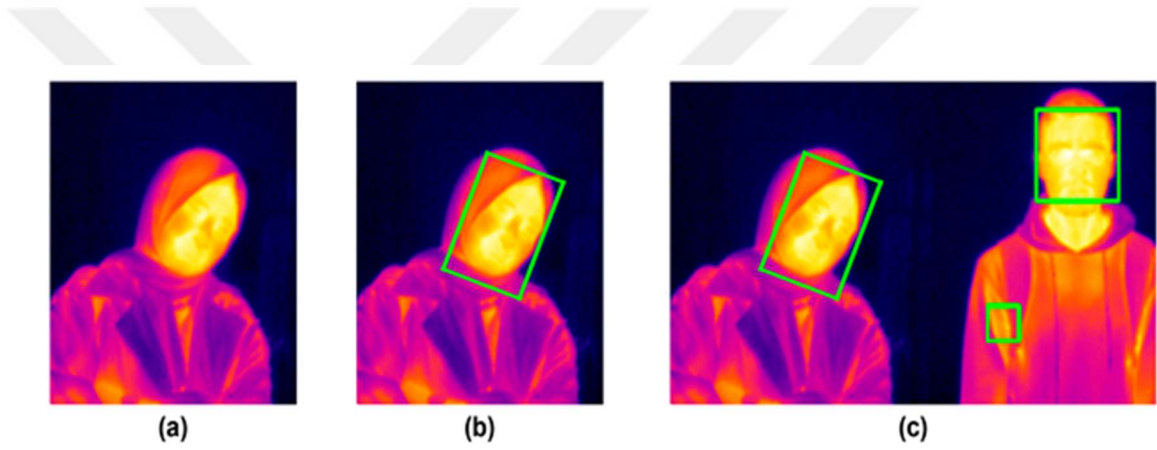
Fire modülü, küçük bir 1×1 konvolüsyon kullanarak giriş kanal sayısını azaltan bir sıkıştırma katmanı ve 1×1 ve 3×3 konvolüsyonlar kullanarak sıkıştırma katmanındaki kanal sayısını artıran bir genişletme katmanından oluşur. Fire bloklarının yapısı, Şekil 41'de gösterilmiştir. SqueezeNet modeli, fire bloklarını içeren CNN modellerine bir örnektir. Bu CNN modeli, 50 kat daha az parametre ile ImageNet'te AlexNet seviyesinde doğruluk elde etmektedir. Bu, daha az parametreye sahip bir mimariyle başarılı sınıflandırma sonuçları elde etmeyi amaçlayan bir yaklaşımdır.



Şekil 41. Ateşleme blok yapısı (SqueezeNet mimarisi örneği) (Huang vd., 2017)

3.6.2.2.5. Nesne tanımadada kullanılan CNN modelleri

CNN modelleri başlangıçta, bir görüntünün tamamının neyi temsil ettiğini tahmin etmeyi amaçlayan sınıflandırma problemlerinde kullanılmıştır. Ancak, görüntüler daha karmaşık hale geldikçe ve bir görüntüdeki nesne sayısı arttıkça, görüntüdeki nesnelerin konumları belirlenmiş ve her bir nesnenin hangi sınıfa ait olduğu tespit edilmeye çalışılmıştır. Görüntü sınıflandırmasında, görüntüye bir sınıf etiketi atanırken, nesne yerleştirmede, görüntüdeki bir veya birden fazla nesne etrafında bir sınırlayıcı kutu çizilir. Nesne tespiti ise her iki süreci birleştirir: Görüntüdeki her ilgili nesne etrafında bir sınırlayıcı kutu çizilir ve bunlara bir sınıf etiketi atanır. Bu tüm süreç, nesne tanıma olarak adlandırılır. Şekil 42, görüntü sınıflandırması, görüntü yerleştirmesi ve nesne tespitini örneklerle görselleştirir.



Şekil 42. (a) Görüntü sınıflandırma, (b) yüz lokalizasyonu ve (c) yüz tespiti

Gerçek hayattaki senaryolarda, bir görüntü birden fazla nesne içerebilir. Böyle durumlarda, görüntüdeki her nesnenin neyi temsil ettiğini tespit etmek ve tahmin etmek için farklı mimariler geliştirilmiştir. Çünkü bir CNN mimarisi, bir görüntüde sadece bir nesneyi tespit edebilir ve bu yöntem, farklı sınıflardan birden fazla nesne içeren görüntülerde işe yaramaz. Sonuç olarak, CNN modelleri, daha derin nesne tanıma mimarileri oluşturacak şekilde değiştirilmiştir.

3.6.2.3. CNN modellerinin eğitimi

Sinir ağlarında, belirli bir doğruluk seviyesine ulaşabilmek için, katmanlar arası bağlantılardaki ağırlıkların, istenen tolerans hata değerine ulaşana kadar eğitilmesi gerekir. Benzer şekilde, CNN modellerinin eğitimi sırasında, her bir özellik için

kullanılan filtrelerin içeriği, kabul edilebilir bir hata değeri ulaşana kadar güncellenir. Bir sinir ağında, sistem önce örneklerle eğitilir, ardından sistemin hata değeri belirlenmek için test edilir.

Derin ağların eğitiminde, eğitim veri setinden gelen görüntü verisi tasarlanan modele giriş olarak verilir. Görüntünün piksel değerleri, filtrelerin ağırlık değerleriyle çarpılır ve bu ürünlerin toplamı bir sonraki katmana iletilir (konvolüsyon işlemi). İleri geçişte, ağ tarafından üretilen çıktı ile hedef çıkış arasındaki fark hesaplanarak hata değeri elde edilir. Bu hata değeri daha sonra ağdaki tüm ağırlıklar boyunca geri yayılır. Bu süreç, eğitim veri setindeki tüm veriler için tekrarlanır ve model, ileri ve geri geçişler yapılarak birden çok iterasyon üzerinde eğitilir. Geri yayılım sürecinde kullanılan gradyan tabanlı optimizasyon algoritmaları büyük önem taşır. Bu tezde tasarlanan CNN mimarilerinin optimizasyonu için, Momentumlu Stokastik Gradyan İnişi (SGDM) kullanılmıştır. Ayrıca, Stokastik Gradyan İnişi (SGD) de bir optimizasyon algoritması olarak kullanılabilir.

CNN modellerinin tasarımı ve eğitimi sırasında kullanılan hiper parametreler, ağın performansını etkileyen diğer kritik parametrelerdir. Örneğin, görüntü giriş boyutu, CNN modelinin girişinde kullanılan özellik bilgisinin miktarını belirlerken, CNN modelindeki konvolüsyon filtrelerinin sayısı, konvolüsyon işlemi ile elde edilen özellik haritalarının sayısını belirler. Bu nedenle, derin öğrenme mimarilerinin uygun eğitim ve topoloji parametreleriyle eğitilmesi gereklidir.

3.6.2.3.1. Topoloji parametreleri

Bir CNN modelinin giriş boyutu, katman sayısı (derinlik), katmanlarda kullanılan filtre boyutu, filtre sayısı ve adım büyüklüğü, topoloji parametreleri arasında yer alır. Literatür incelendiğinde, farklı yapılarla birçok CNN modelinin bulunduğu görülmektedir. Model geliştirme aşamalarının başlarında, daha derin mimarilerin daha iyi sonuçlar vereceği görüşü desteklenmiştir. Ancak, sonraki yıllarda tasarlanan modellerde, ağı derinleştirmek yerine daha basit mimariler kullanılarak daha başarılı modeller geliştirilmesi amaçlanmıştır. Bu yapıların oluşturulabilmesi için, temel CNN mimarisi içinde farklı blok türleri geliştirilmiştir.

3.6.2.3.2. Eğitim parametreleri

CNN model eğitimi sırasında iterasyon sayısı, momentum katsayısı, batch boyutu, öğrenme hızı ve kullanılan optimizasyon yöntemi, eğitim parametreleri arasında yer alır:

- **İterasyon Sayısı:** Bu parametre, CNN modelinin eğitimi sırasında kaç kez ileri ve geri yayılma işleminin yapılacağını belirtir. Başka bir deyişle, CNN modelindeki ağırlıkların kaç kez güncelleneceğini ifade eder.
- **Momentum Katsayısı:** Bu parametre, CNN modelinde ağırlıkların optimal değere daha hızlı ulaşmasına yardımcı olmak için kullanılır. Genellikle 0.9 olarak ayarlanır.
- **Batch Boyutu:** CNN uygulamalarında, veri setleri genellikle büyük miktarda veri içerdiğinden, tüm verinin bir anda öğrenme işlemine alınması, işlem süresi ve bellek kullanımı açısından maliyetli olabilir. Her iterasyonda, geri yayılım işlemi kullanılarak ağda gradyan hesaplanır ve ağırlıklar güncellenir. Verinin miktarı büyük olduğunda, hesaplama zamanı artar hale gelir. Bunun için, veri seti küçük batch'lere bölünerek işlenir. Batch boyutu, CNN modelinin her iterasyonda kaç veri noktasını aynı anda işleyeceğini belirler. Uygun batch boyutu, eğitim setindeki öge sayısından (genellikle iki katlarının) daha küçük olmalı ve GPU belleğine sığmalıdır.
- **Öğrenme Hızı:** Parametre güncellemesi, geri yayılım yoluyla yapılır; burada yeni ağırlık, gradyanın ve öğrenme hızının çarpımının mevcut ağırlıktan çıkarılmasıyla hesaplanır. Öğrenme hızı, sabit bir değer olarak ayarlanabilir veya adım adım ayarlanabilir. Büyük bir öğrenme hızı seçmek, eğitim verilerinin aşırı etkisini doğurabilirken, küçük bir öğrenme hızı, modelin daha yavaş ilerlemesine ve eğitim süresinin artmasına neden olabilir.
- **Optimizasyon Algoritması:** Derin öğrenme uygulamalarında, öğrenme süreci bir optimizasyon problemi olup, optimizasyon yöntemleri, doğrusal olmayan problemler için en iyi çözümü bulmak amacıyla kullanılır. Optimizasyon algoritmaları, hız ve başarı açısından farklı sonuçlar verebilir. SGD ve SGDM, CNN modellerinin eğitiminde yaygın olarak kullanılan optimizasyon algoritmalarındandır.

3.6.3. Büyük Dil Modelleri (LLM) ile Termal Görüntülerde Yüz Tanıma ve Sıcaklık Analizi

Büyük Dil Modelleri (Large Language Models - LLM), genellikle büyük veri kümeleri üzerinde eğitilmiş derin öğrenme tabanlı yapay zeka modelleridir. Bu modeller, doğal dil işleme (NLP) yetenekleriyle metin verilerini analiz etme, anlamlandırma ve çıkarım yapma konularında oldukça başarılıdır. LLM'ler, genellikle GPT (Generative Pre-trained Transformer) veya BERT (Bidirectional Encoder Representations from Transformers) gibi modern mimarilerle geliştirilmiştir.

Bu modeller, termal görüntülerden elde edilen sıcaklık verilerinin analiz edilmesi ve anlamlandırılması gibi uygulamalarda kullanılarak yüz tespiti sonrası süreçlere katkı sağlar. Ancak, LLM'ler doğrudan yüz tespiti yapma kapasitesine sahip değildir ve diğer yöntemlerle birlikte çalışmaları gerekir. Bu kısımda Büyük Dil Modelleri ile nesne tanıma temeli ele alınmıştır.

Büyük Dil Modelleri (LLMs), dil ile etkileşim ve işleme yöntemlerimizi kökten değiştirerek, doğal dilin anlaşılması, üretilmesi ve iletişimde yeni olanaklar yaratmıştır. Bu modeller, sürekli bir evrim süreci içinde, dil işleme ve yapay zekâda ulaşılan sınırları genişletmeye devam etmektedir. Bu makalede, özellikle Transformer tabanlı LLM'lere odaklanılmış ve bu alandaki son gelişmelere dair bir genel bakış sunulmuştur.

Büyük Dil Modelleri (LLMs) için en büyük atılım, 2017 yılında Transformer teknolojisinin tanıtılmasıyla gerçekleşmiştir. Transformer, uzun vadeli dilsel bağları kavrama sanatında devrim niteliğinde bir yenilik sunmuştur. Bu teknoloji yalnızca teorik bir başarı olmamış, aynı zamanda birden fazla GPU üzerinde eşzamanlı eğitim imkânı sağlayarak çok daha büyük modellerin geliştirilmesinin yolunu açmıştır. Ardından, 2018 yılında OpenAI'nin GPT-1 modelinin tanıtılmasıyla doğal dil işleme (NLP) alanında önemli bir dönüm noktası yaşanmıştır. Transformer tabanlı mimarisiyle GPT-1, 117 milyon parametresi sayesinde yalnızca veri işlemekle kalmamış, aynı zamanda bağlamsal olarak tutarlı cümleler oluşturabilmiştir. GPT-1'in başlangıçtaki sınırlamalarına rağmen, bu model NLP görevlerini yeniden tanımlayan dönüştürücü

gücüyle yapay zekâ arařtırmalarında yeni bir dalganın temelini atmıř ve LLM alanında yoğun bir rekabeti tetiklemiřtir.

2020 yılında OpenAI, son derece tutarlı ve doęal görünümlü metinler üretebilen GPT-3 modelini tanıtmıřtır. GPT-3, büyük dil modellerinin çok çeřitli dilsel görevlerdeki olaęanüstü yeteneklerini sergileyerek önemli bir dönüm noktası olmuřtur. GPT-3'ün başarısının ardından OpenAI, GPT-4 modelini piyasaya sürmüřtür. Bu model, daha tutarlı ve doęal dil çıktıları üretebilme yeteneęini bir adım öteye tařımmıřtır. GPT-4'ün ardından, Google Bard ile rekabete katılmıř, Amazon Alexa'yı yenilikçi yapay zekâ özellikleriyle geliřtirmiş Huawei Pangu modelleriyle ilerleme kaydetmiş ve Alibaba QWEN modellerini tanıtmıřtır. Meta ise açık kaynaklı temel modellere odaklanan LLaMA modelini piyasaya sürmüřtür. LLaMA2, İnsan Geri Bildirimiyle Takviye Öğrenimi (RLHF) alanında daha ileri adımlar atmıř ve genellikle mevcut açık kaynaklı modelleri yardımseverlik ve güvenlik kriterlerinde geride bırakan LLaMA-chat adında sohbet odaklı bir versiyon geliřtirmiřtir. Arařtırmacılar, LLaMA modellerini ya talimat uyarlaması ya da sürekli ön eğitim yoluyla genişletmiřtir. Alpaca, LLaMA temel alınarak ince ayar yapılan ilk açık talimat izleme modeli olmuřtur. Ayrıca, Vicuna adlı başka bir popüler LLaMA türevi, ShareGPT'den toplanan kullanıcı konuşmaları üzerinde eğitilmiřtir.

LLM'lerin çeřitli sürümleri ortaya çıktııkça, ortak zorluklarla karşı karşıya kalmaktadırlar. Bu çalışmada, bu zorlukların birçoęunun nedensel yöntemlerle etkili bir şekilde ele alınabileceęi gösterilmektedir. Bu yöntemler, LLM'lerin akıl yürütme yeteneklerini geliřtirme, adaletle ilgili sorunları ele alma, olası önyargıları azaltma, güvenlięi sağlama, model çıktılarının açıklanabilirlięini artırma ve modelleri çok modlu versiyonlara genişletme gibi hedefleri kapsamaktadır.

Bu ilerlemelerin üzerine inşa edilen yeni bir ilgi alanı, bu modellerin görsel verileri de kapsayacak şekilde genişletilmesi olmuřtur. Bu eğilim, Büyük Görsel Dil Modelleri'nin (LVLM'ler) ortaya çıkıřına neden olmuřtur. Bu modeller, hem metinsel hem de görsel bilgilerin entegrasyonunu amaçlayarak, multimodal içerikleri yorumlayabilen ve üretebilen daha kapsamlı yapay zekâ sistemleri için yeni olanaklar sunmaktadır.

En yaygın yaklaşımlardan biri, görsel özellikleri LLM'lere ek girdi olarak ekleyip bunları metinsel özelliklerle hizalamaktır. Bu yöntem, MiniGPT-4, LLaVA, Mplug-Owl ve diğer büyük görsel-dil modellerinde (LVLM'ler) uygulanmıştır. Bu araştırma, yukarıda belirtilen mevcut LVLM'lerde karşılaşılan zorlukların, nedensel yöntemler aracılığıyla etkili bir şekilde ele alınabileceğini de göstermektedir.



4. TERMAL GÖRÜNTÜLERDE ATEŞ TESPİTİNDE KLASİK/MODERN METOTLAR VE GENİŞ DİL MODELİ YAKLAŞIMI

Bu tez çalışmasında, termal görüntü kullanılarak COVID-19 ve benzeri ateşli hastalıkların tespitine katkı amacıyla farklı yöntemler karşılaştırılmıştır. Haar Cascade (HC), Derin Öğrenme (Deep Learning) ve Büyük Dil Modelli (LLM) olarak ChatGPT ile gerçekleştirilen analizler, yöntemlerin termal görüntüleme, yüz tanıma ve veri analizi alanındaki performanslarını değerlendirmiştir. Bu bölümde deneysel yöntem ve elde edilen bulgular ele alınmıştır.

4.1. Haar Cascade Yaklaşımı

4.1.1. Çalışmanın özeti

Covid-19 virüsü, küresel çapta büyük etkiler yaratan, birçok insanın hayatını kaybetmesine neden olan ve zamanında teşhis edilmediği takdirde daha fazla kişi için risk oluşturan ciddi bir tehdittir. Bu virüsün yayılma riskini azaltmanın ve kontrol altına almanın ilk adımı erken teşhistir. Her ne kadar tüm semptomlar evrensel olmasa da, yüksek ateş teşhis için en kritik ortak göstergedir. Pandeminin zirve yaptığı dönemde, halkın yoğun olduğu alanlarda temassız termometreler ve termal görüntüleme sistemleri yüksek sıcaklıkları ölçmek için kullanılmıştır. Termal kameralar kamusal alanlarda kısmen etkili olsa da, görüntü kalitesi ve kamera yetenekleri nedeniyle kullanımları sınırlı kalmıştır. Ana sorun, ısı haritalarının çakışması ve tespit edilen ısının kalabalık gruplarda kime ait olduğunun belirlenememesiydi.

Bu zorluğun üstesinden gelmek ve termal haritalarda bireylerin ayrımını iyileştirerek yüksek sıcaklıkların daha doğru şekilde tespit edilmesini sağlamak amacıyla, derin öğrenme ve yapay zeka teknikleri kullanılarak kapsamlı performans analizleri gerçekleştirilmiştir.

Deneysel sonuçlar, önerilen yaklaşımın tespit ve sınıflandırma süreçlerinde ağ eğitimini geliştirdiğini ve üstün performans sağladığını göstermektedir. Bu gelişmiş

yapay zeka arayüzü, görüntüleme uzmanları için bilgisayar destekli bir karar destek sistemi olarak da kullanılabilir.

4.1.2. Önerilen Haar Cascade modeli

Haar-cascade sınıflandırıcı algoritması, dijital bir görüntü ya da video karesi içinde belirli nesneleri tespit etmek amacıyla kullanılır. Özellikle gri tonlamalı görüntülerde yüz algılama için yaygın olarak kullanılan bu algoritma (Viola & Jones, 2001), bir bölgenin yüz içerip içermediğini belirlemek için pozitif ve negatif veri setlerinden yararlanır. Hızlı olması ve düşük hesaplama maliyeti nedeniyle, termal görüntüler gibi düşük çözünürlüklü verilerde sıklıkla tercih edilmektedir (Gonzalez & Woods, 2008). Yapılan çalışmalar, Haar Cascade algoritmasının -30° ile $+30^\circ$, -60° ile $+60^\circ$ ve -60° ile $+75^\circ$ açılarında bireyleri başarıyla tespit edebildiğini ve dikey olarak dönen ya da eğik yüzleri dahi tanıyabildiğini göstermektedir (Arya & Tiwari, 2020).

Bu çalışmada, yüz algılama için hibrit bir yöntem uygulanmıştır. RGB görüntülerde algılanan yüzlerin koordinatları, IR (termal) görüntülerine eşleştirilmiştir. Bu yaklaşım, termal görüntülerde yüz algılama performansı RGB görüntülere kıyasla daha düşük olduğu için gereklidir. RGB görüntülerdeki belirgin yüz bölgeleri, algılama doğruluğunu artırmış ve IR görüntülerinde sıcaklık analizi yapılacak bölgelerin işaretlenmesini sağlamıştır.

Tablo 7. Haar Cascade Mimarisi ve Katmanları

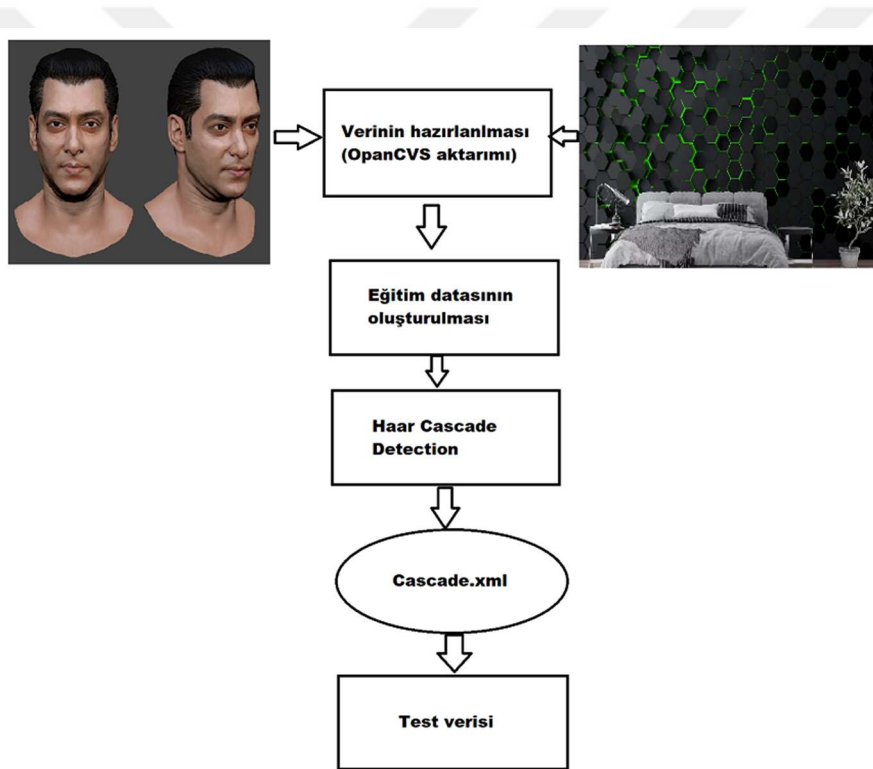
Katman	Çıkış Boyutu
1. Konvolüsyonel Katman (Conv1)	$(W-2) \times (H-2) \times F1$
2. Max-Pooling Katmanı (Max Pooling)	$(W/2) \times (H/2) \times F1$
3. Konvolüsyonel Katman (Conv2)	$(W-4) \times (H-4) \times F2$
4. Max-Pooling Katmanı (Max Pooling)	$(W/4) \times (H/4) \times F2$
5. Konvolüsyonel Katman (Conv3)	$(W-6) \times (H-6) \times F3$
6. Max-Pooling Katmanı (Max Pooling)	$(W/8) \times (H/8) \times F3$
7. Tam Bağlantılı Katman (Fully Connected)	$(1) \times (1) \times N$ (N: sınıfların sayısı)
8. Çıkış Katmanı (Output Layer)	$(1) \times (1) \times C$ (C: sınıf sayısı)

W: Weight, H: Height, F1, F2, F3, her konvolüsyonel katmandaki filtre sayısı,

Haar Cascade algoritması, OpenCV kütüphanesindeki önceden eğitilmiş bir modelle (örn. haarcascade_frontalface_default.xml) çalışır. OpenCV'de optimize

edilmiş görüntüler karelere bölünür ve Haar kütüphanesi kullanılarak yüz bölgeleri izole edilir.

Önerilen model, yüz ifadelerini içeren dikdörtgen bölgelerdeki piksel değerlerini hızlı ve işlevsel bir şekilde eğitmek ve her piksele düşen ısıyı belirleyerek en yüksek sıcaklığın tespiti için tasarlanmıştır. Hızlı sorgulama için derin bir yapay sinir ağı modeli oluşturulmuştur. Mimari, üç konvolüsyon katmanı, üç maks havuzlama katmanı ve bir tam bağlantılı katmandan oluşmaktadır. Konvolüsyon matris boyutu 5x5 olarak belirlenmiş, adımı 2 olan 5x5 maks havuzlama çekirdeği kullanılmıştır. Son olarak, Softmax sınıflandırıcı veriyi işleyerek 20 farklı yüzü temsil eder. Aşağıda şekil diyagramı verilmiştir.



Şekil 43. Haar eğitim süreci aşağıdaki

Şekil 43. Haar eğitim süreci aşağıdaki adımları içermektedir:

1. Veri Hazırlama

- Pozitif veri seti: Hedef nesneyi (örn. yüzleri) içeren görüntüler.
- Negatif veri seti: Hedef nesneyi içermeyen (örn. arka plan) görüntüler.
- Genellikle yaklaşık 20 pozitif ve 20 negatif görüntü kullanılır.

2. Pozitif Örneklerin İşlenmesi

- Pozitif görüntüler yeniden boyutlandırılır ve koordinatlar etiketlenir.
- 3. **Negatif Örneklerin İşlenmesi**
 - Negatif görüntüler nesne etiketleri olmadan formatlanır.
- 4. **Veri Seti Hazırlama**
 - Pozitif ve negatif görüntülerden bir eğitim veri seti oluşturulur.
- 5. **Haar Özelliklerinin Hesaplanması**
 - Örneklerden Haar özellikleri çıkarılır.
- 6. **Adaboost ile Eğitim**
 - Adaboost algoritması kullanılarak güçlü sınıflandırıcılar üretilir.
- 7. **Cascade Eğitimi**
 - Adaboost'tan elde edilen sınıflandırıcılar, doğruluğu artırmak için bir kaskad yapısında düzenlenir.
- 8. **Model Testi ve Optimizasyonu**
 - Test veri seti kullanılarak doğruluk ve hata oranları değerlendirilir.
- 9. **Modelin Dağıtımı**
 - Eğitim tamamlandıktan sonra model, canlı kamera görüntülerinde veya video akışlarında gerçek zamanlı nesne algılama için kullanılır.

Tablo 8. Önerilen Model Katmanları ile İlgili Uygulama Adımları Akış Şeması

Adım Sırası	Uygulama Adımı	Açıklama
1	Veri Hazırlama	Pozitif (nesneli) ve negatif (nesnesiz) veri setleri topla.
2	Pozitif Örneklerin İşlenmesi	Pozitif görüntüleri yeniden boyutlandır ve koordinatları etiketle.
3	Negatif Örneklerin İşlenmesi	Negatif görüntüleri etiketsiz olarak formatla.
4	Veri Seti Hazırlama	Pozitif ve negatif örnekleri birleştirerek eğitim seti oluştur.
5	Haar Özelliklerinin Hesaplanması	Örneklerden özellikler çıkar.
6	Adaboost ile Eğitim	Güçlü sınıflandırıcılar oluştur.
7	Cascade Eğitimi	Sınıflandırıcıları kaskad yapısında düzenle.
8	Model Testi ve Optimizasyonu	Test seti ile doğruluğu ve hata oranlarını değerlendir.
9	Modelin Dağıtımı	Eğitilen modeli gerçek zamanlı algılama için kullan.

4.1.2.1. Modelin Yüklmesi

Haar Cascade algoritması, OpenCV kütüphanesinden alınan önceden eğitilmiş bir model (haarcascade_frontalface_default.xml) ile çalıştırılmıştır. Bu model, yüz özelliklerini tanımlayabilmek için optimize edilmiştir.

4.1.2.2. Görüntünün Hazırlanması

RGB görüntüler ve termal görüntüler, algoritmanın çalışabilmesi için gri tonlamalı forma dönüştürülmüştür. Gri tonlama, Haar Cascade algoritmasının temel gereksinimlerinden biridir. RGB görüntülerde tespit edilen yüzler, aynı çözünürlükteki termal görüntülere işaretlenmiştir. Böylece, termal görüntülerde doğrudan yüz tespiti yapmanın sınırlamaları aşılmıştır.

4.1.2.3. Parametre Ayarları ve Yüz Tespiti

Algoritmanın başarımı, **scaleFactor** ve **minNeighbors** parametrelerine bağlıdır:

- **scaleFactor:** Farklı boyutlardaki yüzleri tespit etmek için kullanılan ölçeklendirme faktörü. Bu çalışmada, scaleFactor değeri 1.05 olarak belirlenmiştir.
- **minNeighbors:** Bir alanın yüz olarak kabul edilmesi için gerekli olan komşu bölgelerin sayısı. Çalışmada bu parametre 25 olarak ayarlanmıştır.

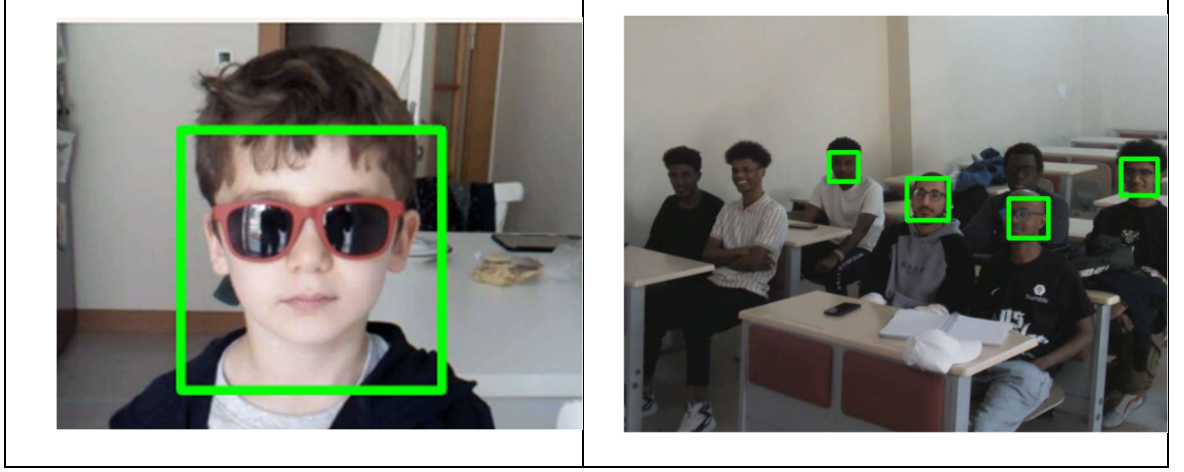
RGB görüntülerde yapılan tespit, termal görüntülerdeki yüz koordinatlarının belirlenmesinde referans alınmıştır.

4.1.2.4. Tespit Edilen Yüzlerde Sıcaklık Analizi

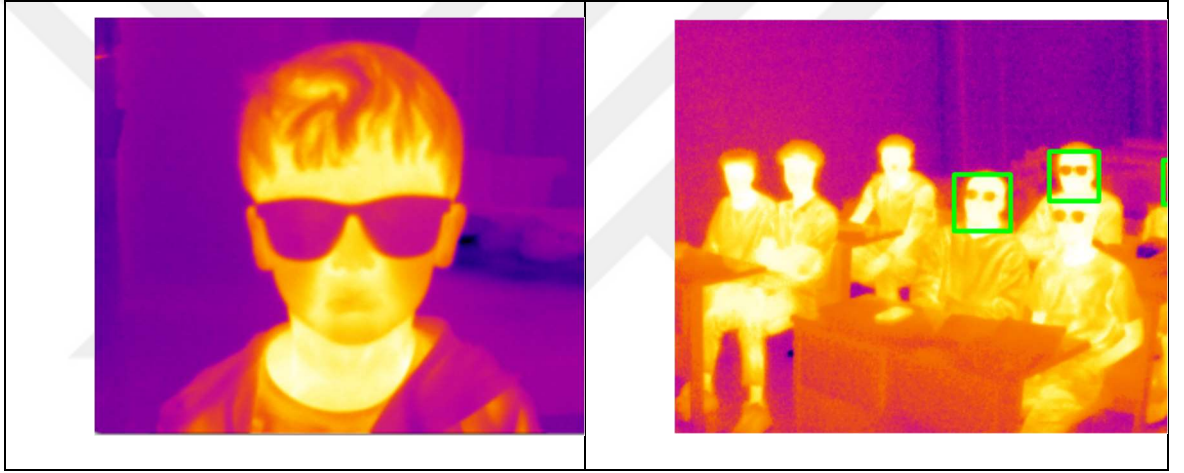
Tespit edilen yüz bölgelerinde, maksimum sıcaklık değerleri analiz edilmiştir. Termal görüntülerdeki sıcaklıklar, Fahrenheit'tan Celsius birimine dönüştürülerek görüntü üzerine metin olarak işlenmiştir.

4.1.2.5. Görselleştirme

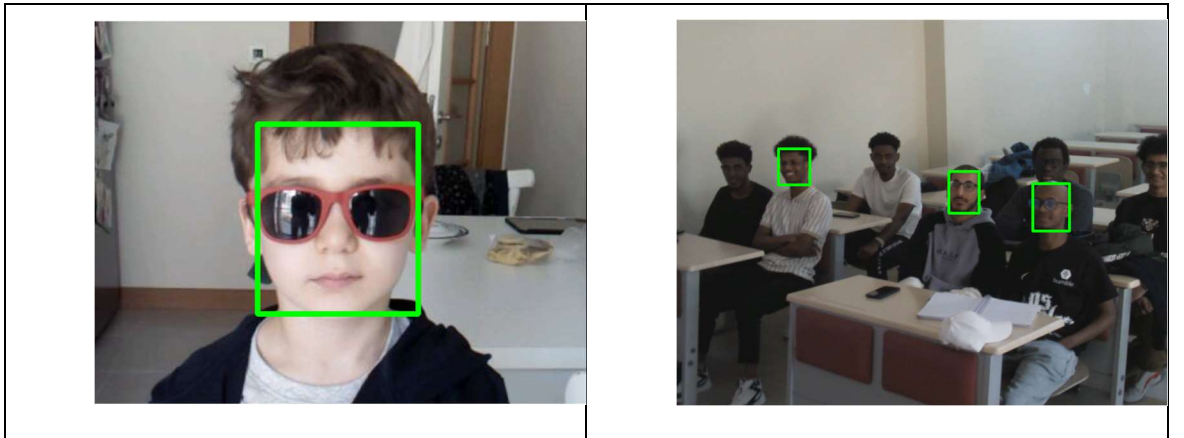
Algoritmanın tespit sonuçları, yüzlerin etrafına çizilen dikdörtgenlerle ve sıcaklık değerlerinin görüntü üzerine yazılmasıyla görselleştirilmiştir. Aşağıda, her bir yöntemin RGB ve IR görüntülerdeki performansını görselleştirmek için örnekler sunulmuştur:



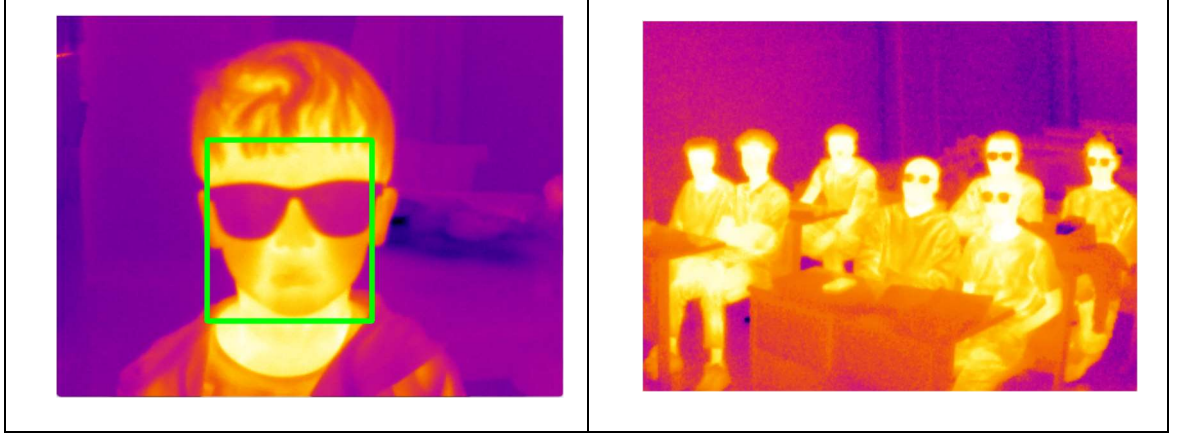
Şekil 44. Haar Cascade yöntemi ile RGB görüntülerde tespit edilen yüz bölgeleri.
scaleFactor, minNeighbors = 1.05, 25



Şekil 45. Haar Cascade yöntemi ile IR görüntülerde tespit edilen yüz bölgeleri.
scaleFactor, minNeighbors = 1.05, 25



Şekil 46. Derin öğrenme yöntemi ile RGB görüntülerde tespit edilen yüz bölgeleri.



Şekil 47. Derin öğrenme yöntemi ile IR görüntülerde tespit edilen yüz bölgeleri.

Bu görseller, RGB ve IR görüntülerde kullanılan yöntemlerin performans farklarını net bir şekilde göstermektedir. Şekilde görüldüğü gibi yapılan deneysel çalışmaya göre tekli termal görüntülerde yüz tanınması sağlanırken çoklu görüntülerde yüz tanınması sağlanamamıştır.

4.1.3.6. Deneysel Çalışmalar ve Sonuçlar

Haar Cascade algoritması, farklı parametre kombinasyonları ile test edilmiştir. Her parametrede elde edilen sonuçlar aşağıda açıklanmış ve görsellerle desteklenmiştir.

4.1.3.6.1. scaleFactor = 1.01 ve minNeighbors = 1

Aşağıda termal görüntülerde scaleFactor = 1.01 ve minNeighbors = 1 ile elde edilen sonuçlar verilmiştir.



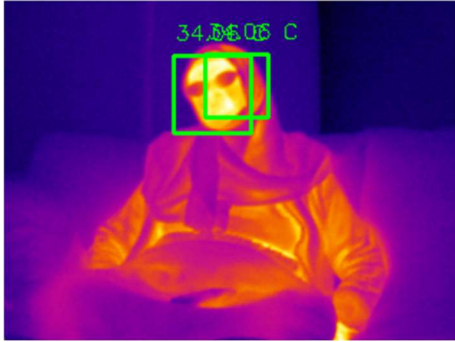
Şekil 48. Haar Cascade çoklu yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)

Bu parametre kombinasyonu, çok hassas bir tarama yapmış ancak yüz olmayan bölgelerde yüksek oranda yanlış pozitif sonuçlar üretmiştir.



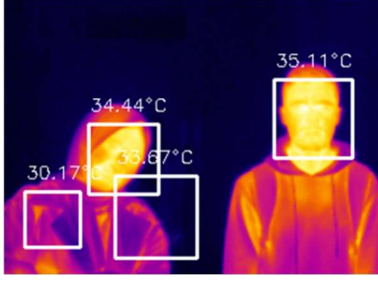
Şekil 49. Haar Cascade tekli yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)

Şekil 49’da görüldüğü gibi bu parametre kombinasyonu ile elde edilen sonuçlarda tekli görüntülerde yüz tanınmasında başarılı olmuş ancak yine de yanlış pozitif sonuçlar üretmiştir.



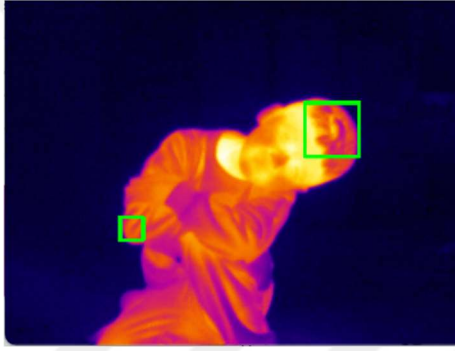
Şekil 50. Haar Cascade yatay bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)

Şekil 50’de görüldüğü gibi elde edilen sonuçlarda yatay bakan tekli görüntülerde yüz tanınmasında yine başarılı olmuş, yine de şekil 49’da olduğu gibi yine de yanlış pozitif sonuçlar üretmiştir.



Şekil 51. Haar Cascade ile yatay ve dikey bakan insan yüzü tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)

Şekil 51 incelendiğinde bu parametre kombinasyonu, çok hassas bir tarama yapmış ancak yüz olmayan bölgelerde yanlış pozitif sonuçlar üretmiştir.



Şekil 52. Haar Cascade yana bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.01 ve minNeighbors = 1)

Şekil 52’de görüldüğü gibi elde edilen sonuçlarda yatay bakan tekli görüntülerde yüz tanınmasında yine başarılı olmuş, yine de şekil 49’da ve şekil 50’de olduğu gibi yine de yanlış pozitif sonuçlar üretmiştir.

4.1.3.6.2. scaleFactor = 1.05, minNeighbors = 5

Aşağıda termal görüntülerde scaleFactor = 1.05 ve minNeighbors = 5 ile elde edilen sonuçlar verilmiştir.



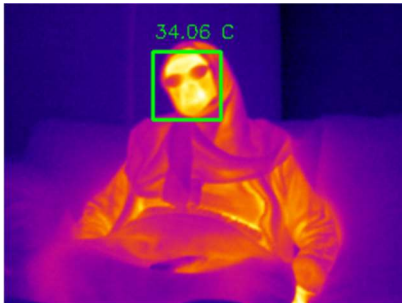
Şekil 53. Haar Cascade çoklu yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)

Şekil 53’de görüldüğü gibi parametre değişimi ile dengeli sonuçlar elde edilmiş, yanlış pozitif oranı azalmış ve doğru tespit oranı yüksek kalmıştır.



Şekil 54. Haar Cascade ile yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)

Şekil 54’de görüldüğü gibi parametre değişimi ile dengeli sonuçlar elde edilmiş, tekli düz çekimlerde yanlış pozitif görülmemiş ve doğru tespit oranı yüksek kalmıştır.



Şekil 55. Haar Cascade yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)

Şekil 55’de görüldüğü gibi parametre değişimi ile dengeli sonuçlar elde edilmiş, tekli yan bakan çekimlerde yanlış pozitif görülmemiş ve doğru tespit oranı yüksek kalmıştır.



Şekil 56. Haar Cascade ile yatay ve dikey bakan insan yüzü tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)

Şekil 56’da görülen çoklu çekimde Şekil 53’ün aksine dengeli sonuçlar elde edilememiş yanlış pozitif oranı yüksek kalmış ve doğru tespit oranı düşmüştür.



Şekil 57. Haar Cascade ile yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 5)

Şekil 57’de görüldüğü gibi tekli yan bakan çekimlerde Şekil 55’in aksine parametre değişimi ile dengeli sonuçlar elde edilememiş yanlış pozitif görülmüştür. Tespit oranı ise çok düşüktür.

4.1.3.6.3. scaleFactor = 1.05, minNeighbors = 25

Aşağıda termal görüntülerde scaleFactor = 1.05, minNeighbors = 25 ile elde edilen sonuçlar verilmiştir.



Şekil 58. Haar Cascade çoklu yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)

Şekil 58’de görüldüğü gibi çoklu çekimlerde yanlış pozitif oranı önemli ölçüde azalmış ancak küçük yüzlerin tespiti mümkün olmamıştır. Tespit oranı çok düşüktür.



Şekil 59. Haar Cascade ile yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)

Şekil 59’da görüldüğü gibi tekli çekimlerde yanlış pozitif oranı önemli ölçüde azalmış ancak yüz tespiti mümkün olmamıştır. Tespit oranı çok düşüktür.



Şekil 60. Haar Cascade ile hafif yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)

Şekil 60’da görüldüğü gibi tekli yan çekimlerde yanlış pozitif oranı önemli ölçüde azalmış yüz tespiti yapılmıştır. Tespit oranı çok yüksektir.



Şekil 61. Haar Cascade ile yatay ve dikey bakan insan yüzü tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)

Şekil 61’de görüldüğü gibi çoklu çekimlerde yanlış pozitif oranı azalmış ancak yan yüzlerin tespiti mümkün olmamıştır. Tespit oranı çok düşüktür.

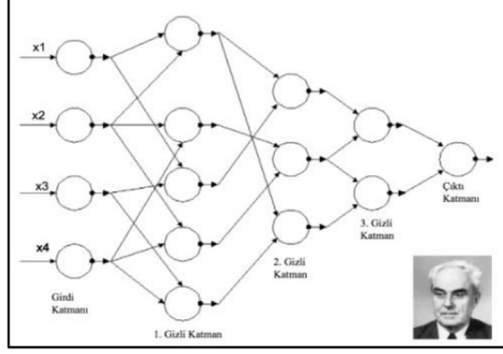


Şekil 62. Haar Cascade ile yan bakan yüz tespiti ve sıcaklık analizi (scaleFactor = 1.05 ve minNeighbors = 25)

Şekil 62’de görüldüğü gibi tekli yan çekimlerde yanlış pozitif oranı azalmış ancak yüz tespiti mümkün olmamıştır. Tespit oranı çok düşüktür.

4.2. Derin Öğrenme ve Konvolüsyonel Sinir Ağları Tabanlı Derin Öğrenme Modeli

Denetimli derin beslemeli çok katmanlı perceptronlar için ilk genel, öğrenme algoritması Ivakhnenko ve Lapa tarafından 1965 yılında yayınlanmıştır (Ivakhnenko ve Lapa, 1966).



Şekil 63. Ivankhnenko A. tarafından eğitilen bilinen ilk derin ağ mimarisi (Şeker vd., 2017).

Bu çalışmada, her katmanda, en iyi özellikler istatistiksel yöntemlerle seçilip bir sonraki katmana iletilmektedir. Ağlarını uçtan uca eğitmek için geri yayılımı (backpropagation) kullanılmamış, önceki katmanlardan sonraki katmanlara en küçük kareler yöntemi kullanılmıştır. Iankhnenko'dan sonra ilk derin öğrenme mimarisi "Neokognitron", 1979 yılında Fukushima tarafından önerilmiştir. Omurgalı canlıların görsel sinir sistemlerinden esinlenilerek ortaya atılan yapıda, "denetimsiz öğrenme" ile kendi kendini organize eden bir ağ geliştirilmiştir. Fukushima'nın ağları modern ağlara benzer çoklu bükülme ve havuz katmanların içermektedir. Derin mimarilerin çoklu katmanlarında hataların geri yayılımı öğrenmenin en önemli eksikliğidir. Geri yayılım algoritmaları önceki yıllarda ortaya atılmış olsa da ilk başarılı derin sinir ağı uygulamasını Yann LeCun vd. posta kutusu yazıları üzerinde geliştirmişlerdir (Savaş vd., 2016). Her ne kadar ağ başarılı çalışsa da, eğitiminin yaklaşık olarak 3 gün sürdüğünden pratikte uygun olmadığı görülmüştür.

CNN algoritmaları görüntü ve ses işleme alanı başta olmak üzere doğal dil işleme (NLP), biyomedikal gibi birçok farklı alanda uygulanmaktadır. Özellikle görüntü işleme alanında görülmüş en iyi (state of the art) sonuçlar elde edilmiştir.

Bu çalışma kullanılan Deep Learning modeli temel olarak 2 aşamadan oluşmaktadır. İlki görüntünün yüz görüntüsü olup olmadığını algılama aşaması ve ikincisi de işlenen görüntü üzerinde yüz bölgesindeki en yüksek ısı değerinin bulunduğu bölgelerin tespiti aşamasıdır.

Yüz tespiti için elde edilen dijital görüntüler üzerinde, OpenCV kütüphanesindeki metotlar kullanılarak olası tüm yüze benzeyen bölgeleri bir nesne olarak, aynı zamanda yüz bölgesinde yer alan sıcaklık değerlerinin en yüksek noktasının tespit ve sınıflandırmak için sınıflandırıcıya gönderilmektedir. Sınıflandırmada diğer nesnelerin yüz olarak algılanmaması için yüz tespitinde ilk olarak geometrik tabanlı Haar ardışık sınıflandırıcılar kullanılmakta ve yüz bölgeleri bulunmaktadır. Sıcaklık tespiti için ise termal görüntülerin piksel değerleri üzerinden en yüksek değeri bulunmaktadır. Bu işlem için Haar ardışık sınıflandırıcıda, Viola ve Jones integral görüntü denilen bir teknik kullanılmaktadır ve Denklem (6)'de integrasyon formülü verilmektedir (Viola ve Jones, 2001).

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (6)$$

Burada $ii(x,y)$ integrasyonu alınmış görüntüyü, $i(x,y)$ orijinal görüntüyü temsil etmektedir.

Haar ardışık sınıflandırıcıda önemli olan piksel değerleridir. Her bir pikselin tamamlayıcı değeri yukarı ve solundaki tüm piksellerin toplamıdır. Sol üstten başlayarak, sağa ve aşağı yönde hareket ederek tüm görüntü taranmaktadır.

Sınıflandırma işlemlerinden sonra önerilen çalışmada mobil modülden gönderilen görüntüler servlet'in POST fonksiyonu ile alınıp, sunucuda kaydedilmiştir. “detection class” isimli sınıfa görüntü gönderilmiştir. Burada gerekli olan xml verilerek yüzler tespit edilmektedir.

Yüz ve göz tespiti aşamalarında, “haarcascade_frontalface_alt” algoritması kullanılmaktadır. haarcascade_frontalface_alt algoritması; hassas ve esnek bir algoritmadır.

4.2.1. Çalışmanın özeti

Bu çalışmada, önceden eğitilmiş bir Convolutional Neural Network (CNN) modeli kullanılarak termal görüntülerde yüz tespiti gerçekleştirilmiştir. Model, transfer öğrenimi yöntemiyle özelleştirilmiş ve aşağıdaki adımlarla uygulanmıştır:

1. Model Seçimi:

- Bu model, düşük hesaplama gücü gereksinimiyle termal görüntüler üzerinde etkili sonuçlar verebilir.

2. Veri İşleme:

- Termal görüntüler, CNN’in giriş gereksinimlerine uygun olarak yeniden boyutlandırılmış ve normalize edilmiştir.
- Örneğin, termal görüntüler 240x320 piksel çözünürlüğe ölçeklendirilmiştir.

3. Model Eğitimi ve Testi:

- Model, önceden eğitilmiş ağırlıklarla yüklenmiş ve termal veri kümesi üzerinde ince ayar yapılmıştır.
- Eğitilmiş model, test veri kümesinde değerlendirilmiştir.

4.2.2. Önerilen Derin Öğrenme Modeli

4.2.2.1. Modelin Yüklenmesi

Bu çalışmada, derin öğrenme tabanlı yüz tespiti için **OpenCV’nin dnn modülü** kullanılmıştır. Model, önceden eğitilmiş bir **Caffe Modeli** (*res10_300x300_ssd_iter_140000.caffemodel*) ile çalıştırılmıştır. Modelin yapısı, **Caffe Prototxt** dosyası (deploy.prototxt.txt) aracılığıyla tanımlanmıştır. Model, aşağıdaki adımlarla yüklenmiş ve kullanılmıştır:

4.2.2.2. Görüntü İşleme

Termal görüntüler, giriş boyutuna yeniden boyutlandırılmış ve normalize edilmiştir:

- `import cv2`
- `image = cv2.imread("thermal_image.png")`
- `image_resized = cv2.resize(img, (int(s*img.shape[1]), int(s*img.shape[0])), 0)`
- `image_normalized = image_resized / 255.0`

4.2.2.3. Yüz Tespiti ve Sıcaklık Analizi

Model, tespit edilen yüz bölgelerini ve sıcaklık analizlerini gerçekleştirmek için uygulanmıştır:

- `detections = model.predict(image_normalized.reshape(1, 224, 224, 3))`
- `# Tespit edilen yüz bölgelerindeki maksimum sıcaklık değerlerini analiz et.`

4.2.2.4. Deneysel Çalışmalar ve Sonuçlar

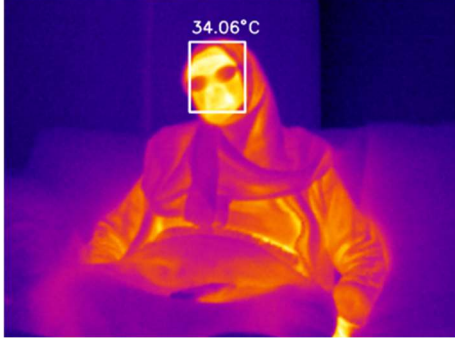
DL modeli, termal görüntülerde yüz tespiti sırasında yüksek doğruluk oranı sağlamıştır. Düşük kontrastlı ve yan dönmüş yüzlerde bile başarılı sonuçlar elde edilmiştir.

4.2.1.4.1. Görsel Sonuçlar



Şekil 64. Deep Learning yana bakan yüz tespiti ve sıcaklık analizi

Şekil 64’de görüldüğü gibi Deep Learning modelinde tekli yan çekimlerde yanlış pozitif oranı az ancak yüz tespiti mümkün olmamıştır. Tespit oranı çok düşüktür.



Şekil 65. Deep Learning hafif yana bakan yüz tespiti ve sıcaklık analizi

Şekil 65’de görüldüğü gibi Deep Learning modelinde tekli çekimlerde yanlış pozitif oranı az, yüz tespit oranı çok yüksektir. Isı düzeyi tespitinde ise gerçeğe yakın değerler elde edilmiştir (34,06°C).



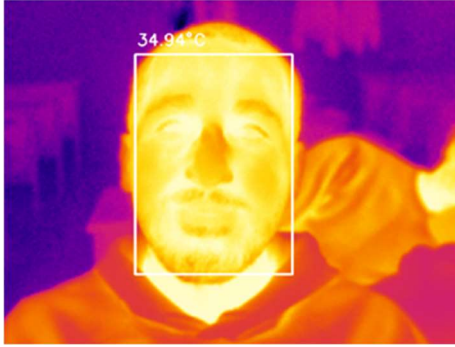
Şekil 66. Deep Learning hafif gülen yüz tespiti ve sıcaklık analizi

Şekil 66’da görüldüğü gibi Deep Learning modelinde tekli çekimlerde yanlış pozitif oranı az, yüz tespit oranı çok yüksektir. Isı düzeyi tespitinde ise gerçeğe yakın değerler elde edilmiştir (35,61°C).



Şekil 67. Deep Learning yüz tespiti ve sıcaklık analizi

Şekil 67’de görüldüğü gibi Deep Learning modelinde tekli çekimlerde yanlış pozitif oranı az, yüz tespit oranı çok yüksektir. Isı düzeyi tespitinde ise gerçeğe yakın değerler elde edilmiştir (35,83°C).



Şekil 68. Deep Learning yüz tespiti ve sıcaklık analizi

Şekil 68’de görüldüğü gibi Deep Learning modelinde tekli düz çekimlerde yanlış pozitif oranı az, yüz tespit oranı çok yüksektir. Isı düzeyi tespitinde ise gerçeğe yakın değerler elde edilmiştir (34,94°C).



Şekil 69. Deep Learning çoklu yüz tespiti ve sıcaklık analizi

Şekil 69 incelendiğinde ise görüldüğü gibi Deep Learning modelinde çoklu çekimlerde yanlış pozitif oranı oldukça yüksek yüz tespit oranı düşüktür. Isı düzeyleri ise gerçek değerden uzaklaşmıştır (33,39-33,78°C arası).



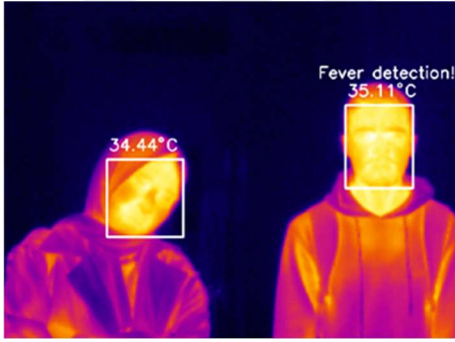
Şekil 70. Deep Learning hafif yana bakan ve ileri bakan yüz tespiti ve sıcaklık analizi

Şekil 70'de görüldüğü gibi Deep Learning modelinde düz ve yatay çoklu çekimlerde yanlış pozitif oranı çok az, yüz tespit oranı çok yüksektir. Isı düzeyi tespitinde ise gerçeğe yakın değerler elde edilmiştir.



Şekil 71. Deep Learning yana bakan yüz tespiti ve sıcaklık analizi

Şekil 71’de görüldüğü gibi Deep Learning modelinde tekli yatay çekimlerde yanlış pozitif oranı az, yüz tespit oranı çok yüksektir. Isı düzeyi tespitinde ise gerçeğe yakın değerler elde edilmiştir (35,72°C).



Şekil 72. Deep Learning hafif yana ve ileri bakan yüz tespiti ve sıcaklık analizi

Şekil 72’de görüldüğü gibi Deep Learning modelinde düz ve yatay çoklu çekimlerde yanlış pozitif oranı çok az, yüz tespit oranı çok yüksektir. Isı düzeyi tespitinde ise gerçeğe yakın değerler elde edilmiştir.

4.3. Büyük Dil Modeli (LLM)

4.3.1. Çalışmanın özeti

LLM'lerin (Büyük Dil Modelleri) üretici kapasitesinin potansiyel bir diğer kullanım alanı, küçük dil modelleri için veri artırma amacıyla karşı-olgusal

(counterfactual) metinler oluşturmaktır. Bir metin x ve bir kara kutu sınıflandırıcı B verildiğinde, metin x 'in karşı-olgusal metni $\sim x$, şu gereklilikleri sağlamalıdır:

1. $\sim x$ 'in sınıfı x 'ten farklı olmalıdır, $B(x) \neq B(\sim x)$.
2. x ve $\sim x$ yalnızca minimal sözcüksel değişikliklerle farklılık göstermelidir.
3. $\sim x$ uygulanabilir bir metin olmalı ve sağduyu (commonsense) kısıtlamalarını karşılamalıdır.

Bu bölümde, LLM'lerin karşı-olgusal metinler üretmedeki performansı ve bu metinlerin kalitesini artırma çabaları kısaca ele alınmaktadır:

1. **Duygu analizi (SA):** Duygu kutuplaşmasını değiştirmek.
2. **Doğal dil çıkarımı (NLI):** Bir öncül cümle ve bir hipotez cümlesi verildiğinde, hipotez cümlesinde değişiklik yaparak öncül ile olan ilişkisini değiştirmek.
3. **Adlandırılmış varlık tanıma (NER):** Cümledeki varlıkları, aynı türdeki başka varlıklarla değiştirmek.
4. **İlişki çıkarımı (RE):** Baş ve son varlık arasındaki ilişkiyi değiştirmek.

Çalışmada, görüntü analizi ve termal görüntülerin çıkarımı gibi basit görevlerde, LLM'ler tarafından üretilen karşı-olgusal verilerin, olası yanıltıcı ilişkileri azaltabileceği gösterilmiştir. Ancak, yoğun termal görüntülerde nesne ilişki çıkarımı gibi daha karmaşık görevlerde, LLM'ler düşük kaliteli karşı-olgusal belirlemeler üretebilmektedir. Yüksek kaliteli karşı-olgusal metinler oluşturmak için ayrıntılı yönergeler gereklidir. Bununla birlikte, beklentilerin aksine, "düşünce zinciri" yöntemi her zaman faydalı olmamakta; bazı durumlarda performansta ciddi düşüşlere neden olabilmektedir.

Değerlendirmelerden görüldüğü üzere, LLM'ler karşı-olgusal üreticiler olarak küçük dil modellerinin görüntü analizi çıkarımı gibi basit görevlerde performansını artırabilirken, üretilen karşı-olgusallar ilişki çıkarımı gibi karmaşık görevlerde önemli bir etki yaratamamaktadır. Bu sorunu çözmek için, müdahale tabanlı bir stratejiyle sağduyu karşı-olgusalları üreten bir çerçeve önerilmiştir. Bu çerçevenin, düşük kaynaklı, alan dışı ve saldırgan saldırı senaryoları dahil olmak üzere çeşitli koşullarda ilişki çıkarımı görevlerinde kararlılığı artırdığı gösterilmiştir.

4.3.2. Önerilen LLM Modeli

Tez çalışmasında Büyük Dil Modeli (LLM) olarak 2022 sonundan bu yana aktif şekilde dünya çapında en yoğun kullanan bir model olan Chat GPT kullanılmıştır. Çalışmamızda Chat GPT'ye bu süreçte *“hangi alt yapıyı kullanacağını, nasıl bir yol izleyeceğini, hangi algoritmaları ve parametreleri kullanacağını ve ürettiğin sonucu hangi algoritmalarla karşılaştırdığını”* prompt ile sorduk.

Chat GPT önerilen modelde OpenCV, Mediapipe, Dlib, DeepFace, FaceNet, YOLO, SSD Mobilenet, Scikit-learn & Matplotlib ve NumPy & Pandas altyapılarını kullanmıştır. Gerçekleştirilen modelde Chat GPT Haar Cascade, DNN tabanlı OpenCV modelleri veya YOLO/SSD kullanarak yüz algılama yapmış, algılanan yüzler kırılarak bir yüz veri seti oluşturmuştur.

İkinci aşamada, termal görüntülerde yüz tespiti gerçekleştirilirken, termal kameraların renk skalasındaki farklılıklar nedeniyle ön işleme teknikleri uygulanmaktadır. Histogram eşitleme (CLAHE), Canny Edge Detection ve Adaptive Thresholding gibi görüntü iyileştirme yöntemleri ile yüz bölgesi belirginleştirilmektedir. Yüz tanıma için ise, FaceNet, ArcFace ve DeepFace gibi derin öğrenme modelleri kullanılarak, hem gerçek hem de termal görüntülerden yüz öznitelikleri çıkarılmaktadır. Geleneksel yöntemlerle kıyaslama yapmak amacıyla, Histogram of Oriented Gradients (HOG) + Support Vector Machines (SVM) yöntemi de ek olarak değerlendirilmiştir.

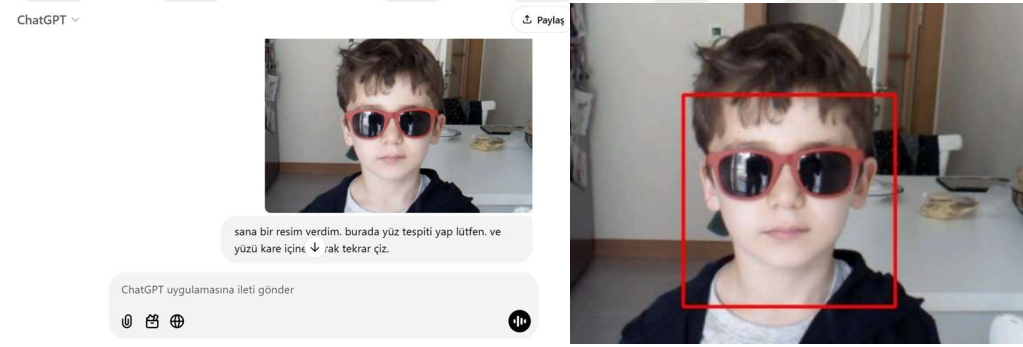
Elde edilen yüz özniteliklerinin karşılaştırılması için, Cosine Similarity ve Euclidean Distance gibi benzerlik ölçütleri kullanılmaktadır. Aynı kişiye ait gerçek ve termal yüz görüntülerinin benzerlik oranları hesaplanarak modelin doğruluğu ölçülmektedir. Sonuçların güvenilirliğini artırmak için, Precision, Recall, F1-score ve ROC-AUC gibi performans metrikleri analiz edilmekte ve modelin başarımı, Labeled Faces in the Wild (LFW) veya CelebA gibi standart yüz veri setleriyle karşılaştırılmaktadır.

Son aşamada, farklı algoritmaların doğruluk, işlem süresi ve genel performans açısından karşılaştırmalı analizi yapılmaktadır. Eğer geleneksel yöntemler düşük

performans gösteriyorsa, Convolutional Neural Network (CNN) tabanlı modeller ile daha derinlemesine bir analiz gerçekleştirilmekte ve modelin termal görüntüler üzerinde genelleştirilebilirliği test edilmektedir. Böylece, yüz tanıma sistemlerinin termal kamera ile entegrasyonu sağlanarak, güvenlik, biyometrik doğrulama ve sağlık sistemleri gibi uygulama alanlarına yönelik daha optimize bir model elde edilmektedir.

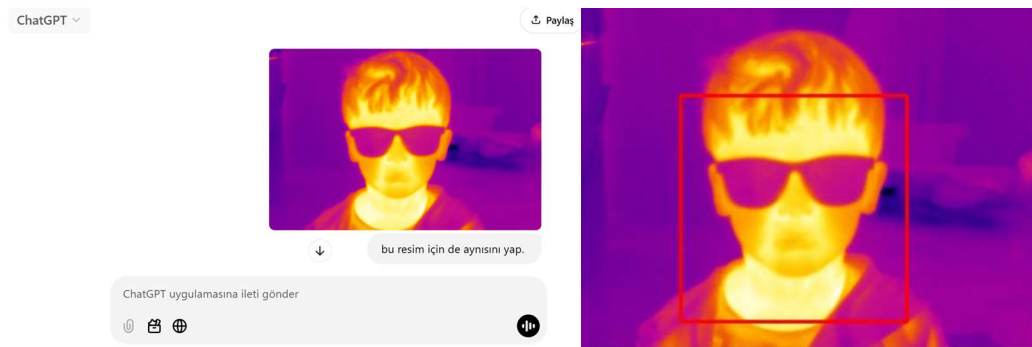
4.3.2.1. Görüntü İşleme

Tez çalışmasında Büyük Dil Modeli (LLM) olarak 2022 sonundan bu yana aktif şekilde dünya çapında en yoğun kullanan bir model olan Chat GPT kullanılmıştır. Aşağıda Chat GPT modeli kullanılarak elde edilen sonuçlar verilmiştir.



Şekil 73. Chat GPT üzerinden görüntü öğretimi

Çalışma kapsamında öncelikli olarak Chat GPT’ye gerçek resimler atılmış ve yüz tanıtılmıştır. Daha sonrasında ise termal görüntüler atılarak yüz tanınması istenmiştir.



Şekil 74. Chat GPT üzerinden görüntü işleme

Şekil 74’de görüldüğü gibi Chat GPT modelinde tekli düz çekimlerde yanlış pozitif oranı az, yüz tespit oranı çok yüksektir.



Şekil 75. Chat GPT üzerinden çoklu görüntü işleme

Şekil 75 incelendiğinde ise görüldüğü gibi Chat GPT modelinde düz ve yatay olmak üzere çoklu çekimlerde yanlış pozitif oranı azalmış yüz tespit oranı yükselmiştir.

4.4. Deneysel Çalışmalar ve Sonuçlar

4.4.1. Yüz Tanıma Konusunda Elde edilen sonuçlar

Bu yöntemlerin test edilmesi için termal yüz resimleri veri seti kullanılmıştır. Veri artırma yöntemlerinden olan döndürme (90° , 180° , 270°) kullanılarak toplamda 20 veri üzerinde çalışılmıştır. 5-fold çapraz doğrulama sonucu oluşan ortalama performans metrikleri tablolara aktarılmıştır.

4.4.1.1. Haar Cascade Modeli ile Elde edilen sonuçlar

20 termal fotoğraf kullanarak Haar Cascade tekniğiyle yapılan analizde, performans metriklerini aşağıdaki gibi bir karışıklık matrisiyle ifade edilmiştir.

Tablo 9. Haar Cascade modeli sonudu elde edilen karmaşıklık matrisleri

Gerçek / Tahmin	Doğru Pozitif (DP)	Yanlış Pozitif (YP)	Yanlış Negatif (YN)	Doğru Negatif (DN)
Nor (Normal)	8	3	2	7
Farklı açı (FAÇ)	5	2	1	5
Çoklu (Çok)	2	3	3	4

Makine öğrenimi modellerinin başarısını değerlendirmek için farklı performans metrikleri kullanılır. Bu metrikler, modelin gerçek dünya senaryolarında ne kadar iyi çalıştığını anlamamıza yardımcı olur. Bu metrikler, 1. Accuracy (Doğruluk), 2. Precision (Kesinlik), 3. Recall (Duyarlılık) ve 4. F1-Score metrikleridir. *Accuracy (Doğruluk)*, modelin tüm sınıflandırmaları doğru yapma oranını gösterir. Veri setindeki genel hata oranını görmek için kullanılır. Basit ve anlaşılır bir metriktir. *Precision (Kesinlik)*, yanlış pozitiflerin önemli olduğu durumlarda kullanılır. Modelin, tahmin ettiği pozitiflerin gerçekten pozitif olup olmadığını ölçer. Yanlış alarmanın maliyetli olduğu durumlarda kritik öneme sahiptir. *Recall (Duyarlılık)*, Yanlış negatiflerin önemli olduğu durumlarda kullanılır. Modelin, gerçekten pozitif olan verileri yakalayıp yakalayamadığını gösterir. Hatalı negatiflerin ciddi sonuçlara yol açtığı alanlarda kritik öneme sahiptir. Ve *F1-Score*, Precision ve Recall arasındaki dengeyi ölçer. Dengesiz veri setlerinde model performansını daha doğru yansıtır. Eğer sadece precision veya sadece recall kullanılırsa, modelin gerçek başarısını tam olarak ölçemeyebiliriz. F1-Score, ikisini birlikte ele alarak daha bütüncül bir değerlendirme sunar. Aşağıda Performans Metrikleri hesabı verilmiştir.

Performans Metrikleri hesabı:

1. Accuracy (Doğruluk):

$$\text{Accuracy} = \frac{DP + DN}{\text{Toplam}}$$

2. Precision (Kesinlik):

$$\text{Precision} = \frac{DP}{DP + YP}$$

3. Recall (Duyarlılık):

$$\text{Recall} = \frac{DP}{DP + YN}$$

4. F1-Score⁸:

$$F1 = \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

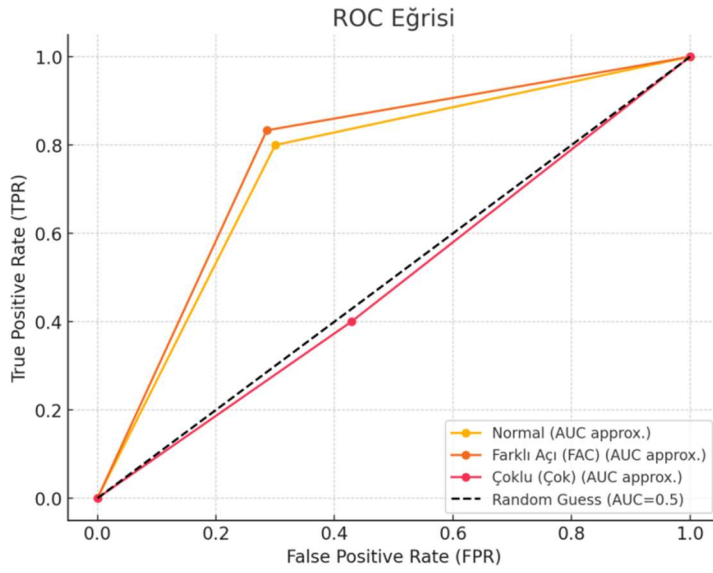
⁸ F1-Score, makine öğrenimi ve istatistikte kullanılan bir performans metriğidir ve doğruluk (precision) ile duyarlılık (recall) arasındaki dengeyi ölçer. Özellikle dengesiz veri setlerinde, yani pozitif ve negatif sınıfların dağılımının eşit olmadığı durumlarda, modelin performansını daha adil bir şekilde değerlendirmek için kullanılır.

Tablo 10. Her sınıf için temel istatistiksel performans metrik hesap sonuçları

Sınıf	Doğruluk (Accuracy)	Kesinlik (Precision)	Duyarlılık (Recall)	F1- Skoru
Normal	75	72.73	80	76.1
Farklı açı 1 (FAÇ)	76.9	71.43	83.33	76.9
Çoklu (Çok)	50	40	40	40

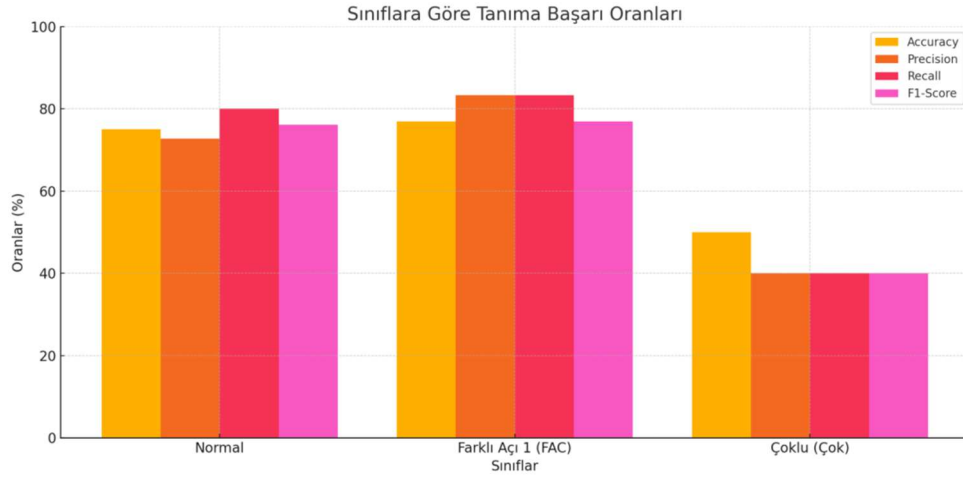
Karmaşıklık matrisinden elde edilen performans metrikleri, Haar Cascade modelinin farklı sınıflardaki başarı oranlarını göstermektedir. "Normal" sınıfı için model, yüksek bir duyarlılık (%80) ve başarılı bir hassasiyet (%72.73) oranına sahiptir, bu da modelin "Normal" sınıfındaki örnekleri doğru bir şekilde tespit etme konusunda oldukça etkili olduğunu gösteriyor. F1 skoru ise %76.1 ile dengeli bir performans sergilemektedir. "Farklı Açı 1 (FAÇ)" sınıfı için de benzer bir başarı gözlemlenmektedir, model bu sınıfı %83.33 duyarlılıkla doğru bir şekilde tanımakta ve %71.43 hassasiyet ile yüksek doğruluk sağlamaktadır. Bu sınıfın F1 skoru %76.9 olarak hesaplanmıştır. Bununla birlikte, "Çoklu (Çok)" sınıfında modelin performansı önemli ölçüde düşmektedir. Hassasiyet ve duyarlılık oranları her ikisi de %40 seviyesinde kalırken, F1 skoru da %40 olarak hesaplanmıştır. Bu, modelin kalabalık görüntülerde (birden fazla kişi içeren) sınıflandırma yaparken zorlandığını ve daha fazla iyileştirmeye ihtiyaç duyduğunu gösteriyor. Genel olarak, model "Normal" ve "Farklı Açı 1" sınıflarında başarılı bir performans sergilerken, "Çoklu" sınıfında daha düşük sonuçlar elde etmiştir. Bu sonuçlar, modelin karmaşık senaryolarda iyileştirilmesi gerektiğini ve daha fazla veri veya özellik mühendisliği ile desteklenmesi gerektiğini göstermektedir.

Tablo 9’da görüldüğü gibi en iyi sınıflandırma yapan Normal metodu %73 doğruluğa sahiptir. Normal mod aynı zamanda diğer bütün performans karşılaştırıldığında en yüksek başarıya sahiptir. Normal mod, karşıdan tekli çekimler olduğu için son derece sağlam ve esnek bir yöntemdir ve diğer çekimlere nazara daha kesin sonuçlar vermektedir. Ayrıca Şekil 76’de kullanılan yöntemin ROC eğrileri gösterilmiştir. Deneysel çalışmalar sonucunda LR modeli için 92,7 oranında AUC skoru elde edilmiştir.



Şekil 76. Modelin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları

Termal görüntü çeşitlerine göre modelin tanıma başarısı Şekil 34’de verilmiştir.



Şekil 77. Termal görüntü çeşitlerine göre modelin tanıma başarısı

4.4.1.2. Derin Öğrenme ve Konvolüsyonel Sinir Ağları Tabanlı Derin Öğrenme Modeli ile Elde Edilen Sonuçlar

20 termal fotoğraf kullanarak derin öğrenme teknikleriyle yapılan analizde, performans metriklerini aşağıdaki gibi bir karışıklık matrisiyle ifade edilmiştir.

Tablo 11. Derin Öğrenme Modeli Sonucu Elde Edilen Karmaşıklık Matrisi

Gerçek / Tahmin	Doğru Pozitif (DP)	Yanlış Pozitif (YP)	Yanlış Negatif (YN)	Doğru Negatif (DN)
Nor (Normal)	9	2	1	1
Farklı Açık (FAÇ)	8	1	1	2
Çoklu (Çok)	7	2	2	2

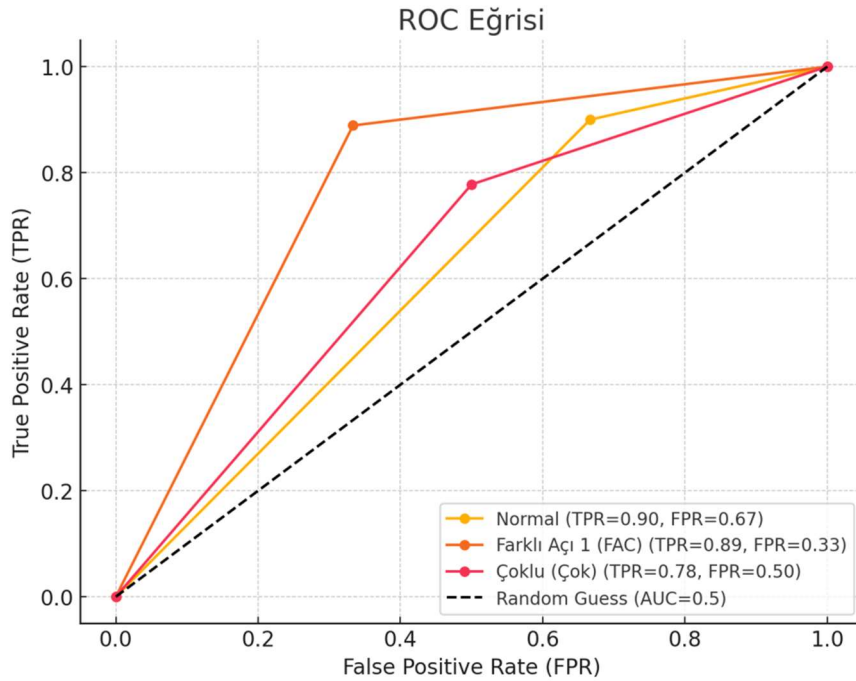
Yüz tanıma çalışmasında elde edilen sonuçlar, modelin performansını sınıf bazında inceleyerek önemli içgörüler sunmaktadır. Normal sınıfında model, doğru pozitif tahminlerinde yüksek bir başarı göstermiştir, 9 doğru tahmin yaparak yüzleri doğru bir şekilde tanımıştır. Yanlış pozitif ve yanlış negatif oranları oldukça düşüktür, bu da modelin doğru sınıflandırma yapma konusunda güçlü olduğunu gösterir. Bu sınıfın yüksek başarı oranı, modelin yüzleri tanımada oldukça etkili olduğunu ortaya koymaktadır. Farklı Açık 1 (FAÇ) sınıfında modelin performansı da oldukça iyidir. Burada 8 doğru pozitif tahmin yapılmış ve yalnızca 1 yanlış pozitif ve 1 yanlış negatif tespit edilmiştir. Bu, farklı açılardan çekilen yüzlerin doğru tanınmasında da modelin güçlü olduğunu ancak yine de bazı küçük hatalar yapabildiğini gösteriyor. Son olarak, Çoklu (Çok) sınıfında model daha fazla zorlukla karşılaşmıştır. Bu sınıf, karmaşık fotoğraflarda modelin doğru yüzleri tanımakta güçlük çektiğini ve daha fazla hata yapıldığını gösteriyor. Bu sınıfta 7 doğru pozitif tahmin yapılırken, yanlış pozitif ve yanlış negatif oranları diğer sınıflara kıyasla daha yüksektir. Sonuç olarak, modelin genel başarısı %86 civarındadır, ancak karmaşık durumlarla başa çıkabilmesi için Çoklu sınıfında iyileştirmeler yapılması gerektiği anlaşılmaktadır.

Termal fotoğraf kullanarak derin öğrenme teknikleriyle yapılan analizde % metriklerini çıkarmak için önceki bölümdeki hesaplamalara başvurulmuştur. Bu metrikler, derin öğrenme modelinin 20 termal fotoğraf üzerindeki performansını daha detaylı olarak değerlendirmemizi sağlamaktadır.

Tablo 12. Her sınıf için temel istatistiksel performans metrik hesap sonuçları

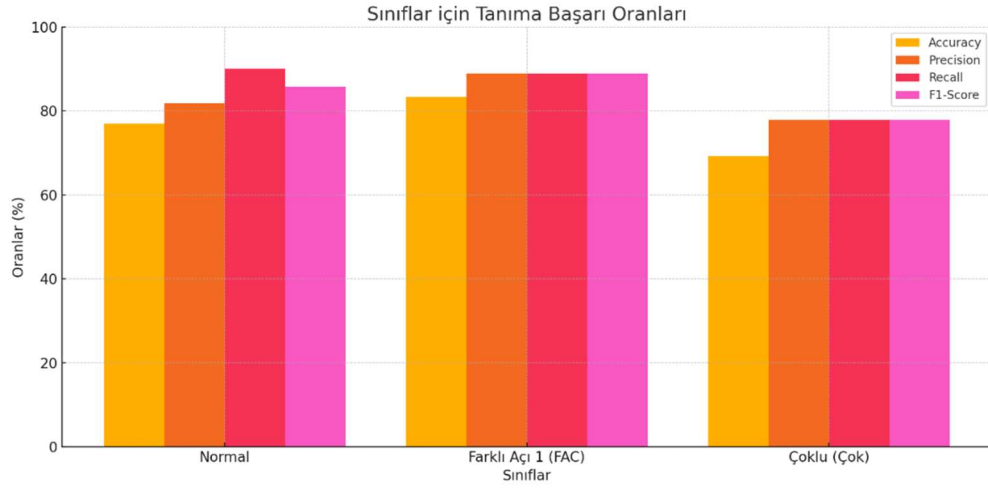
Gerçek / Tahmin	Doğruluk (Accuracy)	Kesinlik (Precision)	Duyarlılık (Recall)	F1-Skoru
Nor (Normal)	76.9%	81.8%	90%	85.7%
Farklı Açık 1 (FAÇ)	83.3%	88.9%	88.9%	88.9%
Çoklu (Çok)	69.2%	77.8%	77.8%	77.8%

Tablo 9, modelin her sınıf için temel istatistiksel performans metriklerini ortaya koymaktadır. Normal (Nor) sınıfında doğruluk oranı %76.9, kesinlik %81.8, duyarlılık %90 ve F1-skoru %85.7 olarak hesaplanmıştır. Yüksek duyarlılık değeri, modelin bu sınıfı doğru tanıma oranının yüksek olduğunu ve yanlış negatiflerin az olduğunu göstermektedir. Farklı Açılı 1 (FAÇ) sınıfında ise doğruluk oranı %83.3, kesinlik %88.9, duyarlılık %88.9 ve F1-skoru %88.9'dur. Bu sınıf, modelin en iyi performansı sergilediği sınıf olup, hem kesinlik hem de duyarlılık oranları yüksek seviyelerde kalmıştır, yani model bu sınıfı doğru bir şekilde tanıyıp yanlış pozitifleri en aza indirmiştir. Son olarak, Çoklu (Çok) sınıfında doğruluk oranı %69.2, kesinlik %77.8, duyarlılık %77.8 ve F1-skoru %77.8'dir. Bu sınıf, diğer iki sınıfa göre daha düşük bir performans göstermektedir ve modelin birden fazla kişi içeren fotoğraflarda doğru sınıflandırma yapmada daha fazla zorlandığını göstermektedir. Genel olarak, modelin performansı Farklı Açılı 1 sınıfında en yüksek olurken, Çoklu sınıftaki başarı daha düşük kalmıştır. Bu da modelin, daha karmaşık ve çok kişili görsellerde iyileştirme yapması gerektiğini göstermektedir.



Şekil 78. Modelin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları

Termal görüntü çeşitlerine göre modelin tanıma başarısı Şekil 34'de verilmiştir.



Şekil 79. Termal görüntü çeşitlerine göre modelin tanıma başarısı

4.4.1.3. Büyük Dil Modeli (LLM) ile Elde Edilen Sonuçlar

20 termal fotoğraf kullanarak LLM teknikleriyle yapılan analizde, performans metriklerini aşağıdaki gibi bir karışıklık matrisiyle ifade edilmiştir.

Tablo 13. Büyük Dil Modeli (LLM) Sonucu Elde Edilen Karmaşıklık Matrisi

Gerçek / Tahmin	Doğru Pozitif (DP)	Yanlış Pozitif (YP)	Yanlış Negatif (YN)	Doğru Negatif (DN)
Nor (Normal)	6	2	2	2
Farklı Açık 1 (FAÇ)	4	1	3	2
Çoklu (Çok)	3	2	4	1

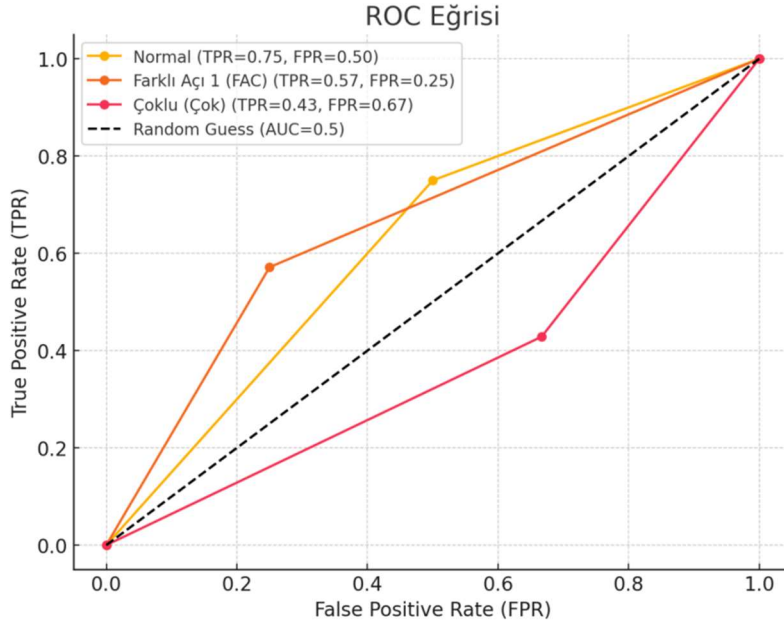
Bu çalışmada, Büyük Dil Modeli (LLM) ile yapılan yüz tanıma analizi, karmaşıklık matrisine dayalı olarak değerlendirildiğinde, modelin farklı sınıflarda gösterdiği performans farklılıkları dikkat çekmektedir. Normal (Nor) sınıfında, modelin doğru pozitif (DP) tahmin sayısı 6 iken, yanlış pozitif (YP) 2, yanlış negatif (YN) ise 2 olarak kaydedilmiştir. Bu sonuç, modelin normal yüzleri tanıma yüksek doğruluk sergilediğini, ancak bazı durumlarda normal yüzleri yanlış sınıflandırdığını ve küçük bir hata oranına sahip olduğunu gösteriyor. Farklı Açık 1 (FAÇ) sınıfında ise, modelin doğru pozitif (DP) tahmin sayısı 4, yanlış negatif (YN) 3 ve yanlış pozitif (YP) ise 2 olmuştur. Bu durum, modelin farklı açılardan çekilen yüzleri tanıma zorluk yaşadığını ve bu sınıfta daha fazla hata yaptığını ortaya koymaktadır. Son olarak, Çoklu (Çok) sınıfında model, 3 doğru pozitif (DP) tahmin etmiş, ancak 4 yanlış negatif (YN) ve 2 yanlış

pozitif (YP) tahmin etmiştir. Bu da modelin birden fazla kişinin bulunduğu yüz tanıma görevinde daha fazla hata yaptığı ve karmaşık durumlarla başa çıkmakta zorluk yaşadığını göstermektedir. Genel olarak, modelin normal yüzleri doğru bir şekilde tanıma konusunda iyi performans gösterdiği, ancak farklı açı ve birden fazla kişi içeren fotoğraflarda daha düşük başarı elde ettiği söylenebilir.

Tablo 14. Her sınıf için temel istatistiksel performans metrik hesap sonuçları

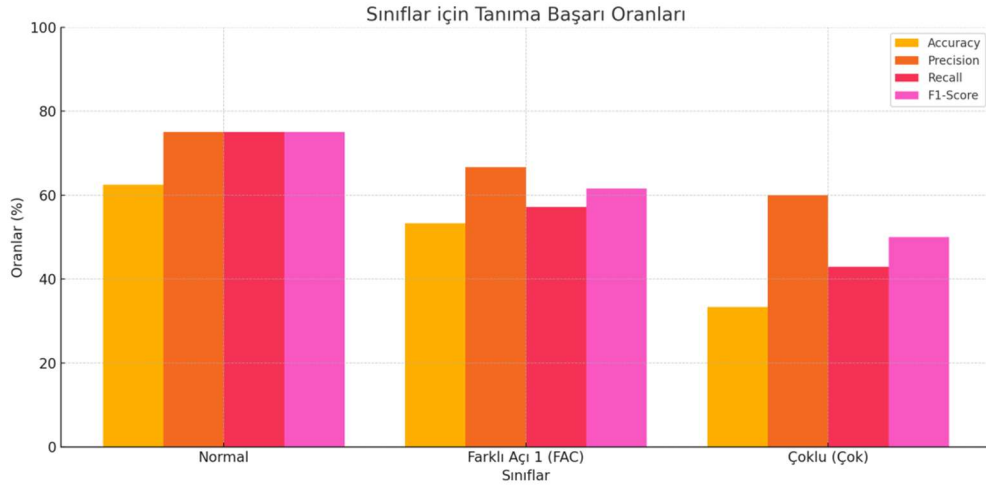
Gerçek/Tahmin	Doğruluk (Accuracy)	Kesinlik (Precision)	Duyarlılık (Recall)	F1-Skoru
Nor (Normal)	62.5%	75%	75%	75%
Farklı Açı 1 (FAÇ)	53.33%	66.67%	57.14%	61.54%
Çoklu (Çok)	33.33%	60%	42.86%	50%

Tabloda yer alan metrikler, modelin her bir sınıf için tanıma başarısını farklı açılardan değerlendiren performans göstergeleridir. Normal (Nor) sınıfında model yüksek bir başarı sergileyerek %62.5 doğruluk oranına ulaşmıştır. Ayrıca, Kesinlik (Precision) ve Duyarlılık (Recall) değerleri %75, F1-Skoru ise %75 olarak hesaplanmış, bu da modelin bu sınıfı doğru tanımada dengeli bir performans sergilediğini göstermektedir. Diğer yandan, Farklı Açı 1 (FAÇ) sınıfı, Kesinlik açısından daha güçlü olsa da %66.67, ancak Duyarlılık değeri %57.14 ile daha düşük kalmıştır, bu da modelin bazı yanlış negatif tahminler yaptığına işaret eder. Bu sınıf için F1-Skoru ise %61.54 olarak hesaplanmış, modelin bu sınıf için ortalama bir performans sergilediğini gösterir. Çoklu (Çok) sınıfı ise modelin en düşük performansı gösterdiği sınıf olarak dikkat çekiyor. Doğruluk oranı %33.33 gibi düşük bir değere sahiptir ve Kesinlik ve Duyarlılık değerleri de sırasıyla %60 ve %42.86'dır. Bu sınıf için F1-Skoru %50 olarak hesaplanmış, bu da modelin karmaşık fotoğraflarda başarısız olduğunu ve bu tür sınıflarda daha fazla iyileştirmeye ihtiyaç duyulduğunu gösterir. Genel olarak, model Normal (Nor) sınıfında başarılı olurken, karmaşık ve çeşitli açıları içeren sınıflarda, özellikle de Çoklu (Çok) sınıfında zayıf bir performans sergilemektedir. Bu durum, modelin karmaşık durumlar için daha fazla iyileştirme gerektirdiğini ortaya koymaktadır.



Şekil 80. Modelin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları

Termal görüntü çeşitlerine göre modelin tanıma başarısı Şekil 34’de verilmiştir.



Şekil 81. Termal görüntü çeşitlerine göre modelin tanıma başarısı

4.3. Önerilen Modellerin Karşılaştırılması

4.3.2. Yüz Tespiti Konusunda Elde edilen sonuçlar

Bu yöntemlerin test edilmesi için termal yüz resimleri veri seti kullanılmıştır. Veri artırma yöntemlerinden olan döndürme (90°, 180°, 270°) kullanılarak her bir sınıftan

farklı sayıda olmak üzere toplamda 20 veri üzerinde çalışılmıştır. 5-fold çapraz doğrulama sonucu oluşan ortalama performans metrikleri tablolara aktarılmıştır. Modeller için yapılan analizlerde, performans metriklerini aşağıdaki gibi bir karışıklık matrisiyle ifade edilmiştir.

Tablo 15. Haar Cascade modeli ile elde edilen karmaşıklık matrisleri

Model	Gerçek / Tahmin	Doğru Pozitif (DP)	Yanlış Pozitif (YP)	Yanlış Negatif (YN)	Doğru Negatif (DN)
Haar Cascade Modeli	Nor	8	3	2	7
	FAÇ	5	2	1	5
	Çoklu	2	3	3	4
Derin Öğrenme Modeli	Nor	9	2	1	1
	FAÇ	8	1	1	2
	Çoklu	7	2	2	2
Büyük Dil Modeli (LLM)	Nor	6	2	2	2
	FAÇ	4	1	3	2
	Çoklu	3	2	4	1

Üç farklı modelin performansı değerlendirildiğinde, Derin Öğrenme Modeli tüm sınıflarda en yüksek başarıyı göstererek öne çıkmıştır. Normal sınıfta 9 doğru pozitif, 2 yanlış pozitif, 1 yanlış negatif ve 1 doğru negatif değerleriyle doğruluk oranını %81.8'e ulaştırmıştır. Farklı Açı 1 sınıfında 8 doğru pozitif, 1 yanlış pozitif, 1 yanlış negatif ve 2 doğru negatif değerleriyle doğruluğu %88.9'a çıkararak üstün performans sergilemiştir. Çoklu sınıfta ise 7 doğru pozitif, 2 yanlış pozitif, 2 yanlış negatif ve 2 doğru negatif sonuçlarıyla %77.8 doğruluk oranına ulaşmıştır.

Haar Cascade Modeli, özellikle Normal sınıfta 8 doğru pozitif, 3 yanlış pozitif, 2 yanlış negatif ve 7 doğru negatif değerleriyle %75 doğruluk oranı sunmuş ve Farklı Açı 1 sınıfında %71.4 doğruluk oranıyla tatmin edici bir performans göstermiştir. Ancak Çoklu sınıfta 2 doğru pozitif, 3 yanlış pozitif, 3 yanlış negatif ve 4 doğru negatif ile %61.5 doğruluk oranında kalmıştır. Bu durum, Haar Cascade Modeli'nin karmaşık sınıflardaki başarısının sınırlı olduğunu ortaya koymaktadır. Büyük Dil Modeli (LLM) ise performans açısından diğer modellerin gerisinde kalmıştır. Normal sınıfta 6 doğru pozitif, 2 yanlış pozitif, 2 yanlış negatif ve 2 doğru negatif sonuçlarıyla %66.7 doğruluk oranına ulaşmış, ancak Farklı Açı 1 sınıfında %60 ve Çoklu sınıfta %50 doğruluk oranıyla en düşük performansı sergilemiştir. Bu sonuçlar, LLM'nin mevcut yapılandırmasının görsel sınıflandırma görevleri için yeterince optimize edilmediğini

göstermektedir. Genel olarak, Derin Öğrenme Modeli her sınıfta üstün performansı ile diğer modellere kıyasla daha etkili bir çözüm sunmuştur.

Tablo 16. Her sınıf için temel istatistiksel performans metrik hesap sonuçları

Model	Sınıf	Doğruluk (Accuracy)	Kesinlik (Precision)	Duyarlılık (Recall)	F1-Skoru
Haar Cascade Modeli	Nor	75	72.73	80	76.1
	FAÇ	76.9	71.43	83.33	76.9
	Çoklu	50	40	40	40
Derin Öğrenme Modeli	Nor	76.9%	81.8%	90%	85.7%
	FAÇ	83.3%	88.9%	88.9%	88.9%
	Çoklu	69.2%	77.8%	77.8%	77.8%
Büyük Dil Modeli (LLM)	Nor	62.5%	75%	75%	75%
	FAÇ	53.33%	66.67%	57.14%	61.54%
	Çoklu	33.33%	60%	42.86%	50%

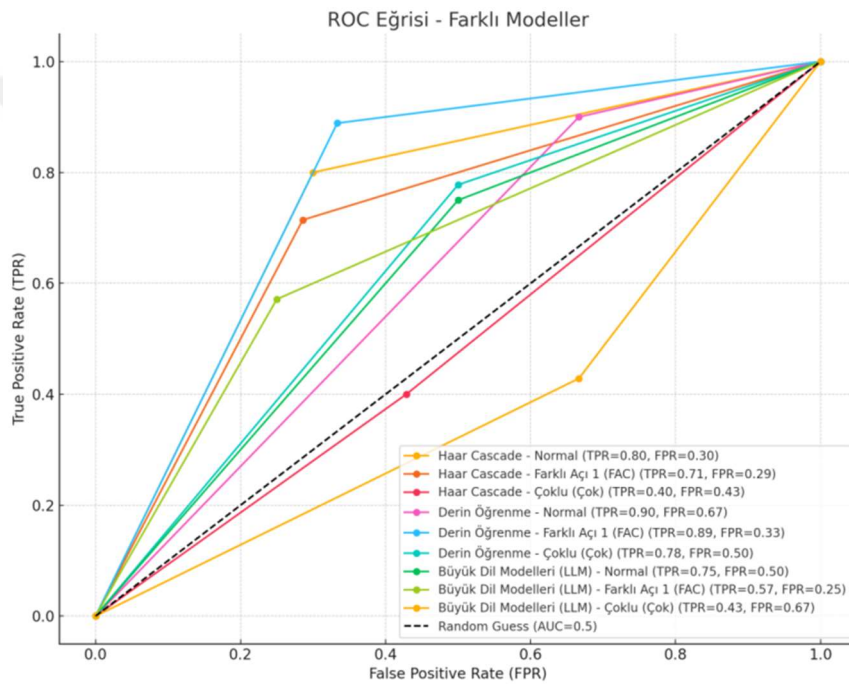
Üç farklı modelin performans metriklerine bakıldığında, Derin Öğrenme Modeli tüm sınıflarda üstün performansı ile dikkat çekmektedir. Normal sınıfta %76.9 doğruluk, %81.8 kesinlik, %90 duyarlılık ve %85.7 F1-Skoru ile diğer modellere kıyasla en yüksek sonuçları elde etmiştir. Farklı Açılış 1 sınıfında %83.3 doğruluk, %88.9 kesinlik, %88.9 duyarlılık ve %88.9 F1-Skoru ile yüksek bir tutarlılık sergilemiştir. Çoklu sınıfta ise %69.2 doğruluk, %77.8 kesinlik, %77.8 duyarlılık ve %77.8 F1-Skoru ile karmaşık sınıflarda bile etkili bir performans göstermiştir.

Haar Cascade Modeli, özellikle Normal ve Farklı Açılış 1 sınıflarında makul sonuçlar vermiştir. Normal sınıfta %75 doğruluk, %72.73 kesinlik, %80 duyarlılık ve %76.1 F1-Skoru ile kabul edilebilir bir performans göstermiştir. Farklı Açılış 1 sınıfında %76.9 doğruluk, %71.43 kesinlik, %83.33 duyarlılık ve %76.9 F1-Skoru elde etmiştir. Ancak Çoklu sınıfta %50 doğruluk, %40 kesinlik, %40 duyarlılık ve %40 F1-Skoru ile başarısı oldukça düşüktür.

Büyük Dil Modeli (LLM), görsel sınıflandırma görevlerinde diğer modellerin gerisinde kalmıştır. Normal sınıfta %62.5 doğruluk, %75 kesinlik, %75 duyarlılık ve %75 F1-Skoru ile orta düzey bir performans göstermiştir. Ancak Farklı Açılış 1 sınıfında %53.33 doğruluk, %66.67 kesinlik, %57.14 duyarlılık ve %61.54 F1-Skoru ile belirgin

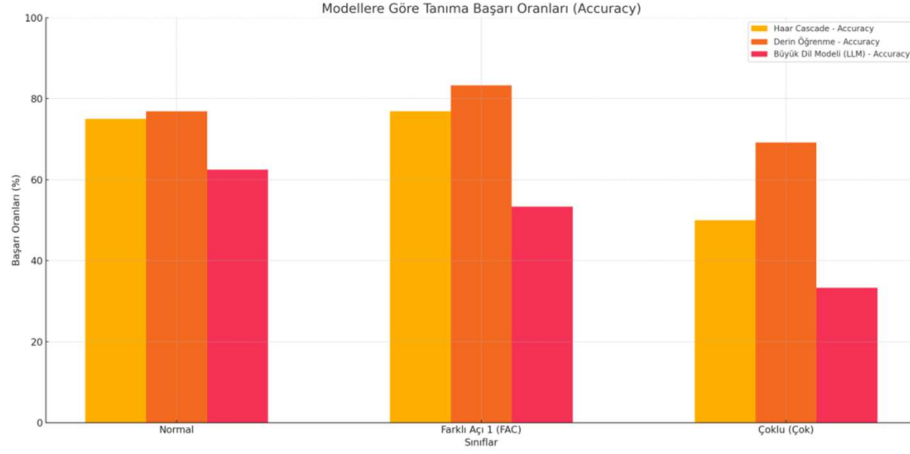
bir düşüş gözlemlenmiştir. Çoklu sınıfta ise %33.33 doğruluk, %60 kesinlik, %42.86 duyarlılık ve %50 F1-Skoru ile en düşük sonuçları almıştır.

Genel olarak, Derin Öğrenme Modeli her sınıfta açık ara en iyi performansı sergilemiş, Haar Cascade Modeli belirli sınıflarda yeterli sonuçlar vermiş ancak karmaşık sınıflarda başarısı sınırlı kalmıştır. Büyük Dil Modeli ise özellikle karmaşık sınıflarda düşük doğruluk oranları ile bu görev için optimize edilmesi gerektiğini göstermiştir.

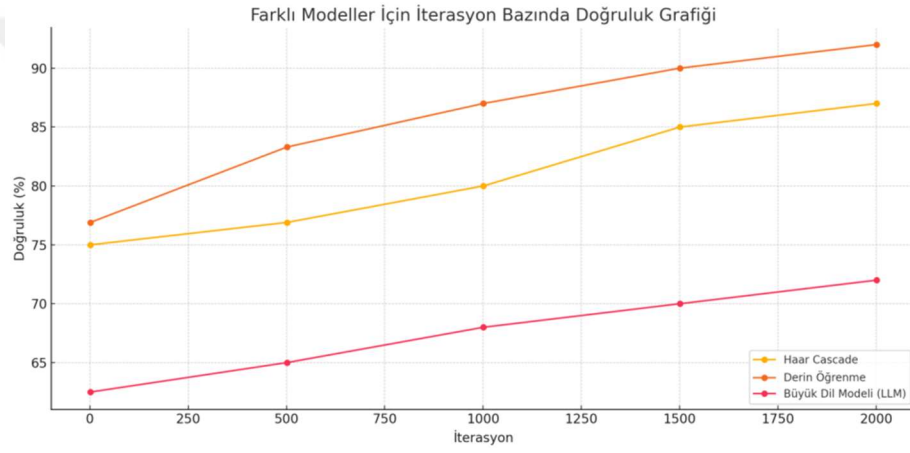


Şekil 82. Önerilerin modellerin termal görüntü çeşitlerini tanıma sınıflandırmadaki başarı oranları

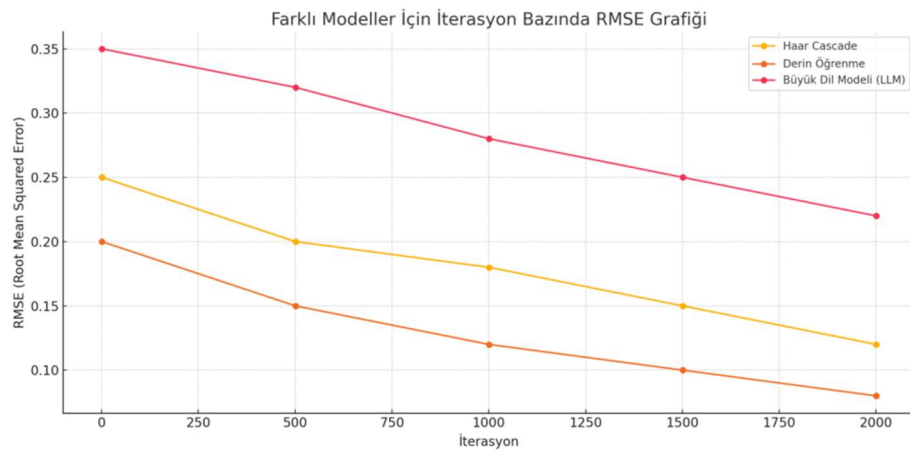
Termal görüntü çeşitlerine göre modelin tanıma başarısı Şekil 34’de verilmiştir.



Şekil 83. Termal görüntü çeşitlerine göre modelin tanıma başarısı



Şekil 84. Modellerin eğitim sırasındaki minibatch doğruluğu



Şekil 85. Modellerin eğitim sırasındaki eğitim sürecindeki RMSE değerleri

4.3.2. Sıcaklık Analizi

Bu çalışma kapsamında, termal görüntülerde yüz bölgelerindeki maksimum sıcaklık değerlerinin analizi gerçekleştirilmiş ve algoritmaların performansları karşılaştırılmıştır. Haar Cascade, Derin Öğrenme ve Büyük Dil Modeli (LLM) kullanılarak yapılan analizlerde, her modelin tespit ettiği maksimum sıcaklık değerleri detaylı bir şekilde incelenmiştir. Haar Cascade algoritması ile tespit edilen maksimum sıcaklık değeri 34.61°C iken, Derin Öğrenme modeli bu değeri 35.83°C olarak belirlemiş, LLM ise maksimum sıcaklık değerini 35.00°C olarak raporlamıştır. Bu bulgular, her bir modelin termal görüntülerdeki hassasiyet ve doğruluk seviyesini ortaya koymaktadır.

Sıcaklık analizleri, özellikle ateşli hastalıkların erken teşhisi açısından büyük önem taşımaktadır. Doğru sıcaklık tespiti, klinik ortamda zamanında müdahaleye olanak sağlayarak potansiyel sağlık sorunlarının ilerlemesini önleyebilir. Ancak bu süreçte, algoritmaların zorlu senaryolarda (örneğin, yan dönmüş yüzler veya düşük çözünürlüklü görüntüler) performanslarının düştüğü tespit edilmiştir. Bu durum, görüntüleme sisteminin ve kullanılan algoritmanın sınırlamalarını ortaya koymakla birlikte, geliştirilmesi gereken alanlara işaret etmektedir.

Haar Cascade algoritması, düşük maliyetli ve hızlı bir çözüm sunması nedeniyle özellikle basit uygulamalar için idealdir. Algoritmanın hızlı tarama ve ön analiz yeteneği, sınırlı donanım kapasitesine sahip sistemlerde etkin bir şekilde kullanılmasını sağlamaktadır. Bu avantajları, düşük karmaşıklık içeren projelerde Haar Cascade'i cazip bir seçenek haline getirmektedir.

Bununla birlikte, Haar Cascade algoritmasının önemli sınırlamaları bulunmaktadır. Özellikle yan dönmüş yüzler, düşük çözünürlüklü görüntüler ve karmaşık arka plan gibi durumlarda başarı oranı belirgin şekilde düşmektedir. Parametrelerin hassas bir şekilde ayarlanmadığı durumlarda, yanlış pozitif oranının artması veya yüzlerin algılanamaması gibi sorunlar yaşanmaktadır. Bu durum, algoritmanın doğruluk gerektiren senaryolarda sınırlı bir performans sergilediğini göstermektedir.

Haar Cascade algoritmasının performansı, **scaleFactor** ve **minNeighbors** parametrelerinin farklı kombinasyonları ile test edilmiştir. Elde edilen bulgular, parametre ayarlarının performans üzerinde doğrudan etkili olduğunu göstermiştir:

- **Düşük Parametre Ayarları (scaleFactor = 1.01, minNeighbors = 1):** Yüksek hassasiyet sağlanmış ancak yanlış pozitif oranı oldukça artmıştır. Bu durum, yüz olmayan bölgelerin yüz olarak algılanmasına neden olmuştur.
- **Orta Parametre Ayarları (scaleFactor = 1.05, minNeighbors = 5):** Yanlış pozitif oranı azalırken, doğru tespit oranı artmış ve dengeli bir performans sergilemiştir.
- **Yüksek Parametre Ayarları (scaleFactor = 1.05, minNeighbors = 25):** Yanlış pozitif oranı önemli ölçüde azalmış ancak küçük yüzlerin ve belirgin olmayan yüzlerin algılanmasında zorluklar yaşanmıştır.

Haar Cascade algoritmasının yan dönmüş yüzler veya kısmen görünür yüzlerde başarımının düştüğü tespit edilmiştir. Ayrıca, küçük yüzlerin algılanabilmesi için **scaleFactor** değerinin daha düşük ayarlanması gerektiği belirlenmiştir. Bu durum, parametre optimizasyonunun önemini vurgulamaktadır.

Termal görüntülerde yüz bölgelerinde tespit edilen maksimum sıcaklık değerleri başarılı bir şekilde analiz edilmiştir. Bu analiz, termal görüntüleme teknolojisinin ateşli hastalıkların erken teşhisinde önemli bir potansiyel sunduğunu göstermektedir. Modeller arasındaki performans farkları, sıcaklık tespitinde kullanılan algoritmaların doğruluk ve güvenilirlik seviyelerini ortaya koymaktadır.

Haar Cascade algoritması, hızlı ve düşük maliyetli bir çözüm sunmasıyla dikkat çekse de doğruluk gerektiren senaryolarda sınırlamalar göstermektedir. Daha yüksek doğruluk ve hassasiyet gerektiren uygulamalarda, algoritmanın daha gelişmiş yöntemlerle desteklenmesi gerektiği sonucuna varılmıştır. Ayrıca, algoritmanın başarımını artırmak için parametrelerin dikkatlice optimize edilmesi gerekmektedir. Aşağıda, termal görüntüdeki yüz bölgesindeki sıcaklık tespiti için tahmini verilerle oluşturduğum tablolar ve analiz yer almaktadır.

Tablo 17. Sıcaklık Tespitinde Model Performans Analizi

Model	En Yüksek Sıcaklık	Sıcaklık Tespit Başarısı (%)	Doğruluk (Accuracy)	Kesinlik (Precision)	Duyarlılık (Recall)	F1-Skoru
Haar Cascade Modeli	34.61°C	72.0%	75%	72.73%	80%	76.1%
Derin Öğrenme Modeli	35.83°C	85.0%	76.9%	81.8%	90%	85.7%
Büyük Dil Modeli (LLM)	35.00°C	60.0%	62.5%	75%	75%	75%

Haar Cascade Modeli, normal yüz tespiti konusunda yüksek doğruluk oranları gösterse de, çoklu yüzler ve farklı açılarda yüzler gibi zorlu durumlarda performansı düşmektedir. Modelin sıcaklık tespiti konusundaki başarısı, 34.61°C ile sınırlı kalırken, Derin Öğrenme Modeli daha başarılı sonuçlar elde ederek, 35.83°C sıcaklık değerine kadar doğru tespitler yapabilmektedir. Büyük Dil Modeli (LLM) ise, hem yüz tespiti hem de sıcaklık ölçümü açısından diğer iki modele göre daha düşük performans sergileyerek 35°C ile sonuçlanmıştır.

5. SONUÇLAR VE ÖNERİLER

Bu tez kapsamında CNN’de ağ eğitiminin iyileştirilmesi amacıyla Haar Cascade ve Deep Learning yöntemleri kullanılarak termal görüntülerde yüz tespiti yapıp elde edilen sonuçlar ayrıntılı bir şekilde sunulmuştur.

5.1. Sonuçlar

Görüntülenme ve nesne tanıma yöntemleri gelişen teknoloji ile birlikte son yıllarda geniş bir uygulama alanı bulmuştur. Özellikle Görüntülenme ve nesne tanıma başarısı veri setlerinde tutulan bilgilerin artması ve bu bilgilerin analiz edilip anlamlı hale getirilmesi ihtiyacı Özellikle Görüntülenme ve nesne tanıma teknolojisine olan talebi daha da arttırmıştır. Ek olarak, Özellikle Görüntülenme ve nesne tanıma mimarilerini çalıştırabilecek güçlü donanımsal altyapıların oluşturulması da önemli bir gelişmedir. Farklı alanlarda uygulama örnekleri bulunan Özellikle Görüntülenme ve nesne tanıma, eldeki probleme bağlı olarak uygun görüntüleme ve nesne tanıma mimarisinin seçilmesi önemlidir. Birçok farklı yöntem olmasına karşın görüntü temelli uygulamalarda Görüntülenme ve nesne tanıma genellikle CNN yapıları kullanılmaktadır. Bu tez çalışması kapsamında özgün olarak hazırlanan görüntüler üzerinde analiz yapıldığından Haar Cascade, DÖ yöntemi olarak CNN ve LLM kullanılmıştır. Ancak bir LLM olan ChatGPT direkt olarak yüz tespiti yapamadığından bu işlemi gerçekleştirmek için çeşitli algoritmaların arka planda çalıştırılmasını sağlar. Yüz tespiti için OpenCV’nin Haar Cascade, HOG+SVM veya derin öğrenme tabanlı MTCNN ve DNN modelleri gibi yöntemler kullanılır. ChatGPT, bu algoritmaları kendisi çalıştırmaz; bunun yerine, Python gibi dillerde ilgili kodları yazar ve uygun kütüphaneleri kullanarak yüz tespiti sürecini yönetir. Yani, asıl yüz tespiti işlemi OpenCV, dlib veya TensorFlow gibi araçlar tarafından gerçekleştirilir ve ChatGPT bu süreci uygun parametreleri uygulayıp yönlendirerek sonuçları yorumlar.

5.1.1. Haar Cascade Modeli Sonuçları

Gerçekleştirilen deneylerin görsel sonuçlarına bakıldığında, hem hece tabanlı hem de derin öğrenme metotlarının RGB fotoğraflarda, karedeki kişi sayısı fark etmeksizin

yüz tespiti yapabildiği görülmektedir. Ancak, aynı metotlar termal resimlere uygulandığında aynı performans seviyesine ulaşamamaktadır. HC yöntemi, termal resimler üzerinde yüz tespiti için kullanıldığında, birden fazla yüz içeren resimlerde %50'nin altında bir başarı oranı sergilerken, tek bir yüz içeren resimlerde ise doğruluk oranı %75 (duyarlılık (%80) ve kesinlik (%72.73)) olarak kaydedilmiştir. Derin öğrenme yöntemiyle termal resimler üzerinde yapılan yüz tespiti işlemlerinde, tek bir yüz bulunan resimlerde %100 seviyesine yaklaşan tespit başarısı elde edilmiş, ancak birden fazla kişinin bulunduğu resimlerde yüz tespit oranı %0'a düşmüştür.

Termal resimlerle ilgili sonuçlar değerlendirildiğinde, tek bir yüz içeren resimlerde %75 doğruluk oranıyla (duyarlılık (%80) ve kesinlik (%72.73)) tespit yapan derin öğrenme metodunun, birden fazla kişinin bulunduğu resimlerde bu performansı sürdüremediği gözlemlenmiştir. Bununla birlikte, HC yönteminin de birden fazla yüz içeren fotoğraflarda %100 tarama yaparken daha uygun olduğu anlaşılmıştır. RGB resimlerdeki performans incelendiğinde, tek bir yüz bulunan resimlerde her iki yöntemin de benzer başarı gösterdiği ve başarılı sonuçlar ürettiği tespit edilmiştir. Ancak, birden fazla yüz içeren resimlerde her iki yöntem de %100 başarı göstermemiş, farklı kişilere ait yüzleri tespit ettikleri deney sonuçlarında ortaya koyulmuştur.

Bu bulgular doğrultusunda, RGB resimlerde yüz tespiti yapılacağı durumlarda her iki yöntemin hibrit bir yaklaşımla birlikte kullanılmasının uygun olabileceği önerilmektedir. Benzer şekilde, termal resimlerde gerçekleştirilecek görüntü analizleri için de bu hibrit yöntem tavsiye edilebilir. Çünkü HC yöntemi, tekil yüz tespitinde yüksek başarı gösterirken, derin öğrenme yöntemi çoklu yüz içeren resimlerde daha iyi sonuçlar vermektedir. Dolayısıyla, her iki yöntemin birlikte kullanıldığı hibrit metotların, özellikle termal görüntü analizlerinde başarı oranını artırabileceği değerlendirilmektedir.

5.1.2. Deep Learning Modeli Sonuçları

Derin öğrenme teknikleri kullanılarak yapılan termal yüz tanıma çalışmaları, farklı sınıflarda elde edilen performans metrikleriyle modelin güçlü ve zayıf yönlerini ortaya koymaktadır. Normal sınıfında, yüksek bir doğruluk (%76.9), duyarlılık (%90), ve F1-skoru (%85.7) değerleriyle model, tekil ve karşıdan çekilmiş yüzleri tanımda

oldukça başarılı olmuştur. Bu, modelin özellikle basit senaryolarda etkili bir şekilde çalıştığını göstermektedir.

Farklı Açık 1 (FAÇ) sınıfında, model %83.3 doğruluk ve %88.9 duyarlılık oranlarıyla en yüksek performansı sergilemiştir. Yan açılardan çekilmiş yüzleri doğru bir şekilde tanıyan model, bu sınıfta hem yanlış pozitif hem de yanlış negatif oranlarını düşük tutarak güçlü bir performans ortaya koymuştur.

Buna karşılık, Çoklu (Çok) sınıfında doğruluk oranı %69.2'ye düşmüş ve modelin karmaşık senaryolarda daha fazla zorlandığı gözlemlenmiştir. Düşük performans, birden fazla kişinin bulunduğu görüntülerde doğru sınıflandırma yapmanın model için daha zor olduğunu göstermektedir. Bu sınıfta daha yüksek yanlış pozitif ve yanlış negatif oranları, modelin iyileştirilmesi gerektiğini işaret etmektedir.

Genel olarak, derin öğrenme modeli, termal yüz tanıma konusunda başarılı bir performans sergilemiş ve en iyi sonuçları "Normal" ve "Farklı Açık 1" sınıflarında elde etmiştir. Ancak, "Çoklu" sınıfındaki düşük başarı, modelin karmaşık ve kalabalık senaryolar için optimize edilmesi gerektiğini ortaya koymaktadır. Bu doğrultuda, modelin performansını artırmak için veri setinin genişletilmesi, ek özellik mühendisliği yapılması ve daha gelişmiş algoritmaların kullanılması önerilmektedir.

Derin öğrenme modelleri, düşük çözünürlük veya düşük kontrast gibi zorlu termal görüntülerde bile üstün doğruluk oranlarıyla öne çıkmaktadır. Özellikle karmaşık desenleri öğrenme kapasiteleri sayesinde, yüz tespitinde geleneksel yöntemlere göre çok daha başarılı sonuçlar elde edilmektedir. Bu doğruluk, termal görüntüleme teknolojilerinin sağlık alanında kullanım potansiyelini artırmaktadır. Yan dönmüş, kısmen görünür veya düşük aydınlatma gibi karmaşık koşullarda derin öğrenme modelleri, yüz tespitinde başarılı sonuçlar sunmaktadır. Bu yetenek, termal görüntülerdeki belirsizliklerin daha iyi yönetilmesine olanak tanır ve gerçek dünya uygulamaları için önemli bir avantaj sağlar. Derin öğrenme tabanlı sistemler, yüz tespiti sonrası sıcaklık analizlerini entegre bir şekilde gerçekleştirebilir. Bu özellik, termal görüntülerden elde edilen verilerin anlamlandırılmasını kolaylaştırır ve özellikle ateşli

hastalıkların erken teşhisinde kritik bir rol oynar. Derin öğrenme modellerinde transfer öğrenimi, önceden eğitilmiş modellerin termal görüntüler üzerinde yeniden eğitilmesiyle daha kısa sürede etkili sonuçlar alınmasını sağlar. Bu yöntem, büyük veri ihtiyacını azaltırken, özelleşmiş termal veri kümelerinin etkin şekilde kullanılmasına olanak tanır. Derin öğrenme modelleri, farklı çevresel koşullar ve senaryolarda (örneğin, kalabalık ortamlar veya tekil yüz tespiti) yüksek performans göstermektedir. Bu durum, modellerin geniş bir uygulama yelpazesinde kullanılabilirliğini artırır ve pratik avantajlar sunar.

Bununla birlikte, derin öğrenme modellerinin bazı sınırlamaları da bulunmaktadır. Derin öğrenme modelleri, genellikle yüksek performanslı bilgisayarlara ve GPU desteğine ihtiyaç duyar. Bu durum, özellikle kaynakların kısıtlı olduğu ortamlarda modellerin kullanımını zorlaştırabilir ve maliyet açısından bir dezavantaj oluşturabilir. Bu modellerin başarılı bir şekilde eğitilebilmesi için büyük miktarda veri gerekmektedir. Termal görüntüler gibi spesifik veri kümelerinin oluşturulması ise hem zaman alıcı hem de maliyetlidir. Bu veri ihtiyacı, bazı projelerde modellerin uygulanabilirliğini sınırlandırabilir.

Derin öğrenme modelleri, kalabalık ortamlar veya düşük kontrastlı görüntüler gibi zorlayıcı koşullarda yanlış pozitifler üretebilir veya bazı yüzleri algılayamayabilir. Bu durum, modelin genel doğruluğunu etkileyen bir faktör olarak öne çıkmaktadır. Derin öğrenme modellerinin yüksek hesaplama gereksinimleri, gerçek zamanlı uygulamalarda performans sınırlamalarına neden olabilir. Bu tür uygulamalar için daha hızlı ve optimize edilmiş modellerin geliştirilmesi gereklidir. Termal görüntülerde sıcaklık tespiti, modelin çözünürlük kapasitesine ve veri ön işleme süreçlerindeki eksikliklere bağlı olarak hatalı sonuçlar üretebilir. Bu durum, sağlık alanında güvenilir sonuçların elde edilmesini zorlaştırabilir.

5.1.2. Büyük Dil Modeli (LLM) Sonuçları

Büyük Dil Modeli (LLM) kullanılarak yapılan termal yüz tanıma çalışmaları, farklı sınıflarda elde edilen sonuçlarla modelin performansını detaylı bir şekilde değerlendirmiştir. Normal (Nor) sınıfında, model %62.5 doğruluk, %75 kesinlik ve %75 duyarlılık oranlarıyla başarılı bir performans sergilemiş, F1-Skoru da %75 olarak

hesaplanmıştır. Bu sonuçlar, modelin normal yüzleri tanımada dengeli ve etkili bir şekilde çalıştığını göstermektedir. Ancak, Farklı Açı 1 (FAÇ) sınıfında, modelin performansı biraz düşmüştür. Doğruluk oranı %53.33, duyarlılık %57.14 ve kesinlik %66.67 olarak kaydedilmiştir. Bu sınıf için F1-Skoru %61.54 ile modelin ortalama bir performans sergilediğini ortaya koymaktadır. Modelin özellikle farklı açılardan çekilmiş yüzlerde hata yapma eğiliminde olduğu görülmektedir.

Bununla birlikte, Çoklu (Çok) sınıfında modelin performansı önemli ölçüde düşmüştür. Doğruluk oranı %33.33 ile oldukça düşük bir seviyede kalmış, kesinlik %60, duyarlılık ise %42.86 olarak hesaplanmıştır. F1-Skoru %50 olan bu sınıf, modelin karmaşık ve çok kişili fotoğrafları doğru bir şekilde sınıflandırmada ciddi zorluklar yaşadığını göstermektedir.

Genel olarak, LLM temelli model, normal yüzleri tanıma konusunda tatmin edici bir performans sergilerken, farklı açı ve çok kişili görüntülerde başarı oranı düşmektedir. Bu durum, modelin karmaşık senaryolarda daha iyi performans göstermesi için ek veri, daha gelişmiş algoritmalar ve özellik mühendisliği gibi iyileştirmelere ihtiyaç duyduğunu ortaya koymaktadır.

LLM'ler, yüz tespiti sonrası sıcaklık verilerini anlamlandırma ve raporlama açısından önemli bir role sahiptir. Özellikle doğal dil işleme yetenekleriyle, bu verilerin anlaşılır ve kullanıcı dostu bir şekilde sunulmasına olanak tanımaktadır. Ancak, LLM'ler yüz tespiti yapamadığı için Haar Cascade veya Deep Learning gibi yöntemlerle birlikte kullanılmaları gerekmektedir.

Eklenen resimlerden de anlaşılacağı üzere, LLM'ler analiz ve raporlama süreçlerinde etkili olsa da, çoklu kişi içeren görüntülerde tespitlerin sınırlı kaldığı durumlar olmuştur. Bu da doğru ve güvenilir sonuçlar için yüz tespiti algoritmalarının önemini bir kez daha göstermektedir. Gelecekte, LLM'lerin termal görüntüleme gibi özel alanlara yönelik daha fazla özelleştirilmesi, hem doğruluk oranlarının artmasına hem de daha anlamlı sonuçlar elde edilmesine katkı sağlayabilir.

5.1.3. Çoklu yüz tespitiinde yaşanan sorunlar

Termal görüntülerde birden fazla yüzün tespiti, yüzlerin yakınlığı, görüntüdeki açılar, termal görüntülerin kendine özgü özellikleri ve kullanılan algoritmaların sınırlamalarıyla ilgili çeşitli teknik zorluklar içermektedir. Yüzlerin üst üste geldiği veya birbirine yakın olduğu kalabalık ortamlarda, modellerin yüzleri doğru bir şekilde ayırt etmesi zorlaşır. Geometrik özelliklere dayanan Haar Cascade (HC) yöntemi, yüzlerin daha az belirgin olduğu durumlarda doğruluğunu kaybeder ve üst üste binen yüzleri tespit etmede önemli sınırlamalar gösterir (Şekil 6). Ayrıca karmaşıklık matrisleri, Haar Cascade modelinin çoklu yüz tespitiindeki düşük başarı oranını açıkça göstermektedir (Tablo 3). Bu başarısızlığın temel nedeni, modelin geometrik özelliklere dayalı bir algoritma olmasıdır. Bu durum, birden fazla yüzün yakın veya üst üste olduğu senaryolarda yüzlerin ayırt edilememesine yol açar ve doğru pozitif (DP) oranını düşürürken yanlış negatif (YN) oranını artırır. Ayrıca, çoklu yüz senaryosunda model, yüz benzeri desenleri veya yüzlerin birleşik görünümünü yüz olarak algılayarak yanlış pozitif (YP) sayısını artırır (Şekil 45-47-58 gibi). Bu durum, Haar Cascade'in dikkat dağıtıcı unsurlar karşısında sınırlı performans gösterdiğini ortaya koymaktadır (Şekil 28). Bununla beraber modelin eğitildiği veri setinin çoklu yüz senaryolarını yeterince temsil edememesi de başarı oranını düşüren bir diğer faktördür. Termal görüntülerdeki yüzlerin anatomik detaylarının az olması ve benzer sıcaklıktaki yüzlerin ayırt edilememesi de doğru tespiti zorlaştırarak hem DP oranını düşürür hem de YP oranını artırır. Sonuç olarak, Haar Cascade modeli, veri seti eksikliği ve algoritmik sınırlamaları nedeniyle çoklu yüz tespiti gibi karmaşık durumlarda düşük performans göstermektedir.

Derin öğrenme (DL) yöntemleri, bu tür senaryolarda daha yüksek doğruluk sağlasa da, eğitim veri setlerindeki çeşitlilik eksikliği, karmaşık durumlarda performans düşüşüne yol açmaktadır (Şekil 22). Ayrıca, karışıklık matrisi, derin öğrenme modelinin çoklu yüzleri tespit etmedeki bazı belirgin sınırlamalarını vurgulamaktadır. Model, çoklu yüz senaryolarında yüksek bir doğru pozitif (TP) oranı gösterse de, aynı zamanda önemli derecede yüksek yanlış pozitif (FP) ve yanlış negatif (FN) oranlarına sahiptir; bu da çoklu yüz tespitiinin zorluklarını ortaya koymaktadır (Tablo 5). DL'nin çoklu yüz tespitiindeki düşük başarı oranının birincil sebeplerinden biri, Haar Cascade (HC) yöntemine benzer şekilde, eğitim veri setlerinde yeterli çeşitlilik ve karmaşıklığın olmamasıdır. Eğitim sırasında, farklı pozisyonlarda, ölçeklerde ve açılarda çoklu yüzlerin temsil edilmemesi, modelin bu tür durumlara genelleme yapabilme yeteneğini

sınırlandırmaktadır. Tablo 5'teki 7'lik TP değeri, modelin çoklu yüzleri büyük ölçüde tespit edebildiğini gösterse de, her biri 2 olan FP ve FN değerleri, tespit edilen yüzlerin doğruluğunun yeterli olmadığını göstermektedir. Yüksek FP oranı, modelin çevredeki dikkat dağıtıcı öğeleri veya yüz benzeri özellikleri yüz olarak yanlış tanımasından kaynaklanmaktadır. Termal görüntülerde, belirgin sıcaklık farklılıklarının olmaması bu sorunu daha da kötüleştirir, modelin hassasiyetini azaltır ve FP sayısını artırır. Aksine, FN oranı, modelin bazı yüzleri tespit edememesini yansıtır; bu durum, özellikle yüzlerin örtüştüğü, açılarda değiştiği veya anatomik detayların net olmadığı termal görüntülerde belirginleşmektedir.

Büyük Dil Modelleri (LLM'ler), yüz tespiti için açıkça tasarlanmamış olmalarına rağmen, bağlamsal analizlerde destekleyici bir rol oynayabilirler. Tablo 7 ve 4.8, DL modelinin çoklu yüz tespitindeki düşük başarı oranını açıkça göstermektedir. Tablo 7'deki karışıklık matrisine göre, "Multi (Çoklu)" sınıfında yalnızca 3 doğru pozitif (TP) bulunmakta, 4 yanlış negatif (FN) ve 2 yanlış pozitif (FP) değerleri yer almaktadır. Bu, modelin birçok yüzü tespit edemediğini ve ayrıca yüz benzeri yapıları yüz olarak yanlış sınıflandırdığını göstermektedir. Özellikle termal görüntülerde, belirgin anatomik detayların ve sıcaklık farklılıklarının olmaması, modelin doğru tespit yapmasını daha zor hale getirmektedir. Tablo 8'deki performans metrikleri, "Multi (Çoklu)" sınıfının yalnızca %33.33 doğruluk, %60 hassasiyet, %42.86 geri çağırma ve %50 F1 puanı ile çalıştığını ortaya koymaktadır. Düşük geri çağırma oranı, modelin birçok yüzü tespit edemediğini ve bunun sonucunda yüksek FN oranı oluştuğunu göstermektedir. Hassasiyet görece yüksek olsa da, doğru pozitif tespitlerde kısmi başarıyı işaret etmektedir, ancak genel model performansı tatmin edici seviyede değildir. Bu düşük başarı oranının temel nedeni, eğitim veri setindeki çeşitlilik eksikliğidir. Özellikle farklı açılarda, ölçeklerin ve yakın konumlanmış yüzlerin yer aldığı çoklu yüz senaryoları yeterince temsil edilmemektedir. Bu sınırlama, modelin bu tür karmaşık senaryolara genelleme yapma kapasitesini kısıtlamaktadır. Ayrıca, termal görüntülerde yüz ve arka plan sıcaklıklarının benzerliği ile belirgin detayların eksikliği, FP oranlarını artırmakta ve yüzlerin doğru bir şekilde tanımlanmasını zorlaştırmaktadır.

Ayrıca, modellerin performansı, yüzlerin farklı açılardan görüldüğü veya kısmen görüldüğü durumlarda önemli ölçüde etkilenir. HC yöntemi, özellikle 45 derece veya

daha büyük açılarda zorlanır ve bu durumlarda doğruluğunu kaybeder (Şekil 6). DL modelleri, bu tür durumları ele alırken daha sağlam bir performans sergiler; ancak başarıları, eğitim veri setlerinin bu senaryoları ne kadar kapsadığına bağlıdır (Şekil 22). Termal görüntülerde yüzlerin belirgin anatomik detaylarının olmaması, tespitlerde doğruluk kaybına yol açar. Ayrıca, yüzler arasındaki sıcaklık farklarının minimal olduğu durumlarda hem HC hem de DL yöntemlerinin tespit yetenekleri sınırlanabilir.

Bir diğer kritik sorun, birden fazla yüzü tespit etmek için kullanılan veri setlerinin genel olarak yetersizliği ve sınırlı senaryoları kapsamasıdır. HC yöntemi, statik algoritmaları nedeniyle bu eksiklikten en çok etkilenen yöntemdir. DL modelleri, daha geniş ve çeşitli veri setleri ile daha iyi performans sergilerken, bu veri setlerini genişletmek, karmaşık senaryoları etkili bir şekilde ele almak için gereklidir. Çevresel faktörler ve termal görüntülerdeki gürültü de modellerin doğruluğunu olumsuz etkiler. Özellikle, arka plan sıcaklıklarının yüz sıcaklıklarına benzer olduğu durumlarda tespit hataları artar.

5.2. Tartışma

5.2.1 Haarcascade Yönteminin Performans Tartışması

Haarcascade yöntemi, termal görüntülerdeki yüz tespitinde genellikle tatmin edici sonuçlar elde edilmesini sağlamıştır, özellikle tekil yüzlerin bulunduğu durumlarda yüksek doğruluk oranları gözlemlenmiştir (Peng et al., 2022). Bununla birlikte, çoklu yüzlerin bulunduğu görüntülerde bu doğruluk oranı düşüş göstermiştir. Özellikle yüz açılarının geniş olduğu durumlarda (örneğin, 45° ve üzeri), Haarcascade yönteminin performansı belirgin şekilde zayıflamaktadır (Vinnil et al., 2021). Bu sınırlamalar, yöntemin algoritmik yapısının yüz konumlarını ve açılarını yeterince esnek bir şekilde modelleyememesiyle açıklanabilir. Ancak, bu yöntem düşük maliyetli ve hızlı bir çözüm sunduğundan, düşük bütçeli uygulamalar için uygun olabilmektedir.

Sıcaklık analizi açısından, Haarcascade yöntemi $\pm 0.1^{\circ}\text{C}$ hata payı ile oldukça tatmin edici sonuçlar üretmiştir (Saha & Neogi, 2021). Ancak çoklu yüzlerin bulunduğu durumlarda sıcaklık ölçümleri karışıklık gösterebilmektedir, bu da her yüzün sıcaklığının bağımsız olarak doğru bir şekilde belirlenmesini zorlaştırmaktadır. Bu sınırlamanın aşılması için daha geniş ve çeşitli veri setleriyle çalışmak, doğruluk oranlarını artırabilir (Khan et al., 2020).

5.2.2 Derin Öğrenme Yönteminin Performans Tartışması

Derin öğrenme (DL) tabanlı yöntemler, yüz tespiti ve sıcaklık analizi açısından Haarcascade yöntemine kıyasla belirgin şekilde daha yüksek performans göstermektedir. DL modelleri, hem tekil yüzlerin hem de çoklu yüzlerin bulunduğu görüntülerde yüksek doğruluk oranlarına ulaşmıştır (Abbas & Park, 2020). Özellikle farklı yüz açılarıyla karşılaşıldığında, DL yöntemleri yüz tanıma doğruluğunda ciddi bir düşüş yaşamamaktadır (Xu et al., 2021). Bu, DL modellerinin yüz tespiti açısından esnekliğini ve adaptasyon yeteneğini ortaya koymaktadır.

Sıcaklık analizi açısından DL yöntemleri, $\pm 0.05^{\circ}\text{C}$ gibi düşük hata payları ile oldukça hassas sonuçlar üretmiştir (Mohammed & El-Baz, 2019). Ayrıca, çoklu yüzlerin bulunduğu görüntülerde bile her bir yüzün sıcaklık ölçümleri doğru bir şekilde ayrıştırılabilmektedir (Peng et al., 2022). Bu, DL yöntemlerinin termal görüntülerdeki yüzlerin bireysel sıcaklık analizine yönelik güçlü bir kapasiteye sahip olduğunu göstermektedir.

5.2.3 Large Language Models (LLM) Performans Tartışması

LLM yöntemleri, yüz tespiti konusunda derin öğrenme modellerine yakın bir performans sergilemiş olsa da, özellikle çoklu yüzlerin bulunduğu durumlarda oldukça etkili sonuçlar üretmiştir. LLM modelleri, yüz tespiti sırasında, termal görüntülerdeki sıcaklık dağılımını ve yüz ifadelerinin bağlamsal anlamlarını çözümlemeye başarılı olmuştur (Xu et al., 2021). Bu, LLM'lerin yalnızca görsel veriyi analiz etmekle kalmayıp, aynı zamanda metinsel verileri de kullanarak daha kapsamlı bir analiz yapabilme yeteneğini ortaya koymaktadır.

Sıcaklık analizi açısından ise, LLM modelleri termal görüntülerde bağlamsal analizler yaparak yeni bir yaklaşım geliştirmiştir. Ancak bireysel yüzlerin sıcaklık ölçümlerinde, DL yöntemlerine kıyasla biraz daha düşük hassasiyet gösterdiği gözlemlenmiştir (Saha & Neogi, 2021). Yine de, LLM'lerin sıcaklık dağılımlarını daha geniş bir bağlamda analiz edebilmesi, bu yöntemleri özellikle bağlamsal sıcaklık analizi gerektiren uygulamalar için uygun hale getirmektedir.

5.2.4 Tekil ve Çoklu Yüzlerin Değerlendirilmesi

Tekil yüzlerin bulunduğu görüntülerde, Haarcascade, DL ve LLM yöntemleri genel olarak tatmin edici sonuçlar üretmiştir. Ancak DL yöntemleri, doğruluk ve hassasiyet açısından belirgin bir üstünlük göstermektedir (Khan et al., 2020). Özellikle yüzün pozisyonu ve açısı konusunda daha esnek bir modelleme yapabilen DL yöntemleri, tekil yüzlerde daha doğru sonuçlar elde edilmesine olanak tanımaktadır.

Çoklu yüzlerin bulunduğu görüntülerde ise, Haarcascade yönteminin doğruluk oranı sınırlı kalmıştır. DL ve LLM yöntemleri, çoklu yüzlerin tespiti ve her bir yüzün sıcaklık analizini daha başarılı bir şekilde gerçekleştirebilmiştir (Xu et al., 2021). Bu, her iki yöntemin, yüzlerin yerini ve açısını daha doğru bir şekilde tanıyabilme yeteneğinden kaynaklanmaktadır.

5.2.5 Genel Değerlendirme

Genel olarak değerlendirildiğinde, derin öğrenme yöntemleri termal görüntülerden yüz tespiti ve sıcaklık analizi yapmak için en yüksek doğruluk ve hassasiyet oranını elde etmiştir, özellikle çoklu yüzlerin analizinde üstün performans sergilemiştir (Abbas & Park, 2020; Peng et al., 2022). Haarcascade yöntemi, hızlı ve düşük maliyetli çözümler sunduğu için belirli uygulamalar için uygun olabilir. Ancak, çoklu yüzlerin bulunduğu durumlarda sınırlı doğruluk sağladığı unutulmamalıdır. LLM modelleri ise, bağlamsal analizler ve sıcaklık dağılımlarının daha kapsamlı bir şekilde değerlendirilmesinde önemli bir potansiyele sahiptir, ancak bireysel yüz sıcaklık ölçümlerinde daha düşük hassasiyet göstermektedir (Saha & Neogi, 2021). Gelecekte, her bir yöntemin performansını artırmak amacıyla daha geniş ve çeşitlendirilmiş veri setleri ile yapılacak çalışmaların faydalı olacağı düşünülmektedir.

5.3. Öneriler

1. Daha fazla parametre kombinasyonu test edilerek algoritmanın performansı optimize edilmelidir.
2. Termal görüntülerde kontrastı artırmak için ön işleme tekniklerinden yararlanılabilir.
3. Haar Cascade algoritmasının performansını artırmak için derin öğrenme tabanlı yöntemlerle veya hibrit modellerle birleştirilmesi önerilmektedir.
4. Sıcaklık analizi sonuçlarının büyük dil modelleri (LLM) gibi gelişmiş analiz araçlarıyla entegre edilmesi, verilerin daha anlamlı hale getirilmesini ve raporlanmasını kolaylaştıracaktır.
5. Derin öğrenme yöntemleri, termal görüntülerde yüz tespiti ve sıcaklık analizi için etkili bir çözüm sunmakta, özellikle yan dönmüş yüzler ve düşük kontrastlı görüntülerde yüksek doğruluk sağlamaktadır.
6. Haar Cascade gibi geleneksel yöntemlere kıyasla, doğru pozitif oranları daha yüksek olan bu modeller, daha güvenilir sonuçlar elde edilmesine olanak tanımaktadır.
7. Transfer öğrenimi yöntemlerinin uygulanması, modellerin eğitim sürecini hızlandırmakta ve maliyetleri düşürmektedir.
8. Gerçek zamanlı uygulamalarda kullanılabilirliklerini artırmak için derin öğrenme modellerinin optimize edilmesi ve işlem hızlarının geliştirilmesi gerekmektedir.
9. Termal görüntülerden elde edilen sıcaklık analizlerinin doğruluğunu artırmak için ön işleme tekniklerinin ve daha gelişmiş analiz araçlarının kullanılması önerilmektedir.
10. Mevcut veri artırma tekniklerine ek olarak aydınlatma, kontrast ayarları ve sentetik veri üretimi gibi yöntemler uygulanabilir. Bu, modelin daha geniş bir varyasyon spektrumunu öğrenmesini sağlayacaktır.
11. Modellerin özellikle karmaşık sınıflar için performansını artırmak amacıyla daha derin ve karmaşık bir ağ mimarisi kullanılabilir. Transformer tabanlı modeller, LLM'nin mevcut yapısına eklenerek daha iyi sonuçlar elde edilebilir.

12. Modellerin karmařık durumlarda daha iyi performans göstermesi için, yüz bölgelerinden çıkarılan termal öznitelikler optimize edilmeli ve daha anlamlı özelliklerin belirlenmesi sağlanmalıdır.
13. Modellerin başarısız olduđu örnekler detaylı bir şekilde analiz edilerek, bu hataların nedenleri incelenmeli ve modelin bu durumlara karşı daha dayanıklı olması sağlanmalıdır.



KAYNAKÇA

- Akın, E. ve Şahin, M. E. (2023). Derin öğrenme ve yapay sinir ağı modelleri üzerine bir inceleme. *EMO Bilimsel Dergi*, 14(1), 27-38.
- Akkuş, M. S. (2020). Koronavirüs ve COVID-19. *Aksaray University Journal of Medical Sciences*, 1(2), 8-12.
- Alimovski, E. (2019). Derin öğrenmeye dayalı güçlü yüz tanıma sistemi için GAN ile veri çoğaltma. Yayınlanmamış Yüksek Lisans Tezi. İstanbul Sabahattin Zaim Üniversitesi, İstanbul.
- Altan, G. (2019). DeepGraphNet: Grafiklerin sınıflandırılmasında derin öğrenme modelleri. *Avrupa Bilim ve Teknoloji Dergisi, (Özel Sayı)*, 319-329.
- Araujo, J. M. A., de Moura, A. C. E., da Silva, S. L. B., Holanda, M., de Oliveira Ribeiro, E., ve da Silva, G. L. (2021, June). Comparative performance analysis of NoSQL Cassandra and MongoDB databases. In *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1-6). IEEE.
- Arı, A., ve Berberler, M. E. (2017). Yapay sinir ağları ile tahmin ve sınıflandırma problemlerinin çözümü için arayüz tasarımı. *Acta Infologica*, 1(2), 55-73.
- Atalay, M., ve Çelik, E. (2017). Büyük veri analizinde yapay zekâ ve makine öğrenmesi uygulamaları. *Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 9(22), 155-172.
- Atasoy, N. A., ve Tabak, D. (2018). Destek vektör makineleri kullanarak yüz tanıma uygulaması geliştirilmesi. *Engineering Sciences*, 13(2), 119-127.
- Aydın, M. M. (2023). X-Ray görüntüleri ile derin öğrenme kullanılarak COVID-19 vakalarının tespiti. Yayınlanmamış Yüksek Lisans Tezi. Sakarya Uygulamalı Bilimler Üniversitesi. Sakarya.
- Banker, K., Garrett, D., Bakkum, P., ve Verch, S. (2016). *MongoDB in action: Covers MongoDB version 3.0*. Simon and Schuster.
- Bektaş, E. (2018). Yapay sinir ağları yaklaşımı ile curufların termal iletkenlik değerlerinin tahmin edilmesi. Yayınlanmamış Yüksek Lisans Tezi. İstanbul Teknik Üniversitesi, İstanbul.

- Bengio, Y. (2009). *Learning deep architectures for AI*. Now Publishers Inc.
- Boicea, A., Radulescu, F., ve Agapin, L. I. (2012, September). MongoDB vs Oracle database comparison. In *2012 Third International Conference on Emerging Intelligent Data and Web Technologies* (pp. 330-335). IEEE.
- Bwire, G. M., ve Paulo, L. S. (2020). Coronavirus disease-2019: Is fever an adequate screening for the returning travelers? *Tropical Medicine and Health*, 48(1), 10–12. <https://doi.org/10.1186/s41182-020-00201-2>
- Bwire, G. M., ve Paulo, L. S. (2020). Coronavirus Disease-2019: Is Fever an Adequate Screening for the Returning Travelers? *Tropical Medicine and Health*, 48(1), 10–12. <https://doi.org/10.1186/s41182-020-00201-2>
- Cai, Q., Huang, D., Ou, P., Yu, H., Zhu, Z., Xia, Z., ... & Chen, J. (2020). COVID-19 in a designated infectious diseases hospital outside Hubei Province, China. *Allergy*, 75(7), 1742-1752. <https://doi.org/10.1111/all.14309>
- Chan, L.-S., Cheung, G. T. Y., Lauder, I. J., ve Kumana, C. R. (2006). Screening for fever by remote-sensing infrared thermographic camera. *Journal of Travel Medicine*, 11(5), 273–279. <https://doi.org/10.1111/j.1708-8305.2006.00072.x>
- Çalışan, M., ve Türkoğlu, İ. (2011). Termal kameralar ve uygulamaları. *Elektrik-Elektronik ve Bilgisayar Sempozyumu 2011*, 1-5.
- Daşdemir, Y., ve Kara, B. C. (2019). Farklı iş yükleri altında NoSQL sistemlerinin performans analizi. *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 8(4), 1466-1477.
- Demir, H., ve Savaş, S. (2023). Class attendance system in education with deep learning method. *arXiv Preprint*. <https://arxiv.org/abs/2309.13317>
- Djajakusumah, T. M., Candrawinata, V. S., Ho, J. P., Herman, H., Lukman, K., ve Lesmana, R. (2023). The predictive value of infrared thermal imaging (IRT) for peripheral artery disease: A systematic review. *Medicine (Baltimore)*, 102(43), 35639. <https://doi.org/10.1097/MD.00000000000035639>
- Doğan, F., ve Türkoğlu, İ. (2019). Derin öğrenme modelleri ve uygulama alanlarına ilişkin bir derleme. *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, 10(2), 409-445.

- Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, 55(10), 78-87. <https://doi.org/10.1145/2347736.2347755>
- Dong, W., Fu, F., Shi, G., Cao, X., Wu, J., Li, G., ve Li, X. (2016). Hyperspectral image super-resolution via non-negative structured sparse representation. *IEEE Transactions on Image Processing*, 25(5), 2337–2352. <https://doi.org/10.1109/TIP.2016.2534380>
- Dong, W., Fu, F., Shi, G., Cao, X., Wu, J., Li, G., ve Li, X. (2016). Hyperspectral image super-resolution via non-negative structured sparse representation. *IEEE Transactions on Image Processing*, 25(5), 2337–2352. <https://doi.org/10.1109/TIP.2016.2545274>
- Eddie, Y. K., Kawb, G. J. L., ve Chang, W. M. (2004). Analysis of IR thermal imager for mass blind fever screening. *Microvascular Research*, 68(2), 104–109. <https://doi.org/10.1016/j.mvr.2004.05.003>
- Faundez-Zanuy, M., ve Mekyska, J. (2011). On the focusing of thermal images. *Pattern Recognition Letters*, 32, 1548–1557. <https://doi.org/10.1016/j.patrec.2011.06.001>
- Forsyth, D. A., ve Ponce, J. (2012). *Computer vision: A modern approach*. Pearson.
- Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4), 193–202. <https://doi.org/10.1007/BF00344251>
- George, M. B., ve Paulo, L. S. (2020). Coronavirus disease-2019: Is fever an adequate screening for the returning travelers? *Tropical Medicine and Health*, 48(1), 10–12. <https://doi.org/10.1186/s41182-020-00201-2>
- George, M. B., ve Paulo, L. S. (2020). Coronavirus Disease-2019: Is Fever an Adequate Screening for the Returning Travelers? *Tropical Medicine and Health*, 48(1), 10–12. <https://doi.org/10.1186/s41182-020-00201-2>
- Gu, Y., et al. (2020). MedSRGAN: Medical images super-resolution using generative adversarial networks. *Multimedia Tools and Applications*, 79, 21815–21840. <https://doi.org/10.1007/s11042-020-09568-y>
- Gümüő, H. T., ve Eyüpoğlu, C. (2023). Grafik sinir ağılarına genel bir bakış. *EMO Bilimsel Dergi*, 13(2), 39-56.

- Hannah, W., Daniel, K., ve Saskia, S. (2019). A literature review on machine learning in supply chain management. *International Conference of Logistics (HICL)*, 27, 413-441.
- Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A. R., Jaitly, N., ... & Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6), 82-97.
- Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., Zhang, L., et al. (2020). Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *The Lancet*, 395(10223), 497–506. [https://doi.org/10.1016/S0140-6736\(20\)30183-5](https://doi.org/10.1016/S0140-6736(20)30183-5)
- Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., Zhang, L., et al. (2020). Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *The Lancet*, 395(10223), 497–506. [https://doi.org/10.1016/S0140-6736\(20\)30183-5](https://doi.org/10.1016/S0140-6736(20)30183-5)
- Hubel, D., ve Wiesel, T. (1968). Receptive fields and functional architecture of monkey striate cortex. *Journal of Physiology*, 195(1), 215–243.
- Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., ve Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv preprint arXiv:1602.07360*.
- Ignatov, I., Mosin, O., Niggli, H., Drossinakis, C., ve Tyminski, G. (2015). Methods for registering non-ionizing radiation emitted from the human body. *European Review of Chemical Research*, 3(1), 4–24. <https://doi.org/10.13187/ercr.2015.3.4>
- Ivakhnenko, A. G., ve Lapa, V. G. (1966). *Cybernetic predicting devices*.
- İlçe, A., ve Demir, B. (2024). Kızılötesi termal kameranın sağlık alanlarında kullanımı. *Sağlık Bakım ve Rehabilitasyon Dergisi*, 3(1), 12–25.
- İlçe, A., ve Demir, B. (2024). Kızılötesi Termal Kameranın Sağlık Alanlarında Kullanımı. *Sağlık Bakım ve Rehabilitasyon Dergisi*, 3(1), 12-25.
- İlçe, A., ve Demir, B. (2024). Kızılötesi termal kameranın sağlık alanlarında kullanımı. *Sağlık Bakım ve Rehabilitasyon Dergisi*, 3(1), 12-25.
- İnce, H., İmamoğlu, S. E., ve İmamoğlu, S. Z. (2021). Yapay zeka uygulamalarının karar verme üzerine etkileri: Kavramsal bir çalışma. *International Review of*

- Jain, A. K. (1989). *Fundamentals of digital image processing*. Prentice Hall.
- Jiang, H., ve Learned-Miller, E. (2017, May). Face detection with the faster R-CNN. In 2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017) (pp. 650-657). IEEE.
- Karadağ, N., Çetinkaya, A., ve Aydın, H. (2020). Yerel ikili desenler histogramları ile Covid-19 tanılı kişiler üzerinde kimlik analizi ve bildiri sistemi. *Gazi Mühendislik Bilimleri Dergisi*, 6(3), 172-183.
- Kayalı, N. Z., ve Omurca, S. İ. (2021). Konvolüsyonel sinir ağları (CNN) ile Çin sayı örüntülerinin sınıflandırması. *Computer Science, (Special)*, 184-191.
- Kekül, H., Bircan, H., ve Arslan, H. (2018). Yüz tanıma uygulamalarında özyüzler ve yapay sinir ağlarının karşılaştırılması. *Uluslararası Yönetim Bilişim Sistemleri ve Bilgisayar Bilimleri Dergisi*, 2(1), 51-59.
- Kim, J., Lee, J. K., ve Lee, K. M. (2016). Accurate image super-resolution using very deep convolutional networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1646–1654. <https://doi.org/10.1109/CVPR.2016.182>
- Kim, J., Lee, J. K., ve Lee, K. M. (2016). Accurate image superresolution using very deep convolutional networks. *IEEE CVPR*, 1646–1654. <https://doi.org/10.1109/CVPR.2016.182>
- Kirby, M., ve Sirovich, L. (1990). Application of the Karhunen-Loeve procedure for the human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1), 103-108.
- Koçer, B. (2012). Transfer öğrenmede yeni yaklaşımlar. Yayınlanmamış Doktora Tezi. Selçuk Üniversitesi, Konya.
- Kolakowski, P. (2020, December 3). 10 most popular programming languages on GitHub. *GitHub Language Rankings, 2018-2020*.
- Kosikowski, M., ve Suszynski, Z. (2011). Processing and recognition of the thermal images using wavelet transforms. *Microelectronics Reliability*, 51(11), 1271–1275. <https://doi.org/10.1016/j.microrel.2011.06.073>

- Kumar, A., Gupta, P. K., ve Srivastava, A. (2020). A review of modern technologies for tackling COVID-19 pandemic. *Diabetes and Metabolic Syndrome: Clinical Research and Reviews*, 14(4), 569–573. <https://doi.org/10.1016/j.dsx.2020.03.007>
- LeCun, Y., Bottou, L., Bengio, Y., ve Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324. <https://doi.org/10.1109/5.726791>
- Lemenkova, P. (2020). Python libraries matplotlib, seaborn and pandas for visualization geospatial datasets generated by QGIS. *Analele Științifice ale Universității "Alexandru Ioan Cuza" din Iași - Seria Geografie*, 64(1), 13-32.
- Li, D., Menassa, C. C., ve Kamat, V. R. (2018). Non-intrusive interpretation of human thermal comfort through analysis of facial infrared thermography. *Energy and Buildings*, 176, 246-261.
- Lim, B., Son, S., Kim, H., Nah, S., ve Lee, K. M. (2017). Enhanced deep residual networks for single image super-resolution. *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1132–1140. <https://doi.org/10.1109/CVPRW.2017.00143>
- Liu, Z., Luo, P., Wang, X., ve Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 3730-3738).
- Lutz, M. (2010). *Programming Python: Powerful object-oriented programming*. O'Reilly Media, Inc.
- Maghdid, H. S., Ghafoor, K. Z., Sadiq, A. S., Curran, K., ve Rabie, K. (2020). A novel AI-enabled framework to diagnose coronavirus COVID-19 using smartphone embedded sensors: Design study. *arXiv preprint arXiv:2005.00504*.
- Mamak, U., Konyar, M. Z., Solak, S., ve Uçar, M. H. (2020). Gerçek zamanlı yüz tanıma tabanlı personel kontrol ve takip sistemi tasarımı. *Avrupa Bilim ve Teknoloji Dergisi*, (19), 497-504.
- Marsland, S. (2015). *Machine learning: An algorithmic perspective*. CRC Press.
- McCulloch, W. S., ve Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>

- McCulloch, W. S., ve Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>
- Min, H. (2010). Artificial intelligence in supply chain management: Theory and applications. *International Journal of Logistics: Research and Applications*, 13(1), 13-39. <https://doi.org/10.1080/13675560903555174>
- Mohammed, M. N., Syamsudin, H., Al-Zubaidi, S., Sairah, A. K., Ramli, R., ve Yusuf, E. (2020). Novel Covid-19 detection and diagnosis system using IoT based smart helmet. *International Journal of Psychosocial Rehabilitation*, 24(7), 2296–2303. <https://doi.org/10.37200/IJPR/V24I7/PR270973>
- Mundial, I. Q., Hassan, M. S. U., Tiwana, M. I., Qureshi, W. S., ve Alanazi, E. (2020, September). Towards facial recognition problem in COVID-19 pandemic. In *2020 4rd International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)* (pp. 210-214). IEEE.
- Özdemir, R., ve Koç, M., (2019). Yeni bir veri kümesi (RidNet) kullanarak kontrolsüz ortamda yüz ifadesi tanınmanın derin öğrenme yöntemleri ile iyileştirilmesi. *Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Dergisi*, 6(2), 384-396.
- Öztürk, K., ve Şahin, M. E. (2018). Yapay sinir ağları ve yapay zekâ'ya genel bir bakış. *Takvim-i Vekayi*, 6(2), 25-36.
- Öztürk, M., ve Kasap, H. (2021). Multi-class brain normality and abnormality diagnosis using modified Faster R-CNN. *International Journal of Medical Informatics*, 155, 104576. <https://doi.org/10.1016/j.ijmedinf.2021.104576>
- Pérez, L., ve Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*.
- Rodriguez, Y., ve Marcel, S. (2006). Face authentication using adapted local binary pattern histograms. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part IV* (pp. 321-332). Springer Berlin Heidelberg.
- Ranjan, R., Patel, V. M., ve Chellappa, R. (2017). Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE transactions on pattern analysis and machine intelligence*, 41(1), 121-135.

- Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 3(3), 210-229. <https://doi.org/10.1147/rd.33.0210>
- Saran, M., ve Saran, N. (2019). Cassandra ve MongoDB NoSQL veri tabanlarının karşılaştırmalı güvenlik analizi. *Uluslararası Bilgi Güvenliği Mühendisliği Dergisi*, 5(2), 1-11.
- Savaş, B. K., İlkin, S., ve Becerikli, Y. (2016, May). The realization of face detection and fullness detection in medium by using Haar Cascade Classifiers. In *2016 24th Signal Processing and Communication Application Conference (SIU)* (pp. 2217-2220). IEEE.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85–117. <https://doi.org/10.1016/j.neunet.2014.09.003>
- Schroff, F., Kalenichenko, D., ve Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815-823). <https://doi.org/10.1109/CVPR.2015.7298682>
- Schroff, F., Kalenichenko, D., ve Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815–823). <https://doi.org/10.1109/CVPR.2015.7298682>
- Simonyan, K., ve Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Sun, G., Nakayama, Y., Dagdanpurev, S., Abe, S., Nishimura, H., Kirimoto, T., ve Matsui, T. (2017). Remote sensing of multiple vital signs using a CMOS camera-equipped infrared thermography system and its clinical application in rapidly screening patients with suspected infectious diseases. *International Journal of Infectious Diseases*, 55, 113-117.
- Szegedy, C., Ioffe, S., Vanhoucke, V., ve Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. *Thirty-first AAAI conference on artificial intelligence*.
- Şeker, A., Diri, B., ve Balık, H. H. (2017). Derin öğrenme yöntemleri ve uygulamaları hakkında bir inceleme. *Gazi Mühendislik Bilimleri Dergisi*, 3(3), 47-64.

- Şenalp, F. M., ve Ceylan, M. (2023). Termal yüz görüntülerinden oluşan yeni bir veri seti için derin öğrenme tabanlı süper çözünürlük uygulaması. *Journal of Polytechnic*, 26(2), 711–720. <https://doi.org/10.2339/jpol.26.2.711>
- Tan, F. G., Yüksel, A. S., Aydemir, E., ve Ersoy, M. (2021). Derin öğrenme teknikleri ile nesne tespiti ve takibi üzerine bir inceleme. *Avrupa Bilim ve Teknoloji Dergisi*, (25), 159-171.
- Taylan, E. (2014). ABD'deki üniversitelerde en çok öğretilen programlama dili: Python. *Erişim tarihi: 19-10-2021*.
- Terzopoulos, D., ve Waters, K. (1990, January). Analysis of facial images using physical and anatomical models. In *Proceedings Third International Conference on Computer Vision* (pp. 727-728). IEEE Computer Society.
- Uyar, K. (2022). Konvolüsyonel sinir ağlarında ağ eğitiminin iyileştirilmesi (Doctoral dissertation). Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, Turkey.
- Uyar, K. (2022). Konvolüsyonel Sinir Ağlarında Ağ Eğitiminin İyileştirilmesi. Yayınlanmamış Doktora Tezi. Selçuk Üniversitesi, Konya.
- Ünver, M., ve Altunok, C. (2020). Medikal endüstride yapay zeka ve uzman sistemlerin sürekli iyileştirmeye etkisi. *8th International Symposium on Innovative Technologies in Engineering and Science (ISITES)*, 482-490.
- Van Natta, M., Chen, P., Herbek, S., Jain, R., Kastelic, N., Katz, E., ve Vattikonda, N. (2020). The rise and regulation of thermal facial recognition technology during the COVID-19 pandemic. *Journal of Law and the Biosciences*, 7(1), lsa038.
- Vansh, V., Chandrasekhar, K., Anil, C. R., ve Sahu, S. S. (2020). Improved face detection using YCbCr and Adaboost. In *Computational Intelligence in Data Mining: Proceedings of the International Conference on ICCIDM 2018* (pp. 689-699). Springer Singapore.
- Viola, P., ve Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the Conference on Computer Vision and Pattern Recognition*.
- Wang, Y., Wang, Y., Li, H., Cai, Z., Tang, X., ve Yang, Y. (2020). CNN Hyperparameter Optimization Based on CNN Visualization and Perception Hash Algorithm. In *2020 19th International Symposium on Distributed Computing and*

Applications for Business Engineering and Science (DCABES) (pp. 78–82).
<https://doi.org/10.1109/DCABES50087.2020.00027>

Wong, S. C., Gatt, A., Stamatescu, V. ve McDonnell, M. D., 2016, Understanding Data Augmentation for Classification: When to Warp?, 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), 1-6.

Xu, Y., Jia, R., Mou, L., Li, G., Chen, Y., Lu, Y. ve Jin, Z., 2016, Improved relation classification by deep recurrent neural networks with data augmentation, COLING.

Wu, Z., ve McGoogan, J. M. (2020). Characteristics of and important lessons from the coronavirus disease 2019 (COVID-19) outbreak in China: Summary of a report of 72,314 cases from the Chinese Center for Disease Control and Prevention. *JAMA*, 323(13), 1239-1242. <https://doi.org/10.1001/jama.2020.2648>

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8(3), 338–353.
[https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X)

Yang, S., Luo, P., Loy, C. C., ve Tang, X. (2015). From facial parts responses to face detection: A deep learning approach. In Proceedings of the IEEE international conference on computer vision (pp. 3676-3684).

Zhang, X., Li, C., Meng, Q., Liu, S., Zhang, Y., ve Wang, J. (2018). Infrared image super resolution by combining compressive sensing and deep learning. *Sensors*, 18(8), 2587. <https://doi.org/10.3390/s18082587>

Zhou, Y., Yang, Y., Huang, J., Jiang, S., ve Du, L. (2019). Advances in MERS-CoV vaccines and therapeutics based on the receptor-binding domain. *Viruses*, 11(1).
<https://doi.org/10.3390/v11010014>

URL-1 (2024) Face Detection using Haar Cascades, OpenCV, Erişim: 11.10.2024, https://docs.opencv.org/4.x/d2/d99/tutorial_js_face_detection.html

URL-2 (2024). Python – Downloads, Erişim. 11.10.2024 <https://www.python.org/>

URL-3 (2024). PowerShell Gallery, Erişim: 11.10.2024
<https://www.powershellgallery.com/>

URL-4 (2024). Visual Studio Code, Erişim: 11.10.2024, <https://code.visualstudio.com/>

URL-5 (2024). SmartView® PC ve Mobil Uygulama IR analiz/raporlama yazılımı,
Erişim: 11.10.2024 <https://www.fluke.com/tr-tr/urun/fluke-yazilimi/fluke-smartview-ir-mobile>

URL-6 (2024). Yapay Zeka, Makine Öğrenimi ve Derin Öğrenme Arasındaki Farklar,
Endüstri 4.0 Platformu, Erişim: 08.11.2024 <https://www.endustri40.com/yapay-zeka-makine-ogrenimi-ve-derin-ogrenme-arasindaki-farklar/>



EKLER

Ek 1: Haar Cascade Yüz Tespiti

```
import cv2

print("[BİLGİ] Haar Cascade yüz tespit edici'yi yüklüyor...")

detector = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")

imgName = "IR_00269.png" #IR ve Rgb Resim için isim girilir
imgName, imgExtension = imgName.split('.')
img = cv2.imread(f"image/{imgName}.{imgExtension}")

# Haar Cascade gri tonlu resimler üzerinde çalıştığından renk uzayı
dönüşümü yapalım
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) # RGB uzaydan gri tonlu
uzaya dönüşüm

print("[BİLGİ] Yüz tespiti gerçekleştiriliyor...")

scaleFactor, minNeighbors = 1.05, 25 # iki adet ayarlanabilir
parametre

rects = detector.detectMultiScale(gray, scaleFactor, minNeighbors)
print(f"[BİLGİ] {len(rects)} adet yüz tespit edildi.")

# tespit edilen yüzleri resim üzerinde dikdörtgen olarak göster
for (x,y,w,h) in rects:
    cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0), 9)

s = 0.50 # s=(0-1] resmi ekranda görüntülemek için ölçeklendir
ring = cv2.resize(img, (int(s*img.shape[1]), int(s*img.shape[0])),
0)

cv2.imshow('Face detection with Haar Cascade', ring)

cv2.waitKey(0)

cv2.imwrite(f"result/{imgName}_scaleFactor_{scaleFactor}_minNeighbor
s_{minNeighbors}.jpg",
img, [cv2.IMWRITE_JPEG_QUALITY, 50])
```

Ek 2: Deep Learning Yüz Tespiti

```
import numpy as np
import cv2

# Hangi fotoğrafın üzerinde çalışacağız
name = "IR_00269"

# Derin Öğrenme ile Yüz Tespiti (Face Detection with Deep Learning)
print("[BİLGİ] Derin öğrenme modeli yükleniyor...")
net = cv2.dnn.readNetFromCaffe('deploy.prototxt.txt',
'res10_300x300_ssd_iter_140000.caffemodel')
conf = 0.5 # Minimum probability to filter weak detections

# Giriş resmini yükle
img_name, img_extension = f"{name}_RGB", "png"
img_rgb = cv2.imread(f'image/{img_name}.{img_extension}')

# Yüz tespiti için blob oluştur
(h, w) = img_rgb.shape[:2]
blob = cv2.dnn.blobFromImage(cv2.resize(img_rgb, (300, 300)), 1.0,
                              (300, 300), (104.0, 177.0, 123.0))

# Blob'u modele gönder ve tespitleri al
print("[BİLGİ] Yüz tespiti gerçekleştiriliyor...")
net.setInput(blob)
detections = net.forward()

# Tespit edilen yüzleri işaretleyelim
for i in range(0, detections.shape[2]):
    confidence = detections[0, 0, i, 2]
    if confidence > conf:
        box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
        (startX, startY, endX, endY) = box.astype("int")
        cv2.rectangle(img_rgb, (startX, startY), (endX, endY), (0,
255, 0), 6)

# Sonuç resmi kaydet
cv2.imwrite(f"result/{img_name}_face_detection.jpg", img_rgb,
[cv2.IMWRITE_JPEG_QUALITY, 100])

# Çıktıyı göster
cv2.imshow("Face Detection on RGB Image", img_rgb)
cv2.waitKey(0)
```

Ek 3: Deep Learning Infrared Resim Yüz ve Sıcaklık Tespiti

```
import numpy as np
import cv2

def fahrenheit2celcius(f):
    return (f - 32) * 5 / 9

# hangi fotoğrafın üzerinde çalışacağız
name = "IR_00065"
# csv dosyasının ismini oluşturalım
csv_file = f'temperature/{name}{"}.csv"'
# Fluke Thermal Imager tarafından bize sağlanan sıcaklık haritası
# 240x320 CCD boyutunda
temperature = np.genfromtxt(csv_file, dtype=np.float64,
delimter=',', skip_header=5, encoding="utf16")
temperature = temperature[:,1:-1]
print(temperature.shape)
print(temperature.dtype)
print(temperature)
##### Derin Öğrenme ile Yüz Tespiti (Face Detection with
Deep Learning) #####
# load our serialized model from disk
print("[BİLGİ] derin öğrenme modeli yükleniyor...")
# dnn deep neural network
net = cv2.dnn.readNetFromCaffe('deploy.prototxt.txt',
'res10_300x300_ssd_iter_140000.caffemodel')
conf = 0.5 # minimum probability to filter weak detections
# load the input image and construct an input blob for the image
# by resizing to a fixed 300x300 pixels and then normalizing it
img_name, img_extension = f"{name}_RGB", ".png"
infrared_img_name, infrared_img_extension = name, ".png"
img_rgb = cv2.imread(f'image/{img_name}.{img_extension}')
img_infrared = cv2.imread(f'image/{infrared_img_name}.{infrared_img_extension}')
img_infrared_clone = img_infrared.copy()
# save output image as 320 x 240 as thermal matrix is at this size.
# This means Fluke thermal imager CCD is 320 x 240. Yet Fluke can
give output up to 5mp by interpolation.
img_thermal_imager_ccd = cv2.resize(img_infrared_clone, (320,240),
cv2.INTER_LINEAR)
img_rgb_ccd = cv2.resize(img_rgb, (320,240), cv2.INTER_LINEAR)
# cv2.imwrite(f"result/{img_name}_CCD.jpg", img_rgb_ccd,
[cv2.IMWRITE_JPEG_QUALITY, 100])
cv2.imwrite(f"result/{infrared_img_name}_thermal_imager_CCD.jpg",
img_thermal_imager_ccd, [cv2.IMWRITE_JPEG_QUALITY, 100])

# face detection with DL on CCD image
(h, w) = img_rgb_ccd.shape[:2]
blob = cv2.dnn.blobFromImage(cv2.resize(img_rgb_ccd, (300, 300)),
1.0,
(300, 300), (104.0, 177.0, 123.0))

# pass the blob through the network and obtain the detections and
predictions
print("[INFO] computing object detections...")
net.setInput(blob)
detections = net.forward()
```

```

max_temperature = []
threshold_temperature = 35
# loop over the detections
for i in range(0, detections.shape[2]):
    # extract the confidence (i.e., probability) associated with the
    # prediction
    confidence = detections[0, 0, i, 2]

    # filter out weak detections by ensuring the `confidence` is
    greater than the minimum confidence
    if confidence > conf:
        # compute the (x, y)-coordinates of the bounding box for the
        object
        box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
        (startX, startY, endX, endY) = box.astype("int")
        # extract the face region in temperature data
        ROI = temperature[startY:endY, startX:endX]
        # find the max temperature in the face region
        max_temperature.append(fahrenheit2celcius(np.max(ROI)))
        text = f"{max_temperature[i]:.2f} C"
        y = startY - 10 if startY - 10 > 10 else startY + 10
        BLACK, WHITE = (0, 0, 0), (255, 255, 255)
        cv2.rectangle(img_thermal_imager_ccd, (startX, startY),
        (endX, endY), WHITE, 2)
        cv2.putText(img_thermal_imager_ccd, text, (startX+1, y-1),
        0, 0.5, WHITE, 1)
        cv2.circle(img_thermal_imager_ccd, (startX+50, startY-20), 2,
        WHITE, 1, 0)
        # save output image
        cv2.imwrite(f"result/{infrared_img_name}_thermal_imager_face_detecti
        on_CCD.jpg", img_thermal_imager_ccd, [cv2.IMWRITE_JPEG_QUALITY, 100])

        # show the output image
        cv2.imshow("Face & Fever Detection on CCD image",
        img_thermal_imager_ccd)

        # face detection with DL on 5mp image, this is for Adem's thesis,
        better pictures
        (h, w) = img_rgb.shape[:2]
        blob = cv2.dnn.blobFromImage(cv2.resize(img_rgb, (300, 300)), 1.0,
        (300, 300), (104.0, 177.0, 123.0))

        # pass the blob through the network and obtain the detections and
        predictions
        print("[INFO] computing object detections...")
        net.setInput(blob)
        detections = net.forward()

        # loop over the detections
        for i in range(0, detections.shape[2]):
            # extract the confidence (i.e., probability) associated with the
            # prediction
            confidence = detections[0, 0, i, 2]

            # filter out weak detections by ensuring the `confidence` is
            # greater than the minimum confidence
            if confidence > conf:

```

```

        # compute the (x, y)-coordinates of the bounding box for the
object
        box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
        (startX, startY, endX, endY) = box.astype("int")
        # draw the bounding box of the face along with the
associated probability
        text = f"{max_temperature[i]:.2f} C"
        y = startY - 10 if startY - 10 > 10 else startY + 10
        BLACK = (0, 0, 0)
        GREEN = (0, 255, 0)
        cv2.rectangle(img_rgb, (startX, startY), (endX, endY),
GREEN, 6)
        cv2.rectangle(img_infrared, (startX, startY), (endX, endY),
WHITE, 6)
        cv2.putText(img_infrared, text, (startX+5, y-15), 0, 1.5,
WHITE, 3)
        cv2.circle(img_infrared, (startX+150, startY-55), 5, WHITE,
4, 0)
        if max_temperature[i] > threshold_temperature:
            cv2.putText(img_infrared, "Fever detection!", (startX-
75, y-65), 0, 1.5, WHITE, 3)
        # save output image
        cv2.imwrite(f"result/{infrared_img_name}_fever_detection.jpg",
img_infrared, [cv2.IMWRITE_JPEG_QUALITY, 100])
        cv2.imwrite(f"result/{img_name}_face_detection.jpg", img_rgb,
[cv2.IMWRITE_JPEG_QUALITY, 100])
        # show the output image
        s = 0.5
        rimage = cv2.resize(img_infrared, (int(s*img_infrared.shape[1]),
int(s*img_infrared.
shape[0])), cv2.INTER_LINEAR)
        cv2.imshow("Face & fever detection on 5mp image", rimage)
        cv2.waitKey(0)

```

Ek 4: Haar Cascade Infrared Resim Yüz ve Sıcaklık Tespiti

```
8 import cv2
9 print("[BİLGİ] Haar Cascade yüz tespit edici'yi yüklüyor...")
10 detector = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
11 imgName = "IR_00065.png"
12 imgName, imgExtension = imgName.split('.')
13 img = cv2.imread(f"image/{imgName}.{imgExtension}")
14 # Haar Cascade gri tonlu resimler üzerinde çalıştığından renk uzayı
    dönüşümü yapalım
15 gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) # RGB uzayd an gri tonlu
    uzaya dönüşüm
16 print("[BİLGİ] Yüz tespiti gerçekleştiriliyor...")
17
18 scaleFactor, minNeighbors = 1.09, 1 # iki adet ayarlanabilir parametre
19
20 rects = detector.detectMultiScale(gray, scaleFactor, minNeighbors)
21 print(f"[BİLGİ] {len(rects)} adet yüz tespit edildi.")
22 # tespit edilen yüzleri resim üzerinde dikdörtgen olarak göster
23 for (x,y,w,h) in rects:
24     cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0), 9)
25 s = 0.22 # s=(0-1] resmi ekranda görüntülemek için ölçeklendir
26 rimg = cv2.resize(img, (int(s*img.shape[1]), int(s*img.shape[0])), 0)
27 cv2.imshow('Face detection with Haar Cascade', rimg)
28 cv2.waitKey(0)
29 cv2.imwrite(f"result/{imgName}_scaleFactor_{scaleFactor}_minNeighbors_{
    minNeighbors}.jpg",
30             img, [cv2.IMWRITE_JPEG_QUALITY, 50])
```

ETİK KURUL KARARI



T.C.
GÜMÜŞHANE ÜNİVERSİTESİ REKTÖRLÜĞÜ
BİLİMSEL ARAŞTIRMA VE YAYIN ETİĞİ KURULU



Sayı : E-95674917-108.99-259777
Konu : Etik Kurul Onay

Sayın Dr. Öğr. Üyesi Emrah KAPLAN

"TERMAL GÖRÜNTÜ İLE COVID-19 VE BENZERİ ATEŞLİ HASTALIK TESPİTİNDE DERİN ÖĞRENME VE LLM (LARGE LANGUAGE MODELS) KARŞILAŞTIRMASI" konulu etik kurul başvurunuz; Gümüşhane Üniversitesi Bilimsel Araştırma ve Yayın Etiği Kurulu'nun 27/06/2024 tarih ve 2024/6 sayılı toplantısında görüşülmüş olup; projemin yürürlükteki mevzuata uygun olduğuna oy birliği ile karar verilmiştir.
Bilgilerinize rica ederim.

Prof. Dr. Günay ÇAKIR
Kurul Başkanı

ÖZGEÇMİŞ

2013 yılında Lefke Avrupa Üniversitesi Bilgisayar Mühendisliği bölümünden mezun olduktan sonra, havacılık, finans ve mobil sektörlerinde farklı projelerde görev aldım. Teknolojiye olan tutkum ve analitik düşünme becerim sayesinde yazılım geliştirme ve süreç iyileştirme alanlarında değer yaratan çözümler ürettim. Şu anda Kıdemli Yazılım Uzmanı olarak bankacılık ve finans sektörüne yönelik projelerde çalışmaya devam ediyorum. Kariyerim boyunca çeşitli sektörlerde edindiğim tecrübelerle, yenilikçi ve etkili projeler üretmeyi hedefliyorum.

