



T.C.

AKSARAY ÜNİVERSİTESİ

SOSYAL BİLİMLER ENSTİTÜSÜ

YÖNETİM BİLİŞİM SİSTEMLERİ ANABİLİM DALI

**PEDİLEN AYAK GÖRÜNTÜLERİ ÜZERİNDE DERİN ÖĞRENME
YÖNTEMLERİ KULLANILARAK HATALI ÜRÜN TESPİTİ**

YÜKSEK LİSANS TEZİ

MESUT HAKSES

DANIŞMAN

DOÇ. DR. KEMAL ADEM

AKSARAY 2023

T.C.
AKSARAY ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
YÖNETİM BİLİŞİM SİSTEMLERİ ANABİLİM DALI

**PEDİLEN AYAK GÖRÜNTÜLERİ ÜZERİNDE DERİN ÖĞRENME
YÖNTEMLERİ KULLANILARAK HATALI ÜRÜN TESPİTİ**

YÜKSEK LİSANS TEZİ

MESUT HAKSES

DANIŞMAN
DOÇ. DR. KEMAL ADEM

AKSARAY 2023

TELİF HAKKI VE TEZ FOTOKOPİ İZİN FORMU

Bu tezin tüm hakları saklıdır. Kaynak göstermek koşuluyla tezin teslim tarihinden itibaren(...) ay sonra tezden fotokopi çekilebilir.

YAZARIN

Adı : Mesut
Soyadı : HAKSES
Bölümü : Yönetim Bilişim Sistemleri
İmza :
Teslim tarihi :

TEZİN

Türkçe Adı : Pedilen Ayak Görüntüleri Üzerinde Derin Öğrenme Yöntemleri Kullanılarak Hatalı Ürün Tespiti

İngilizce Adı : Faulty Product Detection Using Deep Learning Methods on Padded Foot Images

ETİK İLKELERE UYGUNLUK BEYANI

Tez yazma sürecinde bilimsel ve etik ilkelere uyduđumu, yararlandığım tüm kaynakları kaynak gösterme ilkelerine uygun olarak kaynakçada belirttiđimi ve bu bölümler dışındaki tüm ifadelerin şahsıma ait olduđunu beyan ederim.

Yazar Adı Soyadı: Mesut HAKSES

İmza:

T.C.
AKSARAY ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
JÜRİ ONAY SAYFASI

Mesut HAKSES tarafından hazırlanan “Pedilen Ayak Görüntüleri Üzerinde Derin Öğrenme Yöntemleri Kullanılarak Hatalı Ürün Tespiti” başlıklı tez çalışması aşağıdaki jüri tarafından oy birliği ile Aksaray Üniversitesi Yönetim Bilişim Sistemleri Anabilim Dalı’nda Yüksek Lisans Tezi olarak kabul edilmiştir.

Danışman : Doç. Dr. Kemal ADEM

Bilgisayar Mühendisliği, Sivas Bilim ve Teknoloji Üniversitesi

Üye: Doç. Dr. Rahmi BAKİ

Yönetim Bilişim Sistemleri, Aksaray Üniversitesi

Üye: Dr. Öğr. Üyesi Bayram Melih YILMAZ

Bilgisayar Mühendisliği, Sivas Bilim ve Teknoloji Üniversitesi

İmza

.....

.....

.....

Tez Savunma Tarihi: 23 / 06 / 2023

Sosyal Bilimler Enstitüsü Yönetim Kurulu’nun tarih ve sayılı kararı ile onaylanmıştır.

Sosyal Bilimler Enstitüsü Müdürü

Doç. Dr. Sevilay USLU DİVANOĞLU

İmza

TEŞEKKÜR

Tez danışmanlığı süresince çalışmanın sonuçlanmasında ve karşılaşılan zorlukları aşmada her zaman yardımcı ve yol gösterici olan Sayın Doç. Dr. Kemal ADEM'e, çalışmaya değer katan jüri üyeleri Doç. Dr. Rahmi BAKİ ve Dr. Öğr. Üyesi Bayram Melih YILMAZ'a, veri seti oluşturma aşamasında destek olan firma AR-GE personeli Ramazan YILDIZ ve firma Üretim Müdürü Burcu ADIGÜZEL'e, yüksek lisans yapmam konusunda sürekli teşvik ve desteklerini esirgemeyen kardeşim Hasan HAKSES ve arkadaşım Umut Furkan KÜÇÜK'e, her zaman yanımda olan babam Ahmet HAKSES, annem Rukiye HAKSES, eşim Kübra HAKSES ve oğlum Alper HAKSES'e, sonsuz teşekkürlerimi sunarım.

AKSARAY ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
PEDİLEN AYAK GÖRÜNTÜLERİ ÜZERİNDE DERİN ÖĞRENME
YÖNTEMLERİ KULLANILARAK HATALI ÜRÜN TESPİTİ

(Yüksek Lisans Tezi)

Mesut HAKSES, Aksaray 2023

ÖZET

Endüstriyel üretim süreçlerinde hatalı ürünlerin tespiti, üretim kalitesinin sağlanması ve müşteri memnuniyetinin artırılması açısından kritik bir öneme sahiptir. Hatalı ürün tespiti için öncelikle, ürünün kalite kriterlerine uygunluğunu belirlemek için spesifikasyonlar oluşturulmalıdır. Bu spesifikasyonlar, ürünün fiziksel özelliklerini ve kabul edilebilir kalite standartlarını içermelidir. Yaşamsal faaliyetlerin sürdürülmesi, öz bakım ve ulaşım gibi aktivitelerin gerçekleştirilmesinde büyük rol oynayan ayaklar, doğuştan veya sonradan kaybedilebilmektedir. Gelişen teknolojiyle birlikte bu kaybedilen uzuvların yerini protez denenen yapay organlar almaktadır. En yaygın protez çeşitlerinden pedilen ayak üretimini yapan firmamızda üretim sonrası insan gözüyle kalite kontrol yapıldığı için meydana gelen bazı üretim kusurları gözden kaçabilmektedir. Firmamızdaki pedilen ayak üretiminde; kullanılan kimyasalların tam karışmaması veya kalıba enjekte edilmesi sırasında hava yapması sonucu oluşan hava kabarcıkları patlarsa çukur hatası, patlamaz ve kabarcık olarak kalırsa kabarcık hatası olarak ifade edilen hatalı ürünler ortaya çıkmaktadır. Bu hatalı ürünler gelişen teknoloji sayesinde derin öğrenme yöntemlerinin kullanımıyla da tespit edilebilir hale gelmektedir. Bu tez çalışmasında, pedilen ayak üretimi yapan firmadan temin edilen görüntülere Python kodları aracılığıyla görüntü çoğaltma işlemi uygulanması sonucu 1200 görüntüden oluşan veri seti oluşturulmuştur. Bu görüntü veri seti üzerinde derin öğrenme yöntemlerinden evrimsel sinir ağı mimarileri kullanılarak hatalı ürün tespitinin gerçekleştirilmesi amaçlanmaktadır. Toplam 1200 görüntülük veri seti üzerinde Xception, InceptionResNetV2, DenseNet201, NasNetMobile ve EfficientNetV2S mimarileri uygulanarak sonuçlar karşılaştırılmıştır. Deneysel değerlendirmeler neticesinde, InceptionResNetV2 modelinde en üstün başarı oranlarına ulaşılmıştır. 20 epoch değeri kullanıldığında, %99.72 doğruluk, %99 geri çağırma, %99 kesinlik ve %100 f1 skoru elde edilmiştir.

Bilim Kodu: 92431, 92432, 92418

Anahtar Kelimeler: Nesne Tespiti, Derin Öğrenme, CNN, Pedilen Ayak, Hatalı Ürün

Sayfa Sayısı: 84

Danışman: Doç. Dr. Kemal ADEM

AKSARAY UNIVERSITY
GRADUATE SCHOOL OF SOCIAL SCIENCES
FAULTY PRODUCT DETECTION USING IMAGE DEEP LEARNING
METHODS ON PEDİLEN FOOT IMAGES

(M.A. Thesis)

Mesut HAKSES, Aksaray 2023

ABSTRACT

Detection of faulty products in industrial production processes has a critical importance in terms of ensuring production quality and increasing customer satisfaction. For faulty product detection, first of all, specifications should be created to determine the conformity of the product to the quality criteria. These specifications should include the physical properties of the product and acceptable quality standards. The feet, which play a major role in the maintenance of vital activities, self-care and activities such as transportation, can be lost congenitally or later. With the developing technology, these lost limbs are replaced by artificial organs called prostheses. In our company, which produces padded feet, which is one of the most common types of prostheses, some production defects can be overlooked because quality control is carried out with the human eye after production. In the production of padded feet in our company; If the air bubbles, which are formed as a result of incomplete mixing of the chemicals used or air during injection into the mold, burst, faulty products arise, which are expressed as pit error, and if they do not burst and remain as bubbles, faulty products occur. Thanks to the developing technology, these faulty products can also be detected by the use of deep learning methods. In this thesis, a data set consisting of 1200 images was created as a result of applying image duplication through Python codes to the images obtained from the company that produces the padded feet. It is aimed to perform faulty product detection on this image dataset by using convolutional neural network architectures from deep learning methods. The results were compared by applying Xception, InceptionResNetV2, DenseNet201, NasNetMobile and EfficientNetV2S architectures on a total of 1200 images dataset. As a result of the experimental evaluations, the highest success rates were achieved in the InceptionResNetV2 model. Using 20 epoch values, 99.72% accuracy, 99% recall, 99% precision and 100% f1 score were obtained.

Science Code: 92431, 92432, 92418

Key Words: Object Detection, Deep Learning, CNN, Padded Foot, Faulty Product

Page Number: 84

Supervisor: Assoc. Prof. Dr. Kemal ADEM

İÇİNDEKİLER

TELİF HAKKI VE TEZ FOTOKOPİ İZİN FORMU	İ
ETİK İLKELERE UYGUNLUK BEYANI.....	İİ
JÜRİ ONAY SAYFASI.....	İİİ
TEŞEKKÜR.....	İV
ÖZET	V
ABSTRACT.....	VI
TABLO LİSTESİ.....	İX
ŞEKİL LİSTESİ.....	X
KISALTMA VE SİMGE LİSTESİ.....	Xİ
1. GİRİŞ	1
1.1. TEZİN AMACI	2
1.2. TEZİN KAPSAMI.....	2
1.3. TEZİN ÖNEMİ.....	2
2. LİTERATÜR ARAŞTIRMASI	4
3. TEMEL KAVRAMLAR	10
3.1. PEDİLEN AYAK.....	10
3.1.1. Pedilen Ayak Özellikleri ve Çeşitleri	10
3.1.2. Pedilen Ayak Üretim Süreci	13
3.2. NESNE TESPİTİ.....	14
4. YÖNTEM VE MİMARİ.....	17
4.1. VERİ KÜMESİNİN HAZIRLANMASI	17
4.2. DERİN ÖĞRENME.....	18
4.3. EVRİŞİMSSEL SİNİR AĞI (CNN-CONVOLUTIONAL NEURAL NETWORK).....	20
4.3.1. Evrişim Katmanı	23
4.3.2. Düzeltilmiş Doğrusal Birim Katmanı (ReLU).....	24
4.3.3. Havuzlama Katmanları	24
4.3.4. Tam Bağlantılı Katmanlar.....	26
4.4. ANALİZ YAPILAN CNN MODELLERİ	27
4.4.1. Xception Modeli	27
4.4.2. InceptionResnetV2 Modeli	28
4.4.3. DenseNet201 Modeli	29
4.4.4. NasNetMobile Modeli	30
4.4.5. EfficientNetV2S Modeli	30
4.5. BAŞARIM ÖLÇÜTLERİ	33
4.5.1. Doğruluk Ölçütü	33
4.5.2. F1 Skor Değeri.....	34
4.5.3. Geri Çağırma Değeri.....	35
4.5.4. Kesinlik Değeri	35
4.5.5. Karışıklık Matrisi Ölçütü	36

4.6. ÇALIŞMADA KULLANILAN HİPER PARAMETRELER.....	37
4.6.1. Epoch Değeri	37
4.6.2. Analizlerde Kullanılan Model Eğitimi (Training)	38
4.6.3. Batches	39
4.6.4. Öğrenme Oranı (Learning Rate)	39
4.6.5. Optimizasyon Algoritması	40
4.7. DERİN ÖĞRENME KÜTÜPHANELERİ	40
4.7.1. Keras Kütüphanesi	40
4.7.2. Tensorflow Kütüphanesi	41
5. BULGULAR.....	42
5.1. XCEPTION MODELİ ANALİZ SONUÇLARI	43
5.2. INCEPTIONRESNETV2 MODELİ ANALİZ SONUÇLARI.....	46
5.3. DENSENET201 MODELİ ANALİZ SONUÇLARI	48
5.4. NASNETMOBILE MODELİ ANALİZ SONUÇLARI	51
5.5. EFFICIENTNETV2S MODELİ ANALİZ SONUÇLARI	53
5.6. ANALİZDE KULLANILAN MODELLERİN PERFORMANS KARŞILAŞTIRMASI	56
6. SONUÇ VE ÖNERİLER.....	59
7. KAYNAKÇA.....	61
EKLER.....	72
AKSARAY ÜNİVERSİTESİ TEZ DEĞERLENDİRME FORMU	73

TABLO LİSTESİ

Tablo 1: ImageNet1K eğitim sonuçları (Tan, Le, 2021)	32
Tablo 2: ImageNet21K önceden eğitilen ve ince ayarı yapılan modeller (Tan, Le, 2021)..	32
Tablo 3: Modelleri Eğitiminde Kullanılan Parametreler	43
Tablo 4: Xception Modeli Ölçüm Parametre Sonuçları	43
Tablo 5: Xception Modeli Sınıflama Raporları	45
Tablo 6: InceptionResNetV2 Modeli Ölçüm Parametre Sonuçları	46
Tablo 7: InceptionResNetV2 Modeli Sınıflama Raporları	48
Tablo 8: DenseNet201 Modeli Ölçüm Parametre Sonuçları	48
Tablo 9: DenseNet201 Modeli Sınıflama Raporları	50
Tablo 10: NasNetMobile Modeli Ölçüm Parametre Sonuçları.....	51
Tablo 11: NasNetMobile Modeli Sınıflama Raporları	53
Tablo 12: EfficientNetV2S Modeli Ölçüm Parametre Sonuçları	53
Tablo 13: EfficientNetV2S Modeli Sınıflama Raporları	55
Tablo 14: Modellerin Başarım Ölçütleri ve Eğitim Süresi Değerlerinin Karşılaştırılması .	56

ŞEKİL LİSTESİ

Şekil 1: Hatalı Pedilen Ayak Görüntülerinin Hata Türüne Göre Nesne Tespiti Örnekleri .	14
Şekil 2: Veri Setimizdeki Örnek Hatalı ve Hatasız Pedilen Ayak Görüntüleri	17
Şekil 3: Görüntü Çoğaltma İşlemleri Sonrası Elde Edilen Pedilen Ayak Görüntüleri	18
Şekil 4: Yapay Zeka, Makine Öğrenmesi ve Derin Öğrenme (Bakshi, 2019)	19
Şekil 5: Derin Öğrenme Türleri (Deng & Yu, 2014).....	20
Şekil 6: Evrişimsel Sinir Ağı Mimarisinin Yapısı (Rubik's Code, 2018)	22
Şekil 7: Görüntü Evrişim Aşamaları ve Çıktı Görüntüsü Elde Etme	24
Şekil 8: Ortalama Ortaklama ve Maksimum Ortaklama Katman Yapısı (Doğan, 2020)	25
Şekil 9: Değerlerin Tam Bağlı Katman Aşamasında Kullanmayı Sağlayan Düzleştirme İşlemi (Konca, 2020)	27
Şekil 10: Xception Mimarisi (Konate, 2022).....	27
Şekil 11: InceptionResNet-v2 mimarisi (Szegedy vd., 2016).....	29
Şekil 12: DenseNet201 Modelinin Yapısı (Hayıt & Çınarar, 2022).....	29
Şekil 13: NasNetMobile Modelinin Yapısı (Herdian vd., 2019)	30
Şekil 14: EfficientNetV2 Modellerinin Eğitim Verimlilikleri (Tan & Le, 2021).....	32
Şekil 15: İkili Sınıflandırmaya Ait Karışıklık Matrisi	37
Şekil 16: Jupyter Notebook Görselinin Google Colab Ortamındaki Görünümü	42
Şekil 17: Veri Kümesi Dağılımının Görseli.....	43
Şekil 18: Xception Modeli Eğitim Grafikleri	44
Şekil 19: Xception Modeli Karışıklık Matrisleri	45
Şekil 20: InceptionResNetV2 Modeli Eğitim Grafikleri	47
Şekil 21: InceptionResNetV2 Modeli Karışıklık Matrisleri	47
Şekil 22: DenseNet201 Modeli Eğitim Grafikleri	49
Şekil 23: DenseNet201Modeli Karışıklık Matrisleri	50
Şekil 24: NasNetMobile Modeli Eğitim Grafikleri	52
Şekil 25: NasNetMobile Modeli Karışıklık Matrisleri	52
Şekil 26: EfficientNetV2S Modeli Eğitim Grafikleri	54
Şekil 27: EfficientNetV2S Modeli Karışıklık Matrisleri	55

KISALTMA VE SİMGE LİSTESİ

API	Application Programming Interface
CNN	Convolutional Neural Network
DER	Dynamic Elastic Response
f	Aktivasyon fonksiyonu
FLOPS	Saniyede kayan nokta işlemleri
FN	Yanlış Negatif
FP	Yanlış Pozitif
GPU	Graphic Processing Unit
NAS	Neural Architecture Search
PU	Poliüretan
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Networks
R_CNN	Region Based CNN
SACH	Solid Anide Cushion Heel
SGK	Sosyal Güvenlik Kurumu
tf	tensorflow
TN	Gerçek Negatif
TP	Gerçek Pozitif

1. GİRİŞ

Endüstriyel üretim süreçlerinde hatalı ürünlerin tespiti önemli bir sorun oluşturmaktadır. Bu sorunun çözümü için geleneksel yöntemler yerine son yıllarda görüntü işleme ve derin öğrenme tekniklerinin kullanımı ön plana çıkmaktadır. Derin öğrenme yöntemleri, görsel verilerin analizi ve sınıflandırılması konusunda yüksek başarı oranları sağlamaktadır. Bu bağlamda pedilen ayak üretimi gibi örnek bir endüstriyel üretim sürecinde hatalı ürün tespiti için derin öğrenme yöntemlerinin kullanılması oldukça önemli bir araştırma alanıdır.

Bu çalışmanın amacı, pedilen ayak görüntüleri üzerinde derin öğrenme yöntemlerini kullanarak hatalı ürün tespiti yapmaktır. Bu amaç doğrultusunda, öncelikle pedilen ayak üretimi yapan bir firmadan temin edilen veri seti üzerinde çalışılmıştır. Bu veri seti, hatalı ve hatasız ürünleri içermektedir. Derin öğrenme yöntemleri kullanılarak bu veri seti üzerinde eğitim gerçekleştirilmiş ve hatalı-hatasız şeklinde sınıflandırma yapabilen bir model oluşturulmuştur.

Bu araştırmanın önemi, birçok açıdan ortaya çıkmaktadır. İlk olarak, derin öğrenme tekniklerinin kullanılması, insan faktörünü devreden çıkararak daha sağlıklı bir hata tespiti yapılmasını sağlamaktadır. Bu durum, üretim süreçlerindeki insan hatalarının azalmasına ve kalite kontrolünün iyileştirilmesine yardımcı olmaktadır. İkinci olarak, bu araştırma pedilen ayak üretimi gibi örnek bir endüstriyel süreçte derin öğrenme yöntemlerinin etkinliğini göstererek, benzer üretim süreçlerinde de bu yöntemlerin kullanımını teşvik edebilir.

Bu çalışmanın temel katkılarından biri, derin öğrenme yöntemlerinin hatalı ürün tespitinde ne kadar etkili olduğunu ortaya koymasıdır. Bu yöntemler, büyük miktarda veriyi hızlı bir şekilde analiz edebilme yetenekleriyle öne çıkmaktadır ve doğru sınıflandırma yapabilme kabiliyetiyle üretim süreçlerinin verimliliğini artırmaktadır. Bu nedenle, pedilen ayak üretimi gibi endüstriyel süreçlerde derin öğrenme yöntemlerinin kullanılması, üretim kusurlarının tespitinde daha doğru ve güvenilir sonuçlar elde etmeyi sağlamaktadır.

Bu tez, pedilen ayak görüntüleri üzerinde derin öğrenme yöntemlerini kullanarak hatalı ürün tespiti üzerine bir araştırma sunmaktadır. Araştırmanın amacı, endüstriyel üretim süreçlerinde kalite kontrolünün iyileştirilmesine ve hatalı ürünlerin erken tespitine yönelik pratik bir çözüm sunmaktır. Bu çalışmanın sonuçları, endüstriyel üretim süreçlerinde hatalı ürün tespiti için derin öğrenme yöntemlerinin etkinliğini göstermektedir.

1.1. Tezin Amacı

En yaygın protez çeşitlerinden pedilen ayak üretimini yapan firmamızda üretim sonrası insan gözüyle kalite kontrol yapıldığı için meydana gelen bazı üretim kusurları gözden kaçabilmektedir. Bu kusurlar pedilen ayak üretiminde kullanılan kimyasalların tam karışmaması veya kalıba enjekte edilmesi sırasında hava yapması sonucu oluşan hava kabarcıkları patlarsa çukur hatası, patlamaz ve kabarcık olarak kalırsa kabarcık hatası olarak ifade edilen kusurlardır.

Bu çalışmanın motivasyonunu oluşturan amaç ise;

- Derin öğrenme yöntemleri ile protez ayak üretim sürecinde birincil kaynak formatında elde edilen orijinal görüntülerle hatalı ürünleri ortaya çıkarmak,
- Xception, InceptionResNetV2, DenseNet201, NasNet Mobile ve EfficientNetV2S gibi CNN modellerini hatalı üretilen pedilen ayakların tespitinde kullanmak ve doğru hatalı ürün tespitindeki sonuçların kıyaslanması ile en iyi hatalı ürün tespit modeli belirlenerek protez ayak üretim sürecinin geliştirilmesine katkı sağlamaktır.

1.2. Tezin Kapsamı

Üretim kusurlarının tespitinde insan faktörünü devreden çıkarıp derin öğrenme yöntemleriyle daha sağlıklı hata tespiti yapma amacımızı gerçekleştirmek için pedilen ayak üretimi yapan firmadan temin edilen veri kümesine görüntü çoğaltma işlemi uygulanması sonucu 1200 görüntüden oluşan veri seti elde edilmiştir. Bu veri setini kullanarak, görüntü işleme, görüntü sınıflandırma ve nesne tespiti gibi görsel verilerle çalışmak için etkili bir yöntem olarak kabul edilen CNN sinir ağı türünün çeşitli derin öğrenme modelleriyle eğitim gerçekleştirilmiştir. Xception, InceptionResNetV2, DenseNet201, NasNetMobile ve EfficientNetV2S derin öğrenme modelleri kullanarak karşılaştırmalı olarak en doğru hatalı ürün tespiti yapan model ortaya çıkarılacaktır.

1.3. Tezin Önemi

İnsan vücudu sürekli olarak atmosfer basıncına ve yer çekimine maruz kalmaktadır. Bu sebeple ayakta durmak ve yürümek vücut sağlığı için olmazsa olmazdır. Yaşamsal faaliyetlerin sürdürülmesi, öz bakım, ulaşım gibi aktivitelerin gerçekleştirilmesinde ayaklar büyük rol oynamaktadır. Bu organlar doğuştan veya sonradan kaybedilebilmektedir. Gelişen

teknolojiyle birlikte bu kaybedilen uzuvların yerini protez denen yapay organlar almaktadır. Günümüz şartları protezi bir bilim dalı haline getirmiştir. Günümüzde birçok protez üretilmektedir. En yaygın protez çeşitlerinden biri de pedilen ayaktır.

Pedilen ayak üretiminde kullanılan kimyasalların tam karışmaması veya kalıba enjekte edilmesi sırasında hava yapması sonucu oluşan çukur hatası ve kabarcık hatası olarak ifade edilen hatalı ürünler üretilmektedir. Bu ürünlerdeki hatalar operatör tarafından göz yordamıyla tespit edilmekte olup, gelişen teknoloji ile derin öğrenme yöntemleri kullanılarak da yapılabilmektedir. Derin öğrenme, birden çok hesaplama katmanından oluşan bir model tarafından gerçekleştirilen işlemlerle verilerin soyutlama düzeylerini öğrenme yeteneğine sahip bir öğrenme yaklaşımıdır. Bu yöntemler, konuşma tanıma, görsel nesne tanıma, nesne algılama gibi birçok alanda teknolojide önemli gelişmelere olanak sağlamıştır. Derin öğrenme, verilerin temsillerini çıkarmak ve karmaşık ilişkileri anlamak için çok katmanlı bir yapı kullanmakta ve bu sayede daha yüksek bir öğrenme kapasitesine sahip olmaktadır.

Gelişen teknoloji sonucu birçok alanda olduğu gibi üretim yapan firmalarda da elde edilen ürünlerin hatalı ürün tespiti ile ilgili kalite kontrol çalışmalarında derin öğrenme yöntemlerine başvurulmaktadır.

2. LİTERATÜR ARAŞTIRMASI

Protezler insanoğlunun hayatına ilk olarak tahta bacak veya kanca olarak girmiştir. Özellikle Dünya Savaşlarında ortaya çıktığını söylemek mümkündür. Savaş sonrası dönemde protez ihtiyacı Amerikan Protez ve Ortez Derneği'nin kurulması ile modern protez yöntemlerinin yaygınlaştığı bilinmektedir (Yılmaz, 2009).

Protez ayakların temel amacı, yürüme aktivitesini gerçekleştirebilmek için eksik veya işlevini kaybetmiş olan uzuvların yerine geçecek bir yapı sağlamaktır. Teknolojik ilerlemeler, protez ayakların koşma aktivitelerine uygun olarak da tasarlanabilmesine olanak sağlamıştır. Bu bağlamda, enerji depolama özelliği olan ayaklar geliştirilmiştir ve bu amaçla grafit kompozit, Delrin (DuPont), Kevlar (DuPont), poliüretan elastomerler ve esnek kauçuk gibi malzemeler kullanılmaktadır. Bu malzemeler, enerjiyi depolayabilme yetenekleri ve dayanıklılıkları sayesinde protez ayakların performansını artırmak için tercih edilmektedir. (Tang vd., 2008). Ayrıca kompozit malzemeler ayak, boru ve soket yapımında kullanılmaktadır (Ramakrishna vd., 2001).

Akdoğan (2002), üç farklı protez ayağın yürüyüş, denge, esneklik, ampute tarafta ağırlık taşıma ve ambulasyon aktiviteleri açısından kıyasladığı çalışmasında 20 farklı unilateral diz altı ampute SACH kullanmıştır. Bir haftalık süreç sonrasında dinamik ayakla yapılan aktivitelerin konvansiyonel ayağa kıyasla daha başarılı yapıldığı sonucuna ulaşmıştır. Protez ayak tercihinde zorlanan hastaların dinamik ayağı tercih edebileceğini ifade etmiştir.

Görüntü işleme yöntemleri ile protez ayak üretim sürecinin geliştirilmesine yönelik alan henüz literatürde çalışılmış değildir. Ancak literatürdeki önemli çalışmalar arasında görüntüleme tabanlı kalite kontrolü üzerine günlük iş hayatını kolaylaştırıcı bir yenilikçi yaklaşım sunan Adem, K. (2013) çalışmasında, karton paketlerin şerit dikişlerindeki hata oranlarını tespit ederek kalite kontrol süreçlerine önemli bir katkı sağlamıştır.

Derin öğrenme yöntemleri ile farklı alanlarda çok sayıda görüntü sınıflandırma ve nesne tespiti çalışması yapılmıştır (Adem, 2022; Yurdusev, vd. 2023; Kılıçarslan, vd. 2021; Hekim, vd. 2020; Yılmaz, vd. 2023). Adem (2022), aktivasyon fonksiyonlarının ve katman sayısını dairesel Hough dönüşümü ve evrişimli sinir ağları kullanılarak eksüdaların saptanması üzerine yaptığı çalışmasında evrişimli sinir ağları modelinde ReLU, ELU, Leaky ReLU, Softplus ve Swish aktivasyon fonksiyonları kullanılarak eksüda tespiti yapılmıştır. Elde edilen sonuçların karşılaştırılmasıyla DiaretDB0 ve DiaretDB1 veri tabanlarındaki deneysel sonuçlara göre Swish aktivasyon fonksiyonunun diğerlerinden daha iyi bir performansa sahip

olduğu sonucuna ulaşmıştır.

Ayrıca Çağıl ve Yıldırım (2020), endüstriyel kapı ve pencere üretimi yapan bir firma, müşterilere montaj için gerekli aksesuar parçalarını temin etme sürecinde ortaya çıkan hataları önlemek için çözüm arayışına girmiştir. Geleneksel el ile parça ekleme yönteminin hatalara yol açtığı gözlemlenmiştir. Bu sorunu çözmek amacıyla firma, Google Colab Platformu, Python Programlama Dili ve OpenCV kütüphanesini kullanarak görüntü işleme ve derin öğrenme tabanlı nesne tespiti yöntemlerini uygulayarak bir sistem geliştirmiştir. Bu sistemde, her bir aksesuar parçasının görüntüsü tanıtılmış ve müşteri taleplerinin hatasız ve daha hızlı bir şekilde karşılanabilmesi sağlanmıştır. Bu sayede, müşteriye yanlış veya eksik parça temin etme riski azaltılmış ve süreç verimliliği artırılmıştır.

Tekstil endüstrisi özelinde kumaşlardaki hata tespitini inceledikleri çalışmalarında Şeker vd. (2016), insan gözü ile hata tespiti çalışmalarında %60 seviyelerinde hata tespiti yapılabildiğini bunun da ciddi manada zaman kaybına neden olduğu sorununu bulgulamışlardır. Bu sorunun çözümü için kumaş hata tespitinde çok katmanlı mimarilerle ayırt edilen ve yüksek başarı gösteren derin öğrenme uygulayarak hata tespitlerini bilgisayar ortamında yapılmasına imkân sağlamış ve otomatik kodlayıcının giriş değeri ile hiper parametrelerini ayarlayarak hata tespiti ortaya çıkarma başarısını artırmışlardır.

Derin öğrenme ile tekstil tipinin tanınmasını inceleyen Gürgen (2019), daha önce var olan tekstil tipi tanıma probleminin maliyet unsurunun yüksek olduğu sensör tabanlı sistemler yerine bireylerin rahatlıkla ulaşabileceği standart kamera ile elde edilen görüntüler, resimler, fotoğraflar aracılığıyla sınıflandırılması konusu üzerinde durmuştur. Geliştirilen yöntem sayesinde, optimum seviyede maliyetle kullanım kolaylığı sağlanarak günlük kullanımı yaygın olan tekstil ürünlerinin sınıflandırılması yapılmıştır.

Görüntü algılama teknolojileri, temelde insan gözü ile görüntü algılaması ve tanımlamasına dayalı olup, insandaki görme sisteminde yapılan süreçlerin bilgisayar ortamına aktararak makineye öğreniminin çalışılmasıdır. Görüntü işleme, mevcut görüntülerin üzerinde iyileştirme yapılması ve bilgisayar aracılığıyla görüntülerin işlenmesi olarak tanımlanan bir yöntemdir. Bu yöntemle, fotoğraflanan görüntüler sayısallaştırılır ve matematiksel olarak karakterize edilir. Görüntü işleme, farklı mühendislik disiplinlerinde birçok problemi kolaylıkla çözme imkanı sağlamış ve önemli bilimsel veriler elde edilmesine katkı sağlamıştır. Bu yöntem, görüntüler üzerinde yapılan analizlerle nesnelerin tanınması, ölçülmesi, sınıflandırılması, hareketin takibi gibi birçok uygulama alanında kullanılmaktadır.

Bu sayede, görüntü işleme, bilimsel araştırmalarda ve mühendislik projelerinde önemli bir araç haline gelmiştir (Edizer, 2006, Çayırılı, 2006, Taşçı, 2011, Uras, 2006, Yılmaz, 2007).

Çoşkun (2022), Endüstriyel karo üretiminde hatalı ürünlerin geleneksel yöntemlerden olan insan gözü ile tespitini kolaylaştırmak ve süreçte yaşanan olumsuzlukların en aza indirgenmesine yardımcı olmak maksadıyla derin öğrenme yöntemleri ile daha sağlıklı bir şekilde hatalı ürün tespitine odaklandığı çalışmada ‘Kalite Kontrol Bütünleşik Otomasyon Sistemi’ adını verdiği sistemi geliştirmiştir. Bu sayede hata tespit ve kalite kontrol süreçlerinin otomasyona aktarılarak dijitalleşmesine imkan tanımıştır. 5 farklı kusur tespitini 150 veri seti üzerinde incelediği çalışmada bağıl hatalarını $8.78508E-4$ olarak bulmuştur.

Endüstriyel anlamda üretim süreci yaşanan gıda sektörüne örnek verilebilecek Antep fıstığının görüntü işleme teknikleri ile sınıflandırılmasına yönelik derin öğrenme yöntemini kullanan Aktaş (2022), dış kabuk rengine göre bir sınıflandırma yapmıştır. Antep fıstığının hasat zamanından raflarda tüketim sürecine kadar hassas kurutma teknikleri kullanılarak ürünün kalitesinin korunması marka değerinin sürdürülebilirliği için hayati öneme sahiptir. Bu nedenle dış kabuk rengine göre sınıflandırma yapan Aktaş (2022), çalışmada AlexNet ağ yapısını kullanarak sıfırdan veri toplama yöntemiyle endüstriyel bir test düzeneği inşa etmiştir. Böylece dış kabuk rengine göre olgunlaşmış Antep fıstıklarının tespiti yapılmıştır.

Görüntü algılama teknolojileri aracılığıyla Covid-19 pandemi süresince bulaşı azaltıcı etkisi olan maske kullanımının takibini insansız hava araçları ile tespitine odaklanan birtakım çalışmalar yapılmıştır (Akar, 2022; Jasim, 2022; Daşgın, 2023).

Akar (2022), derin öğrenme yöntemi kullanarak maske takma düzeyini incelemiştir. Araştırmada, birden fazla insansız hava aracının çektiği görüntüler kullanılarak açık ve kapalı mekanlarda maske kullanımına dikkat edilme düzeyi değerlendirilmiştir. Bu değerlendirmeyi gerçekleştirmek için YoloV4, VGG16, VGG19, ResNet101, ResNet152 ve MobileNetV1 gibi derin öğrenme modelleri uygulanmıştır. Elde edilen sonuçlar, gerçek zamanlı uygulamalar için en uygun sonuçların YoloV4 modeli ile elde edildiğini göstermiştir. Bu çalışma, maske takma durumunun otomatik olarak tespit edilmesinde derin öğrenme modellerinin etkili bir araç olduğunu ortaya koymuştur.

Jasim (2022), maske kullanım düzeyini incelemek için insansız hava aracı yerine güvenlik kameralarını kullanan bir çalışma gerçekleştirmiştir. Bu çalışmada, derin öğrenme yöntemleri arasında SSD ve MobilNetV2 modelleri tercih edilmiştir. Test değerlendirmelerinde, MobilNetV2 modeliyle doğruluk değeri oran olarak %99,72 ve f1 ölçüm değeri %100 olan

yüksek güvenilirlik düzeyinde sonuçlar elde edilmiştir. Bu çalışma, güvenlik kameraları aracılığıyla maske kullanımının tespit edilmesinde derin öğrenme yöntemlerinin etkili bir şekilde kullanılabileceğini göstermiştir. Elde edilen sonuçlar, maske kullanımının doğru ve güvenilir bir şekilde belirlenmesinde MobilNetV2 modelinin başarılı olduğunu göstermektedir.

Devrim (2019), bireylerin yüz ifadeleri görüntüleri ile duygusal durumlarının tespitini incelediği çalışmada yedi farklı duygunun görüntülerde derin evrimsel sinir ağları kullanmıştır. Geliştirilen gerçek zamanlı yüz ifadesi tanımlama sisteminde Imagenet veri seti ölçeğinde InceptionResNetV2'nin diğer modellerle kıyaslandığında daha iyi sonuç verdiğini tespit etmiştir.

Pandeminin etkilerini azaltmak ve hastalıklarla mücadeleyi sürdürebilmek için Covid-19 yayılımının kontrol altına alınmasında maske kullanım düzeyinin denetlenmesi önemli bir rol oynamaktadır. Maske kullanımının denetimi, teknolojik cihazlar aracılığıyla izlenmesi neredeyse bir gereklilik haline gelmiştir. Bu bağlamda, Covid-19 yayılımını en aza indirme amacıyla Yüz Maskesinin Evrimsel Sinir Ağı (CNN) modelleriyle tespitini hedefleyen Daşgın (2023), transfer öğrenme tabanlı modelleri kullanmıştır. Kaggle kütüphanesinde bulunan bir veri setindeki 906 resim kullanarak VGG19, DenseNet121, NasNetMobile, EfficientNetV2M ve InceptionV3 modellerini değerlendirmiştir. Test sonuçlarına göre, NasNetMobile yöntemiyle yapılan analiz sonucunda doğruluk değeri %99.35, kesinlik değeri %99, geri çağırma değeri %99 ve f1 ölçüm değeri %99 olan sonuçları elde ederek en iyi başarı oranını sağlamıştır. Bu çalışma, derin öğrenme modellerinin kullanılarak maske tespitinin etkili bir şekilde gerçekleştirilebileceğini ve Covid-19 yayılımının kontrolünde önemli bir araç olabileceğini göstermektedir.

Danaei Mehr (2017), akciğer tomografi görüntülerini Data Science Bowl ve Kaggle veri tabanlarını kullanarak kanser teşhisinde evrimsel sinir ağları yöntemiyle sağlıklı ve hastalıklı olarak kategori etmiştir. AlexNet ve GoogleNet mimarileri ile sınıflandırıldığında GoogleNet %96.360 doğruluk oranı ile daha yüksek doğruluk oranını verdiği görülmüştür. Böylece akciğer kanserinin erken teşhisi ile doktorlara yardımcı olacak bir sistem geliştirilerek alana katkı sağlanmıştır.

Konvansiyonel sinir ağları ile bitki yapraklarının sağlıklı olma düzeylerini inceleyen Uysal (2020), zeytin bitkisini ele aldığı çalışmada 3400 veri setini üç farklı sınıflandırmada kategorize etmiştir. Halkalı leke hastalığı, pasakarı zararlısı ve sağlıklı olmak üzere 3

kategoride incelenen veri seti CNN modeli %94 doğruluk oranı ile başarılı bir şekilde sınıflandırılarak hastalıklı yaprakların teşhisi ve tedavisine yönelik acil önlem işlemlerinin başarıyla gerçekleştirilmesine zemin hazırlamıştır.

Özaydın (2022), çiftçilerin tarım alanlarında verimliliği artırmalarına yardımcı olmak amacıyla tohum ve fidelerin sınıflandırmasına odaklandığı çalışmada, 12 farklı fidenin 4750, 9 farklı tohumun ise 5285 veri setinde incelemesini yaparak InceptionResNETv2 modelinde doğruluk değeri %96,55 tespit edilmiştir. Böylece tarımsal faaliyetlerin iyileştirilmesinde çiftçiler açısından tohum ve fide sınıflandırmasında kolaylık sağlanmasına yardımcı olacak bir çalışma yapılmıştır.

Sürdürülebilir enerji kaynaklarından olan güneşten enerji elde edilmesi için kullanılan panellerde enerji verimliliğinin artırılması amacıyla hasarlı olanların tespiti ve onarımının yapılması önem arz etmektedir. Geleneksel yöntemlerle tespitinin çok zor olduğu hasarlı güneş panellerinin derin öğrenme yöntemiyle tespitine odaklanan Koç (2022), elektrolüminans (EL) görüntülerinden 6528 adet monokristal ve polikristal yapıdaki farklı görüntüyü kullandığı çalışmada Xception, Vgg16, Vgg19, Resnet50, DenseNet201 ve MobileNet kullanarak başarılı tespitler yapmıştır.

Artan (2019), sigorta sektörü için önemli olan hasarlı araçların hasar düzeylerini belirlemede evrimsel sinir ağlarını kullandığı çalışmada kendilerinin elde ettikleri görüntülerde 3 hasar tespitine odaklanmıştır. Böylece sigorta şirketlerinin iş yükünü oluşturan küçük ama daha sık yaşanan hasar tespitinde meydana gelen zaman kaybının en aza indirilmesi amaçlanmıştır. Özellikle veri seti açısından problem yaşamayan büyük sigorta şirketlerinde bu sistemin kullanılabilirliğinin artacağı önerisinde bulunmaktadır.

Son yıllarda hayatımızın bir parçası haline gelen güvenlik tedbirlerinin özellikle kalabalık mekanların girişlerinde sıklıkla karşılaştığımız X-ray cihazları ile yasaklı nesnelerin (silah, bıçak, pense, İngiliz anahtarı ve makas) tespitinin yapılması geleneksel yöntemlerden ziyade daha pratik sonuçları olduğu kabul edilmektedir. X-ray görüntülerinde yasaklı nesne tespitinin geliştirilmesine katkıda bulunan Sorgun (2022), çalışmada 100 görüntü üzerinde Faster R-CNN algoritmasının ResNet50, ResNet101, ResNet152 ve Inception ResNet V2 modellerini kullanmış, sonuç olarak Faster R-CNN ResNet101 modeli 0,5 IoU için %90,6 başarı düzeyiyle zaman ve doğruluk açısından en uygun model olduğunu tespit etmiştir.

Literatürde yer alan görüntü algılama teknolojileri ile endüstriyel açıdan hataların tespitine yönelik çalışmalardan görüldüğü üzere birçok farklı derin öğrenme modelleri kullanılmıştır.

Buradan hareketle görüntü işleme yöntemleri ile protez ayak üretim sürecinin geliştirilmesine yönelik alan henüz literatürde çalışılmadığını söylemek mümkündür. Bu çalışma literatürde tespit edilen bu boşluğu doldurmak amacıyla InceptionResNetV2, Xception, NasNet Mobile, DenseNet201, EfficientNetV2S ve InceptionV3 derin öğrenme modelleri kullanılarak protez ayak üretim sürecinin geliştirilmesine katkı sağlanması amaçlanmıştır.



3. TEMEL KAVRAMLAR

3.1. Pedilen Ayak

Diz altı amputelerde protez bileşenleri arasında ayak önemli bir yere sahiptir. Pedilen ayağın ilk amacı, anatomik ayak-ayak bileği rolünü ortaya koymaktır. (Kapp vd., 1992; Sin vd., 2001).

Pedilen ayakların, insan ayağına; anatomik, fonksiyonel ve estetik görünüm açısından mümkün olabildiğince benzemesi beklenmektedir. Bu sebeple bir pedilen ayağın topuk kısmında şok absorpsiyonu, farklı zeminlere uyumu, yürüme kabiliyetinin uygun bölümlerinde itme ve sallanma aşaması kısalmalıdır. Hafif ve uzun ömürlü pedilen ayak üretimi önemlidir. (Versluys vd., 2009).

3.1.1. Pedilen Ayak Özellikleri ve Çeşitleri

Protezde anatomik ayak ve ayak bileğine benzeyen bir pedilen ayak üretim sistemi oluşturmak önemli derecede zordur. Pedilen ayağın normal ayak ölçülerinde ve uygun ağırlıkta olması gerekir. Bu amaç dikkate alınarak geliştirilmeye çalışılan pedilen ayakların sahip olması gereken özellikleri eklem hareketinin sağlanması, şokların absorbe edilmesi, desteğin sabit bir yüzeyden sağlanması, kas fonksiyonlarının yerine getirilmesi ve estetik görünümde olmasıdır.

Diz altı amputelerde temel olarak 5 farklı protez ayak tasarımı bulunmaktadır:

- 1- Solid Anide Cushion Heel (SACH) Ayak
- 2- Konvansiyonel Ayak (Tek eksenli Ayak-Single Axis Foot)
- 3- Çok Eksenli Ayaklar (Multiaxial Ayaklar)
- 4- Ayak Bileği Oynamayan Esnek Omurgalı Ayaklar; Kingsley Stored Energy ve İDİO Dinamik Ayak.
- 5-Enerji Depolayan Ayaklar (Dynamic Elastic Response-DER) (Robert vd., 1996).

Sach Ayak; 1949 yılında geliştirilen bir tür protez ayaktır. Bu ayak, ayak bileğindeki hareketin kısıtlı olduğu ve topuk bölgesinde bulunan bir lastik malzemeden dolayı sert bir ayak bileği ve yumuşak bir topuk yapısına sahiptir. Solid Anide Cushion Heel adı verilen ve sert ayak bileği ile yumuşak topuk anlamına gelen bu tanımlama, kısaltması olan SACH ayak olarak kullanılmıştır (Şener ve Erbahçeci, 2001; Rihs ve Polizzi, 1998; Czerniecki vd., 1991).

Günümüzde, yaygın olarak kullanılan protez ayak türleri arasında, hafif, dayanıklı, kolayca tamir edilebilen, maliyet açısından ekonomik ve estetik açıdan tatmin edici özelliklere sahip olan SACH ayak öne çıkmaktadır (Robert vd., 1996).

SACH ayak, içerisinde bir tahta omurgayı barındıran, esnek bir parçanın çevrelediği omurgaya sahip bir protez ayaktır. Ayrıca, ayak ön kısmına uzanan bir plastik parça, ayak bağlantı vidası ve sıkışabilme özelliği olan bir topuk lastiği de içermektedir (Şener ve Erbahçeci, 2001; Czerniecki vd., 1991; Lehmann vd., 1993). SACH ayak, şokların emilmesi sürecinde topuk lastiğinin yoğunluğu ve amputenin vücut ağırlığı arasındaki ilişkiye dayanmaktadır. Yürüme esnasında, SACH ayakta etkili olan ana bileşenler, omurga ve topuk lastiğidir. Topuk lastiği, yumuşak, orta ve sert olmak üzere üç farklı kategoride bulunmaktadır. Amputenin kilosu, yaşının yanı sıra amputasyon seviyesi, güdük boyu ve aktivite düzeyi, topuk lastiğinin seçimi açısından önem taşımaktadır (Şener ve Erbahçeci, 2001; May, 1996; Bowker vd., 1992).

Konvansiyonel ayak; transvers eksen etrafında dorsifleksiyon ve plantarfleksiyon hareketlerine izin veren bir protez ayaktır. Bu ayak, sert bir tahta omurga ile metatars lastiğine kadar uzanır ve omurgayı çevreleyen kauçuk köpük bir parça, topuk kısmında bulunan kauçuk plantarfleksiyon tamponu ve kama şeklindeki yarı sert dorsifleksiyon durdurucu gibi bileşenlerden oluşur. Ayak plantarfleksiyon hareketine giderken, plantarfleksiyon tamponu sıkışarak hareketi sınırlar ve böylece dorsifleksiyon kaslarına destek olur. Dorsifleksiyon hareketi ise ön tarafta bulunan dorsifleksiyon durdurucu ile sınırlanır. Konvansiyonel ayaklar, sagittal düzlemde 15° plantar fleksiyon ve 3-50° dorsifleksiyon hareketine izin verir. Streslerin emilimi, topuk vuruşunda plantar fleksiyon lastiğinin sıkışmasıyla gerçekleşir (Şener ve Erbahçeci, 2001, May, 1996; Bowker vd., 1992; Culham vd., 1986).

Konvansiyonel ayaklar, duruş fazının başında topuk vuruşunu takip eden plantarfleksiyon hareketini kontrol edemezler. Ayağın ani olarak yere temas etmesi, dizde ekstansiyon ve buna bağlı olarak stabilizasyon sağlar. Dorsifleksiyon tamponu, plantarfleksiyon tamponundan biraz daha serttir ve orta duruş ve itme fazında aşırı dorsifleksiyonu engeller. Duruş fazında dorsifleksiyonun artması, dizde fleksiyon momentini artırır. Bu artan fleksiyon momenti denge ve stabilizeyi azaltır. Tamponların sertliği, amputenin ağırlığı ve aktivite düzeyi gibi faktörlere göre ayarlanmalıdır (May, 1996; Prince vd., 1998).

Çok eksenli ayaklar; Greissenger ve çift eksenli ayaklar gibi gruplara ayrılır ve tek eksenli ayakların özelliklerine ek olarak inversiyon ve eversiyon hareketlerine izin verirler. Bu lateral

yönde gerçekleşen hareketler ve fransvers, sagital ve vertikal düzlemde ortaya çıkan kuvvetleri absorbe ederek engelbeli zeminlerde daha iyi bir stabilite ve kontrol sağlarlar. Ayrıca, yaygın skar dokusu olan güdükleri olan ve obez olan amputelerde, şok absorpsiyonunun iyi olması nedeniyle tercih edilebilirler. Bununla birlikte, ayaktaki fazla hareketlilik, statik koşullarda stabiliteyi azaltabilir. Çok eksenli ayakların dezavantajları ise ağırlığı, tamirinin zorluğu ve maliyetidir (Şener ve Erbahçeci, 2001, Robert vd., 1996).

Ayak Bileği Oynamayan Esnek Omurgalı Ayaklar; Kingsley stored-energy (STEN) ve Otto-Bock İDİO, dinamik ayaklar kategorisine giren ayak türleridir. Bu ayaklar, SACH ayaklara benzerlik gösterirken, esnek bir ön omurgaya sahiptirler. Çok eksenli ayaklardan daha hafif olan bu ayaklar, inversiyon, eversiyon ve transvers rotasyona izin vermezler. Bu ayaklar, geç duruş fazında şok emilimini sağlar ve aktif veya obez amputeler için faydalı olabilir (Robert vd., 1996).

Dinamik ayak; içerisinde bir tahta omurgayı barındıran, omurgayı çevreleyen yüksek dansiteli poliüretan köpükten oluşan ve parmak uçlarına kadar uzanan bir bileşen içeren bir protez ayaktır. Ayrıca, yumuşak ve geniş poliüretan bir topuk kaması ve orta dansiteli poliüretan köpükten oluşan bileşenler de bulunur. Omurganın kısa olması, düz veya topuklu ayakkabıların kullanımında gereken ayar değişikliğine otomatik olarak uyum sağlamaktadır. Bu ayak, kullanılan malzemenin özelliği nedeniyle yürüme sırasında ayaktaki yuvarlanma hareketini kolaylaştırır ve şok absorpsiyonunu gerçekleştirir. Ayrıca, hafif, esnek ve maliyeti düşüktür (Şener ve Erbahçeci, 2001).

Sten ayak; altı bileşenli bir tasarıma ve SACH ayağa benzeyen bir görünüme sahiptir. Üç adet tahta blok, iki adet esnek kauçuk blok, SACH topuk kaması ve bu bileşenlerin etrafını saran poliüretan köpükten oluşur (Şener ve Erbahçeci, 2001).

Enerji Depolayan Ayaklar; Carbon Copy II, Carbon Copy II Light, Carbon Copy III, Flex Walk, Flex ayak, Quantum ayak, Seattle ayak, Seattle Light ayak ve Spring Light ayaklardır (Robert vd., 1996).

Enerji depolayıp serbest bırakabilen bu ayaklar, yüksek aktivite düzeyine sahip amputeler için idealdir. Bu protez ayaklar, yürüme ve koşma esnasında ön ayaktaki kuvvetlerin kontrolünü sağlar. Amputenin dakikadaki adım sayısı arttıkça, topuk bölgesinde geçirilen süre azalırken, ön ayak üzerinde daha fazla yüklenme gerçekleşir. Ön ayağa etki eden kuvvetlerin artması, dorsifleksiyon momentini de artırır (Şener ve Erbahçeci, 2001; Robert vd., 1996).

Dorsifleksiyon momenti, omurganın yapısal değişikliklere uğramasıyla birlikte enerji depolanmasını ve itme fazında bu enerjinin serbest bırakılmasını sağlayarak vücudun sagittal düzlemde ileriye hareket etmesine olanak tanır. Enerji depolayabilen ve serbest bırakabilen bu ayaklar, amputenin doğal ve normal bir şekilde yürüyebilmesinde büyük öneme sahiptir (Şener ve Erbahçeci, 2001; Robert vd., 1996).

Protez kullanıcılarının ampütasyon seviyeleri, mesleki gereksinimleri, işlevsellik düzeyleri, yaşları ve kiloları, protez ayak seçimi aşamasında önem taşımaktadır. Türkiye'de, yerli ve yabancı üretim olmak üzere farklı markalarda protez ayaklar kullanılmaktadır. Ekonomik olması ve Sosyal Güvenlik Kurumu (SGK) ödeme listesinde yer almasından dolayı, Solid Ankle Cushion Heel – 11 Sabit Ayak Bileği Yumuşak Topuk (SACH) ayak, dinamik ayak ve konvansiyonel ayak gibi protez ayak çeşitleri ülkemizde en çok tercih edilenler arasındadır. Bu tercihler, protez ayakların uygun maliyetli olması ve SGK tarafından desteklenmesi gibi faktörlere dayanmaktadır (Topuz vd., 2010).

3.1.2. Pedilen Ayak Üretim Süreci

Pedilen ayakların üretiminde kullanılan hammaddelerden biri poliüretandır. Poliüretan, organik üniteler zinciri üzerinde karbamat bağlantılarıyla birleştirilen polimerik bir malzemedir. Genellikle PU olarak kısaltılan poliüretan, esnek köpükler ile esnemeyen köpükler, performansı yüksek yapıştırıcılar, dayanıklı elastomerler, sentetik olan lifler, pedilen ayaklar, contalar, halıların alt kısmı ile sert plastiklerin imalatında kullanılır. PU, çeşitli endüstrilerde yaygın olarak kullanılan çok yönlü bir malzemedir ve özellikle pedilen ayakların üretiminde sağlamlık ve elastikiyet gereksinimlerini karşılamak için tercih edilmektedir.

PU malzemeden mamul pedilen ayaklar, üretimdeki teknik yöntem ve kullanılan malzemeden kaynaklanan topuk esnekliği ve enerji depolama özelliğinden kaynaklı protez kullanıcılarına rahat bir yürüyüş deneyimi ortaya koyar. Modüler ve klasik protezlerde kullanılmak üzere dizayn edilmiştir.

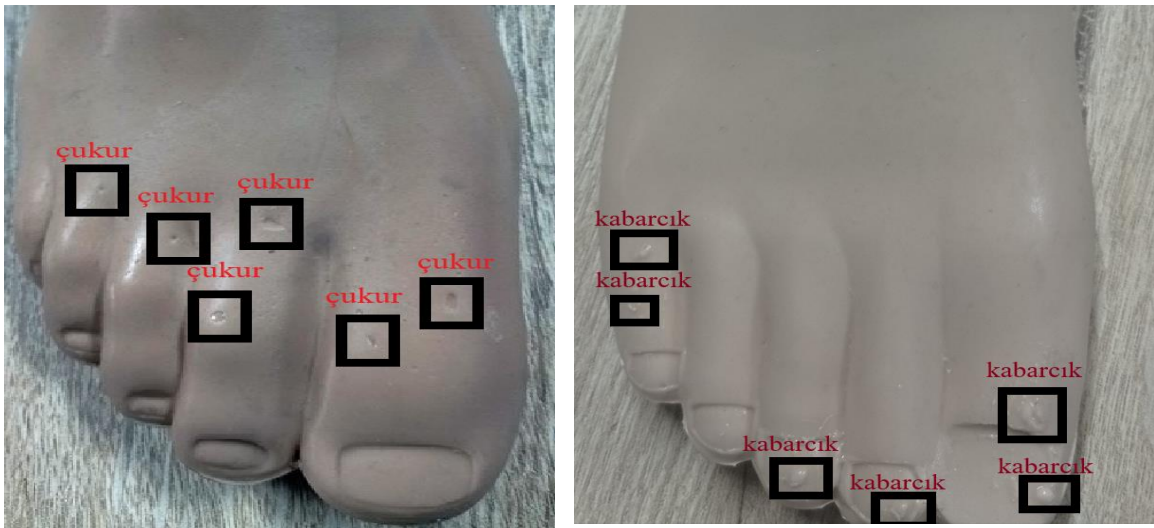
Veri setimizin temin edildiği pedilen ayak üretimi yapan firmadaki pedilen ayakların üretim sonrası hatalı veya hatasız olduğu ile hatalı olarak tanımlananların çukur hatalı veya kabarcık hatalı olduğu firmanın kalite kontrol uzmanları tarafından belirlenmektedir.

3.2. Nesne Tespiti

Nesne tespiti, bilgisayar görüşü ve yapay zeka alanlarında kullanılan bir tekniktir. Temel olarak, dijital görüntülerde veya videolarda nesnelerin tanınmasını ve sınıflandırılmasını hedefler.

Son dönemde, bilgisayarla görme ve görsel algılama gibi alanlar, farklı teknolojilerin ilerlemesiyle birlikte artan ilgi görmektedir. Nesne tespiti de iş ortamlarında ve kişisel kullanımlarda giderek yaygınlaşan bir gereklilik haline gelmiştir. Tıbbi görüntüleme, hareketli görüntü yakalama tarımsal sektörler, istihbarat ve güvenlik gibi birçok alanda nesne tespiti sıkça kullanılmaktadır.(Alkhder, 2021).

Nesne tespiti, bir görüntüde bulunan birden fazla nesnenin algılanması ve ardından bu nesnelerin etiketlenmesi işlemidir. Bu süreçte görüntüdeki farklı öğelerin algılanması ve belirlenmesi, nesne tespiti olarak adlandırılmaktadır (Alqadiri, 2022). Bilgisayarın görsel verileri analiz ederek görüntülerdeki nesneleri algılama ve tanımlama yeteneği, bilgisayarla görme alanında önemli bir beceri olarak kabul edilir. Bu beceri bilgisayarın görsel verileri işleme ve anlama yeteneği olarak tanımlanabilir. Bu yetenek görüntülerdeki nesneleri algılayarak tanımlama işlemini gerçekleştiren bir algoritma veya sistem tarafından sağlanır (Alkhder, 2021). Nesne tespitindeki temel amaç, veri setimizdeki hatalı ürünlere ait görüntülerin yer aldığı Şekil 1’de de görüldüğü gibi görüntülerde bulunması gereken nesnelerin varlığını tespit etmek, bu nesnelere ait konumları ve sınıfları belirlemektir. Bu süreç, hayali dikdörtgen sınırlayıcıların görüntüyü taramasıyla gerçekleştirilerek, nesnelerin tespit edilmesini sağlayan bir sistem oluşturmayı hedeflemektedir.



Şekil 1: Hatalı Pedilen Ayak Görüntülerinin Hata Türüne Göre Nesne Tespiti Örnekleri

Nesne tespiti işlemi, nesnelerin konumlarını bulma ve nesneleri sınıflandırma görevlerini içeren iki aşamadan oluşur. Nesne tespiti, birden fazla nesnenin tanınması ve tanımlanmasıyla ilgilenirken, aynı zamanda nesnelerin bulunduğu konumu ve hangi sınıfa ait olduklarını belirlemeyi de amaçlar (Alqadiri, 2022).

Hareket eden nesnelerin sayılması ve tespiti, çeşitli yöntemler aracılığıyla gerçekleştirilmektedir. Bu yöntemler optik akış, ardışık çerçeveler arasındaki fark ve arka plan çıkarımı olarak gruplandırılmaktadır (Balaji ve Karthikeyan, 2017; Luo vd., 2014). Optik akış, piksel seviyesinde yoğunluğu tahmin ederek hareketli nesneleri tespit etmek için kullanılan bir yöntemdir. Bu yöntem, nesne tespiti için önceden herhangi bir bilgiye ihtiyaç duymaz. Kamera hareket halindeyken iyi sonuçlar vermesine rağmen, gerçek zamanlı sistemler için çok fazla hesaplama gerektirdiğinden uygun değildir. Çerçeve farkı yöntemi ise ardışık iki çerçeve arasındaki piksellerin yoğunluk değişimine dayanarak hareket tahmini yapar. Bu yöntem, dinamik ortamlarda iyi sonuçlar verirken, arka plan ve ön alanın renklerinin birbirine yakın olduğu durumlarda yanlış sonuçlar üretebilir ve tespit edilen nesnelerde boşluklar oluşabilir. Arka plan çıkarımı yöntemi, hareketli veya sabit bir nesnenin tespiti için etkili bir yöntemdir. Nesne tespiti, arka plan ile ön alan arasındaki farklılıklara dayanarak gerçekleştirilir. Arka plan çıkarımının başarımı, arka plan modelleme ve güncelleme yöntemine bağlıdır (Karaköse vd., 2016)

Bir videodaki veya görüntüdeki nesnenin tespiti için, görsel öznitelikler ve hareket bilgileri olmak üzere iki temel bilgi kullanılmaktadır. Görsel öznitelikler, renk, doku, şekil gibi görsel özellikleri ifade ederken, hareket bilgileri ise nesnenin hareket durumuyla ilgili bilgileri içerir. Bu bilgilerin edinilmesi ve kullanılması, belirli bir işlem gücü ve işlem zamanı gerektirir. Ayrıca bu bilgileri tek başına kullanan yöntemler olduğu gibi, özellikle nesne takibi ve tespitinin zor olduğu durumlarda farklı öznitelikleri ve hareket bilgilerini birlikte kullanan yöntemler de bulunmaktadır. Bu yöntemler, birden fazla görsel özniteliği ve hareket bilgisini entegre ederek daha güvenilir ve hassas sonuçlar elde etmeyi amaçlamaktadır (Luo vd., 2014). Nesne tespiti için kullanılan bilgilerin çeşitliliği, başarı oranını artırdığı gibi işlem süresinde de bir artışa neden olmaktadır. Bu durum, gerçek zamanlı uygulamalarda nesne takibi için son derece önemli bir sorun oluşturmaktadır (Luo vd., 2014; Risha ve Kumar, 2016).

Bir görüntüde nesne tespiti için kullanılan geçici değişiklikler yöntemi, yüksek hata oranlarına neden olabilmektedir. Bu nedenle, daha kullanışlı ve başarılı olan diğer yöntemlerden biri, belirlenmiş nesnenin olduğu görüntüden sadece arka plan görüntüsünün

çıkarılmasıdır. Sonraki adımda, elde edilen fark görüntüsü belirli bir eşik değerine tabi tutulur ve istenmeyen nesneler elemine edilmektedir (Aldhaheri ve Edirisinghe, 2014; Hardas ve Vibha, 2015).

Nesneleri özellikleri dikkate alınarak gruplandırmak amacıyla sınıflandırma işlemi uygulanmaktadır. Sınıflandırma işlemi sırasında nesnenin benzersiz öznitelikleri kullanılmaktadır (Hanbay ve Üzen, 2017). Doku, gradyan, zaman, renk ve hareket, nesne sınıflandırmada en yaygın olarak kullanılan öznitelik türleridir. Renk temelli sınıflandırma, etkili bir yöntem olmasına rağmen ışık değişimlerine duyarlı olması nedeniyle hesaplama süresi artmaktadır. Doku, hesaplama süresi ve doğruluk oranı yüksek olan bir öznitelik türüdür. İnsan takibi için gradyan ve hareket öznitelikleri kullanılabilir. Ancak gürültüye karşı hassas olabilirler. Hareket ve zamansal değişimlerde hesaplama süresi arttığından dolayı maliyet yüksektir (Tan vd., 2021).

Nesne tespit algoritmaları, temelde geleneksel yöntemler ve derin öğrenme yöntemleri olmak üzere iki kategoriye ayrılmaktadır. Derin öğrenmeye dayalı nesne tespit yöntemleri, hız ve doğruluk açısından geleneksel yöntemlere göre daha üstün sonuçlar sunmaktadır (Tao vd., 2017)

4. YÖNTEM VE MİMARİ

4.1. Veri Kümesinin Hazırlanması

Pedilen ayak üretimi yapan firmadan temin edilen veri kümesine Python kodları aracılığıyla döndürme, parlaklık derecesini artırma, filtreleme ve yeniden boyutlandırma gibi görüntü çoğaltma işlemleri uygulanması sonucu elde edilen 1200 görüntü veri setimizi oluşturmaktadır. Temin edilen pedilen ayakların hatalı veya hatasız olduğu ile hatalı olarak tanımlanan pedilen ayakların çukur hatalı veya kabarcık hatalı olduğu firmanın kalite kontrol uzmanları tarafından belirlenmiştir.

Çalışmamızda 300 çukur hatalı, 300 kabarcık hatalı olmak üzere toplam 600 hatalı ve 600 hatasız pedilen ayak resimlerinden oluşan veri kümesi kullanılmıştır. Veri kümesinde 210 çukur hatalı, 210 kabarcık hatalı olmak üzere toplam 420 hatalı ve 420 hatasız pedilen ayak görüntüleri eğitim için, 90 çukur hatalı 90 kabarcık hatalı olmak üzere toplam 180 hatalı ve 180 hatasız pedilen ayak görüntüleri test için yer almaktadır. Şekil 2, veri setimizde bulunan görüntülerden bazı örnekleri içermektedir.



Şekil 2: Veri Setimizdeki Örnek Hatalı ve Hatasız Pedilen Ayak Görüntüleri

Şekil 3, firmadan temin edilen veri kümesine Python kodları aracılığıyla döndürme, parlaklık derecesini artırma, filtreleme ve yeniden boyutlandırma gibi görüntü çoğaltma işlemleri uygulanması sonucu elde edilen görüntülerden bazı örnekleri içermektedir.



Şekil 3: Görüntü Çoğaltma İşlemleri Sonrası Elde Edilen Pedilen Ayak Görüntüleri

4.2. Derin Öğrenme

İlk önce yapay zeka alanında ortaya çıkan ve sonrasında makine öğrenimi teknikleriyle güçlendirilen bir öğrenme yaklaşımıdır. Bu yaklaşım, çok katmanlı hesaplama modelleri kullanarak verilerin soyutlama düzeylerini öğrenmeye olanak tanımaktadır. Konuşma tanıma, görsel nesne tanıma, nesne algılama gibi birçok alanda teknolojiye önemli gelişmelere imkan sağlamıştır. Dolayısıyla, veri girişi olarak verilen bilgileri hızlı bir şekilde işleyerek özellikleri ve görevleri algılayabilen bir öğrenme yöntemidir. Derin öğrenme, önceki çalışmaları da içine alarak geniş bir kapsama sahiptir ve karmaşık veri setlerinin analizini ve anlamlandırılmasını amaçlayan bir alandır (LeCun ve diğerleri, 2015).

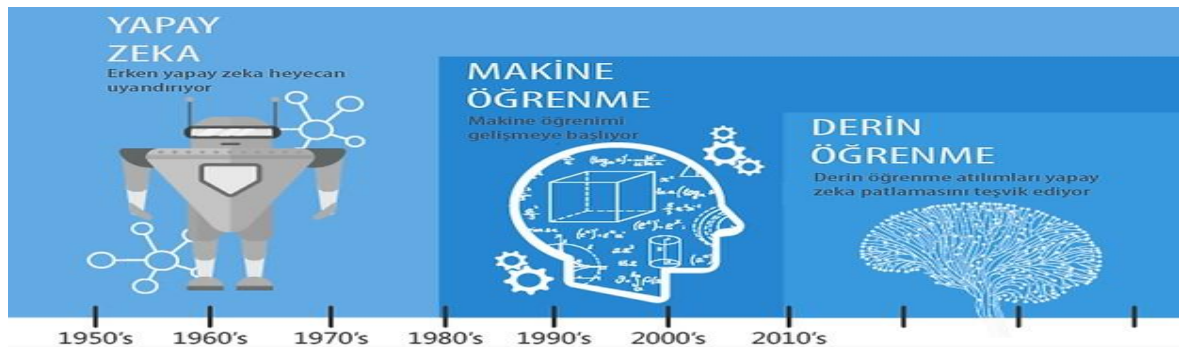
Derin öğrenme, yapay zeka ve makine öğrenimi disiplinlerinin birleşiminden ortaya çıkan ve daha gelişmiş bir yapıya sahip olan bir alt alan olarak kabul edilmektedir. Derin öğrenmenin, makine öğrenimine kıyasla daha yüksek bir esneklik ve doğru model oluşturma yeteneği olduğu bilinmektedir. Bu yöntem, karmaşık veri setlerini analiz etme ve anlama konusunda önde olan bir yaklaşımdır (Militante & Dionisio, 2020).

Derin öğrenme teknikleri, genellikle kompleks ve büyük veri kümelerinden anlam ifade eden bilgilerin çıkarılması adına sıklıkla kullanılmaktadır. Bu yöntemler, derin sinir ağları ve diğer ileri öğrenme algoritmaları aracılığıyla, veri setlerindeki gizli desenleri keşfetme ve karmaşık ilişkileri modelleme yeteneğine sahiptir. Derin öğrenme sayesinde, büyük veri kaynaklarından anlamlı bilgiler çıkarılarak, tahminlerde bulunma, sınıflandırma, tanıma ve diğer görevleri gerçekleştirme imkanı sağlamaktadır (Zhang, 2021).

Derin öğrenme, beyin nöronlarının yapısından esinlenerek geliştirilen bir yöntemdir. Bu yöntem, sinir ağlarının çok katmanlı yapısını kullanarak bilgi işleme ve öğrenme yeteneklerini modellemeyi amaçlar. Beyindeki sinir ağlarındaki nöronların etkileşimi ve bağlantılarına benzer şekilde, derin öğrenme algoritmaları da çok katmanlı yapılarda bulunan yapay sinir ağlarını kullanarak verileri işleyip özellikleri çıkarmak ve karmaşık ilişkileri öğrenmek için tasarlanmıştır. Bu sayede, derin öğrenme, büyük veri setlerindeki desenleri tanımlama ve karmaşık görevleri gerçekleştirme konusunda üstün performans sergileyebilmektedir (Militante & Dionisio, 2020).

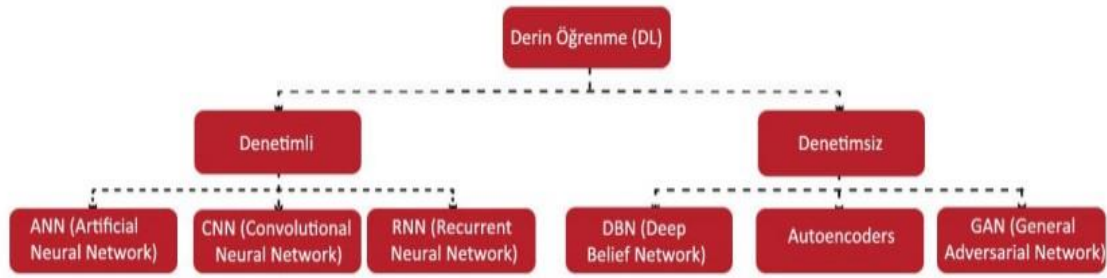
Makinelerin insan benzeri görevleri yerine getirebilmesi, kendi kendine öğrenebilmesi düşüncesiyle birçok teori ve yöntemler geliştirme ve boyutu büyük veri setlerinden özellikleri çıkarma yollarını keşfetme amacıyla çalışmaktadır. Bu çerçevede, yapay zeka ve makine öğrenimi alanları, makinelerin karmaşık görevleri gerçekleştirme yeteneklerini geliştirmek için çalışmaktadır. Makinelerin öğrenme süreci, örnek verilere dayanarak modeller oluşturma, desenleri tanıma, karar verme ve tahmin yapma gibi işlevleri içerir. Bu süreçte, geniş veri kümelerinden anlamlı özelliklerin çıkarılması ve makinenin kendini geliştirmesi hedeflenmektedir (Jasim, 2022).

Özellik çıkarımının yanı sıra, dönüştürme işlemlerinin gerçekleştirilmesi için birden fazla doğrusal olmayan katmanın bileşimi kullanılmaktadır. Bu yapı, her katmanın çıktısını bir önceki katmanın girdisi olarak kullanarak işlem yapmaktadır. Bu sayede, işlenmemiş veri kümesinden otomatik olarak özellikler çıkarılır. Bu çok katmanlı yapının her bir aşaması, veriye katman katman uygulanan dönüştürmeleri gerçekleştirir ve daha karmaşık ve yüksek düzeyli temsillerin oluşmasını sağlamaktadır. Bu süreçte, verinin özgün formundan yola çıkarak daha anlamlı ve özelleştirilmiş bir temsilin elde edilmesi amaçlanmaktadır (Kaya & Önal, 2021). Şekil 4'te Yapay Zeka, Makine Öğrenmesi ve Derin Öğrenme gelişim süreci ve aşamaları yer almaktadır.



Şekil 4: Yapay Zeka, Makine Öğrenmesi ve Derin Öğrenme (Bakshi, 2019)

Derin öğrenme mimarileri tıpta ve tarımda olduğu gibi birçok farklı alanlarda geniş bir kullanım alanına sahiptir. Bu alanlarda, derin öğrenme yöntemleri, hastalıkların tanımlanması, tespiti ve meyve hastalıklarının belirlenmesi gibi birçok uygulamada yaygın bir şekilde kullanılmaktadır. Derin öğrenme, büyük veri setlerinin analiz edilmesinde ve karmaşık desenlerin tanımlanmasında kritik bir rol oynamaktadır. Tıbbi görüntüleme yöntemlerinde, derin öğrenme algoritmaları kanser tespiti, radyolojik analiz ve hastalık teşhisi gibi alanlarda etkili bir araç olarak kullanılmaktadır. Tarım sektöründe ise derin öğrenme, bitki hastalıklarının erken teşhisi, verimlilik analizi ve ürün kalitesi değerlendirmesi gibi konularda başarılı sonuçlar sağlamaktadır (Militante & Dionisio, 2020). Uzmanlık alanına göre iki çeşit olan derin öğrenmenin türleri denetimli ve denetimsiz olarak ifade edilmektedir (Şekil 5). Çalışmamızda derin öğrenme en başarılı yöntemi olan ve denetimli sınıfına giren CNN modelleri ile analiz yapılmıştır.



Şekil 5: Derin Öğrenme Türleri (Deng & Yu, 2014)

4.3. Evrişimsel Sinir Ağı (CNN-Convolutional Neural Network)

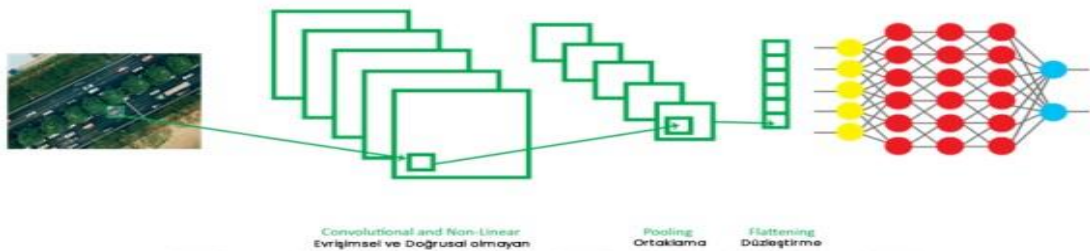
1998 yılında LeCun, Bottou, Bengio ve Haffner tarafından gerçekleştirilen araştırma, evrişimli sinir ağlarının ve aynı zamanda LeNet modelinin ilk kez ortaya çıktığı önemli bir dönüm noktası olmuştur. Bu çalışma, sinir ağları alanında büyük bir ilerleme olarak kabul edilmekte ve modern derin öğrenme yöntemlerinin temelini atmıştır. LeCun ve ekibi, evrişimli sinir ağlarıyla nesne tanıma ve sınıflandırma gibi görevlerde olağanüstü başarı elde etmek için LeNet modelini geliştirmiştir. Bu model, girdi verilerini katmanlar halinde işleyen ve her katmanda filtrelerin öğrenilmesini sağlayan bir yapıya sahiptir. LeNet, rakam tanıma gibi basit görüntü işleme görevlerinde büyük bir etki yaratmış ve daha sonra birçok ileri seviye derin öğrenme modelinin temelini oluşturan bir prototip haline gelmiştir. LeCun, Bottou, Bengio ve Haffner'ın bu çalışması, evrişimli sinir ağlarının alanındaki önemli bir dönüm noktasını temsil etmekte ve günümüzde hala büyük bir etki yaratmaktadır (Simonyon & Zisserman, 2015).

LeNet'in ilk çıktığı dönemde, posta kodları ve rakamların tanınması için tasarlanmış bir görevi başarılı bir şekilde yerine getirmiştir. Ancak bugün geldiği noktada, LeNet çok daha gelişmiş bir model haline gelmiştir. Milyarlarca parametreyi hesaplayabilen ve verilerden anlam çıkararak karmaşık sorunlara çözümler sunabilen bir güce sahiptir. Artık sadece basit tanıma işlemlerini gerçekleştirmekle kalmayıp, derin öğrenme alanında bir dönüm noktası olarak kabul edilmektedir. LeNet, karmaşık veri setlerini analiz edebilmekte, desenleri tanımlayabilmekte ve tahminlerde bulunabilmektedir. Bu gelişmeler, yapay zeka ve bilgisayar görüşü alanında devrim niteliğindedir ve LeNet'in başarıları, yapay zekanın potansiyelini göstermektedir (Akar, 2022).

Evrişimsel sinir ağları, bilgisayarla görme alanında büyük bir öneme sahiptir ve görüntü tabanlı uygulamalarda büyük başarılar elde etmiştir. Evrişimsel sinir ağları, görüntü verilerini işlemek ve özelliklerini çıkarmak için özel bir mimari kullanmaktadır. Bu sayede, karmaşık görsel problemleri çözmekte ve yüksek doğruluk oranlarıyla sonuçlar üretmektedir. Yüz tanıma uygulamalarında, evrişimsel sinir ağları yüz özelliklerini algılayarak benzersiz tanımlama yapabilmektedir. Nesne tanıma konusunda ise evrişimsel sinir ağları, farklı nesneleri tanıyabilme yeteneğiyle ön plana çıkmaktadır. Görüntü sınıflandırma alanında, bu sinir ağları görüntülerdeki özellikleri analiz ederek doğru sınıflandırma yapabilmektedir. Sürücüsüz araç yapımı için ise evrişimsel sinir ağları, çevre görüntülerini işleyerek araçların etrafındaki nesneleri algılayabilmekte ve güvenli bir şekilde hareket etmelerini sağlayabilmektedir. Hastalık tespiti konusunda, evrişimsel sinir ağları, medikal görüntüler üzerinde analiz yaparak hastalıkların erken teşhisini sağlayabilmektedir. Bu yüksek başarılarından dolayı, evrişimsel sinir ağları, bu alanlarda tercih edilen bir yöntem olmuştur (Akar, 2022; Militante & Dionisio, 2020).

Evrişimsel sinir ağları, girdi olarak aldıkları görüntüler üzerinde etkileyici bir başarı sergileyen derin öğrenme modelleri olarak bilinmektedir. CNN veya ConvNet olarak da adlandırılan bu sinir ağları, bir görüntüde bulunan farklı nesneleri başarıyla ayırt edebilme yeteneğine sahiptir. Görüntülerdeki özellikleri hiyerarşik olarak öğrenerek, nesnelerin tanınması ve sınıflandırılması konusunda olağanüstü sonuçlar elde edebilmektedir. Evrişimsel sinir ağları, daha düşük seviyedeki özelliklerden daha yüksek seviyedeki soyut kavramlara kadar geniş bir aralıkta bilgi öğrenebilmektedir. Bu sayede, görüntü tabanlı görevlerde, özellikle nesne tanıma ve sınıflandırma gibi karmaşık problemlerde en başarılı modellerden biri olarak kabul edilmektedir. Evrişimsel sinir ağlarının bu özellikleri, derin öğrenmenin etkileyici gücünü ve potansiyelini ortaya koymaktadır (Kaya & Önal, 2021).

CNN (Convolutional Neural Network), derin öğrenmenin mimarilerinden biri olan yapay sinir ağlarına benzer şekilde çok katmanlı bir yapıya sahiptir. Girdi, birden çok gizli ve bir çıktı katmanından oluşmaktadır. Evrişimli sinir ağlarının genel yapısı incelendiğinde, temel olarak üç ana katmandan oluştuğu görülür: evrişim katmanı, havuzlama katmanı ve tam bağlı katman. Evrişim katmanı ve havuzlama katmanı, girdi olarak alınan görüntüden özellik çıkarımı yapmaktadır. Bu katmanlar, görüntülerdeki desenleri algılayarak önemli özellikleri vurgulamaktadır. Tam bağlı katman ise, çıkan özellikleri çıktı katmanına bağlar ve sonuçları elde eder. Bu yapı, evrişimli sinir ağlarının gücünü ve etkinliğini sağlamak için optimize edilmiştir. Her katman, önceki katmanın çıktısını alır ve ardışık olarak verileri işler, bu da ağın derinliğini ve karmaşıklığını artırır. CNN'in bu yapısal özellikleri, görüntü tabanlı problemlerde etkili bir şekilde çalışmasını sağlar ve yüksek başarı oranları elde etmesini mümkün kılar (Özbay & Altunbey Özbay, 2020). Şekil 6'da Evrişimsel Sinir Ağı mimarisinin yapısı gösterilmektedir. Tam bağlantılı katman, bir önceki katmandaki nöronların kendisine bağlanmasıyla oluşur. Bu bağlantılar, sınıflandırma işlemi için sonraki katmandaki diğer nöronlara veri aktarımını sağlar. Tam bağlantılı katman, girdi verilerini düzenler, özellikleri vektörler halinde temsil eder ve bu temsilleri kullanarak sınıflandırma ve tahminleme yapmaya hazırlık gerçekleştirir. Bu katman, önceki katmandaki çıktıların toplanmasını, birleştirilmesini ve işlenmesini sağlar. Böylece, verilerin daha derin ve kapsamlı bir şekilde analiz edilmesi ve sonuçların elde edilmesi mümkün olur (Harriat Christa vd., 2021). Evrişimli sinir ağları (CNN), derin öğrenme mimarisinde diğer modellerden ayrılmaktadır. Çünkü görüntünün kendisini girdi olarak alır ve görüntünün tamamı yerine etkili özellikleri bulmak için de çalışmaktadır. Bu sayede, CNN görüntünün lokal özelliklerini tanımlayabilmekte ve daha yüksek seviyeli öznitelikler oluşturabilmektedir. Akademik bir tabirle ifade edecek olursak, CNN, her bir katmanda özellik çıkarımı yaparak görüntüdeki önemli desenleri ve yapıları belirlemektedir. Bu özellikler, daha sonra sınıflandırma, tespit veya diğer görevler için kullanılabilir. Bu özelliği sayesinde, CNN modelleri, görüntü işlemede etkili bir araç olarak kabul edilmektedir (Mehedi Shamrat vd., 2021).



Şekil 6: Evrişimsel Sinir Ağı Mimarisinin Yapısı (Rubik's Code, 2018)

4.3.1. Evrişim Katmanı

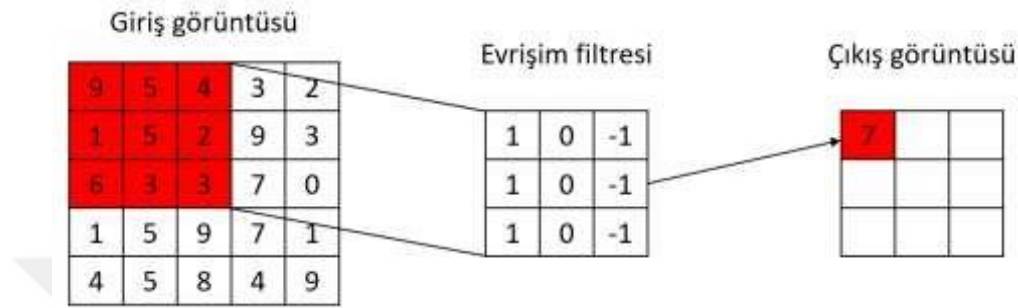
Derin öğrenme, RNN (Recurrent Neural Networks) ve CNN (Convolutional Neural Networks) olmak üzere iki temel algoritma üzerine kurulmuştur. RNN'ler genellikle işlem sırasını takip etmek ve bir sonraki adımı tahmin etmek için kullanılırken, CNN sınıflandırma algoritması olarak yaygın bir şekilde kullanılmaktadır. Özellikle görüntü işleme, nesne tanıma ve resim sınıflandırma gibi alanlarda etkin bir şekilde kullanılmaktadır. CNN, çok katmanlı bir yapay sinir ağıdır ve genellikle görüntü analizi ve nesne tespiti için kullanılmaktadır. Hayvanların görü sistematiğine benzer bir yaklaşımla geliştirilmiştir. Filtreleme katmanları kullanılarak oluşturulan bir yapıya sahiptir. Belirlenen amaca göre özellik çıkarmak için belirli filtreler kullanılmaktadır. Özellikle sınıflandırma çalışmalarında başarılı sonuçlar elde edilmiştir (Doğan ve Türkoğlu, 2019).

CNN mimarisinin temel yapı taşı olan bu katman, görüntü üzerinde özelliklerin algılanması için kullanılan filtre veya çekirdeklerle ve özellik haritaları bileşenleriyle işlev görmektedir (Akar, 2022). CNN mimarisinin temel birimi olan bu filtreler, girdi olarak sağlanan görüntüden daha küçüktür ve giriş görüntüsünün önemli noktalarının filtrelendiği özellik haritasını belirlemek için kullanılmaktadır (Jasim, 2022). Bu katman, her filtre için girdi görüntüsü üzerinde bir kaydırma işlemi yaparak her bir bölgedeki özellikleri algılar ve ardından özellik haritalarını oluşturur. Bu özellik haritaları, daha sonra ağı daha derin katmanlarında kullanılmak üzere diğer katmanlara aktarılmaktadır. Bu sayede, CNN mimarisi, karmaşık özellikleri algılamak ve daha yüksek seviyeli temsiller oluşturmak için filtrelerin ve özellik haritalarının birleşik kullanımını sağlamaktadır.

Bu katman, girdi görüntüsünü birleştirerek sonucu bir sonraki katmana ileten bir aktarım işlemi gerçekleştirir. Aktarım sırasında, belirli boyutlarda farklı filtreler kullanılarak görüntü üzerinde hareket ettirme işlemi gerçekleştirilir. Bu filtreler genellikle 2x2, 3x3 veya 5x5 gibi boyutlara sahip olabilmektedir. Görüntü üzerindeki hareket işlemi, filtrelerin tüm görüntüye uygulanmasıyla gerçekleştirilmektedir (Kaya & Önal, 2021). Bu işlemler sayesinde evrişim katmanı, girdi olarak sunulan görüntüden anlamlı bilgilerin çıkarılması ve elde edilen özellik haritasının oluşturulması hedeflenmektedir (Özbay & Altunbey Özbay, 2020). Evrişim katmanı, görüntünün özelliklerini vurgulayarak daha yüksek seviyeli temsiller oluşturmada ve ağı daha derin katmanlarında kullanılmak üzere bu özellikleri aktarmaktadır.

Şekil 7'de gösterilen çıktı, giriş görüntüsünün 5x5 boyutunda bir matrisine, 3x3 boyutunda bir evrişim katmanı filtresi uygulanarak elde edilir. Filtre uygulama işlemi, filtrenin dikey ve

yatay yönde hareket ettirilerek girdi görüntüsü üzerinde gerçekleştirilir ve bu işlem tüm görüntü üzerinde genel olarak etkinleştirilir. Şekilde sunulan işlem, filtrelerin görüntünün farklı bölümlerinde uygulanarak özelliklerin belirlenmesini sağlar. Evrişim katmanı, ağın ilerleyen katmanlarında daha karmaşık özelliklerin öğrenilmesine katkıda bulunan bir yapı taşıdır (Türkoğlu vd., 2020).



Şekil 7: Görüntü Evrişim Aşamaları ve Çıktı Görüntüsü Elde Etme

4.3.2. Düzeltilmiş Doğrusal Birim Katmanı (ReLU)

Düzeltilmiş doğrusal birim katmanı (ReLU-Rectified Linear Unit), derin öğrenme ve yapay sinir ağlarında yaygın olarak kullanılan bir aktivasyon fonksiyonudur. ReLU fonksiyonu, girdinin sıfırdan büyük olduğu durumlarda giriş değerini direkt olarak çıktı olarak verirken, sıfırdan küçük olduğu durumlarda ise çıktıyı sıfır olarak belirlemektedir. Aktivasyon fonksiyonu, derin öğrenme modelleri arasında verilere bağlı olacak şekilde çıktı üretebilmek amacıyla kullanılan bir mekanizmadır. Bu amaca yönelik olarak, önceki çalışmalarda çoğunlukla tercih edilen düzeltilmiş doğrusal birim katmanıdır (Alqadiri, 2022).

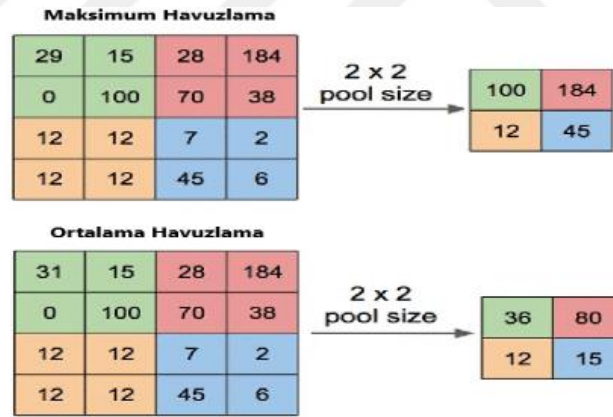
Öğrenme sürecinde, doğrusallık gerektirmeyen problemlerde de etkili bir performans sergileyen ReLU, öğrenme işleminin gerçekleştirilmesi için gereklidir. Bu aktivasyon fonksiyonu, doğru olmayan özellikleri yakalayabilme yeteneği sayesinde geniş bir uygulama alanına sahiptir. Doğrusal olmayan durumlar için ideal bir seçenek ve öğrenme sürecinde önemli bir role sahip olan düzeltilmiş doğrusal birim katmanı, (Maas vd., 2013) Eşitlik 1'deki belirtildiği gibi matematiksel olarak ifade edilmektedir (Harriat Christa vd., 2021).

$$f(x) = \text{maksimum}(0, x) \quad (1)$$

4.3.3. Havuzlama Katmanları

Derin öğrenme mimarilerinde veri boyutunu azaltmak ve çıktıyı özetlemek amacıyla kullanılan bir bileşen havuzlama katmanıdır. Bu katman genellikle evrişim tabanlı

modellerde kullanılır ve öznitelik haritalarının boyutunu küçültmeye yardımcı olmaktadır. Havuzlama katmanı, öznitelik haritalarının boyutunu azaltarak hesaplama yükünü azaltırken aynı zamanda özelliklerin öğrenilmesini kolaylaştırmakta ve bu katman, özellik çıkarım sürecindeki aşamalar arasında yer alır ve genellikle evrişim katmanlarıyla birlikte kullanılarak derin öğrenme modellerinin performansını artırmaktadır. Hesaplama işlemleri hızlı bir şekilde gerçekleştirilen havuzlama katmanlarında kendinden önce olan katmandan aktarılan özelliklerle ilgili haritaları gruplandıran yeni bir nitelik haritaları oluşturulmak amacıyla işlenmektedir (Jasim, 2022). Dolayısıyla derin öğrenme modellerinin etkili bir şekilde çalışmasına katkıda bulunmakta ve veri işleme sürecini optimize etmektedir. Katman temel amacına uygun olarak tüm özellikleri kaybetmeden dizinin boyutunun sıradaki katman için azaltılma işlemi gerçekleştirmektedir. İşlemler tamamlandığına boyutta meydana gelen azaltma işlemi bilgi kaybına da sebep olabilmektedir. Aslında meydana gelen bilgi kaybı durumu olumsuz gibi anlaşılabilir da bu aşamadaki işlem sonraki katmanlar adına hesaplama miktarını azalttığından sistemde fazla öğrenmeye vakit harcamayı engelleyerek zaman kazandırdığından için avantajlı hale gelmektedir (Kaya & Önal, 2021).



Şekil 8: Ortalama Ortaklama ve Maksimum Ortaklama Katman Yapısı (Doğan, 2020)

Şekil 8'de Ortalama Ortaklama ve Maksimum Ortaklama olarak havuzlama katmanının yapısını gösteren iki farklı tür bulunmaktadır.

Maksimum ortaklama katmanı: Yüksek verim göstermesi nedeniyle havuzlama katmanı, genellikle geliştiriciler tarafından yoğun bir şekilde kullanılmaktadır. Bu katmanda, filtrelerin taranması işlemi sırasında çıktı değerleri, görüntüdeki piksellerin en yüksek değerine göre seçilmektedir. Havuzlama katmanı, derin öğrenme modellerinde sıkça tercih edilen bir bileşendir. Bu katmanda, filtrelerin yardımıyla görüntü üzerinde taranma işlemi gerçekleştirilir ve her bir taranma bölgesinden en büyük değer seçilerek çıktı oluşturulur. Bu işlem, görüntünün önemli özelliklerini koruyarak veri boyutunu azaltmada etkilidir. Yani,

piksellerin en yüksek değerine dayalı olarak seçilen değerler, görüntünün belirgin özelliklerini yansıtır ve önemsiz ayrıntıları filtreler. Böylece, özneliklerin özetlenmesi ve veri boyutunun azaltılması sağlanmaktadır (Akar, 2022).

Havuzlama katmanı, derin öğrenme modellerinin başarısını artıran bir bileşen olarak kabul edilmektedir. Yüksek performansı ve veri işleme sürecindeki etkinliği nedeniyle, geliştiriciler tarafından sıklıkla kullanılmaktadır. Bu katman, modelin genel performansını iyileştirirken hesaplama yükünü azaltmakta ve eğitim sürecini hızlandırmaktadır.

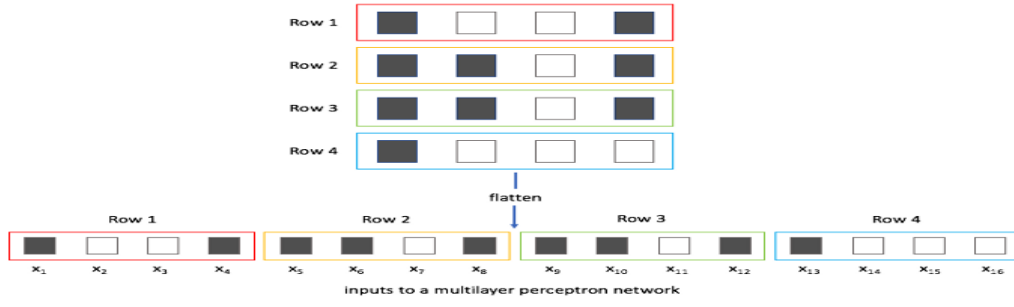
Ortalama ortaklama katmanı: Filtrelerin özellikleri taraması sırasında, çıkış aşamasına sevk edilen değerler seçilen bölgenin ortalama değeri tespit edilerek belirlenmekte ve bu aşamada yapılan işlem sonradan gelecek ve yeni olan dizideki bütün hücrelere uygulanmaktadır.

Havuzlama katmanında gerçekleştirilen tarama işlemi esnasında, filtreler belirli bir bölge üzerinde hareket etmekte ve bu bölgedeki değerleri incelemektedir. Seçili olan bölgenin ortalaması hesaplanmakta ve bu değer çıkış işlemine gönderilmektedir. Bu işlem, yeni bir dizi oluşturulurken tüm hücrelere uygulanmaktadır. Bu işlem, veri boyutunu azaltırken önemli özellikleri koruma amacı taşımaktadır. Seçilen bölgenin ortalaması, bu bölgenin temsil edici bir özelliğini yansıtmakta ve yeni dizideki hücrelere aktarılmaktadır. Böylece, verinin temel özellikleri özetlenmekte ve daha küçük bir veri boyutu elde edilebilmektedir (Jasim, 2022).

4.3.4. Tam Bağlantılı Katmanlar

Tam bağlantılı katmanlar, sinir ağlarında ardışık bir dizilimde yer alır ve nöronlar, kendisinden önceki katmandaki işlemlere tamamen bağlıdır. Evrişim ve havuzlama katmanlarından elde edilen sonuçlar birleştirilerek, nesnenin tanımlanmasında kullanılan özellikler için etiket verisi oluşturulmakta ve nesnenin sınıfı belirlenmektedir (Arvas, 2022).

Tam bağlantılı katmanlar, sınıflandırma işlemini gerçekleştirmek için önceki katmanları birbirine bağlayarak değerleri ilişkilendirmekte ve bunları tek boyutlu bir yapıya dönüştürmektedir (Akar, 2022). Şekil 9'da gösterilen düzleştirme işlemi sonrasında, ağlar tarafından kullanılabilecek hale getirilen giriş verileri elde edilmektedir.

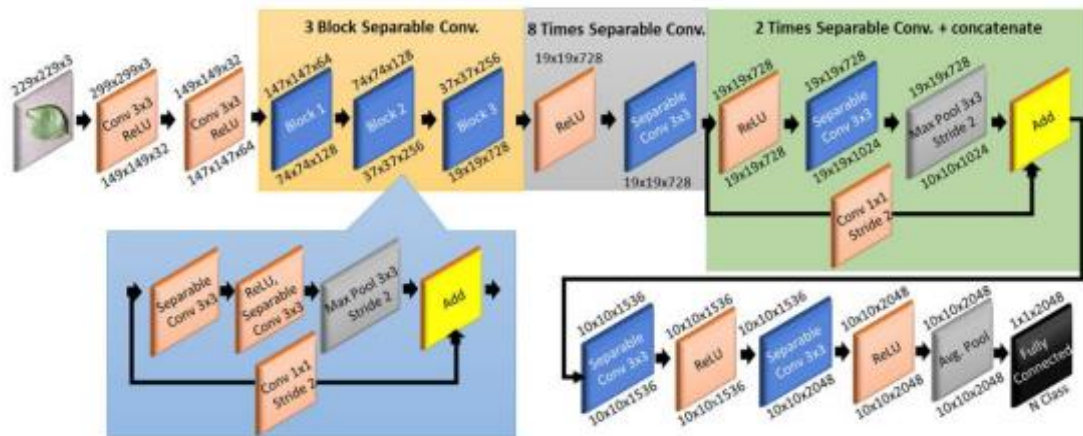


Şekil 9: Değerlerin Tam Bağlı Katman Aşamasında Kullanmayı Sağlayan Düzleştirme İşlemi (Konca, 2020)

4.4. Analiz Yapılan CNN Modelleri

4.4.1. Xception Modeli

Xception, Inception mimarisinin bir uzantısı olarak Francois Chollet tarafından tanıtılmıştır (Chollet, 2017). Bu derin evrişimli sinir ağı, yeni başlangıç katmanlarıyla birlikte gelmektedir (Rahimzadeh ve Attar, 2020). Xception veya Extreme Inception olarak da bilinen bu CNN modeli, 36 evrişim katmanına sahiptir ve özellik çıkarma bloğunun temelini oluşturmaktadır (Sutaji ve Yıldız, 2022). Derinlemesine ayrılabilir evrişim katmanlarının doğrusal bir yığına dayanan bu mimari, geleneksel Inception'ın daha gelişmiş bir versiyonunu sunmaktadır (Kassani ve ark., 2019). Xception, kalıntı bağlantılarını kullanır ve 3 akıştan oluşur: giriş akışı, orta akış ve çıkış akışı. Eğer giriş görüntüsü 299x299 boyutunda değilse, modele beslenmeden önce boyut ayarlaması yapılması gerekmektedir (S ve ark., 2019). Bu evrişim katmanları, ağı birbirinden ayırır ve birleştirir, böylece bir bütün oluştururlar (Chollet, 2017). Xception mimarisi Şekil 10'da gösterilmektedir.



Şekil 10: Xception Mimarisi (Konate, 2022)

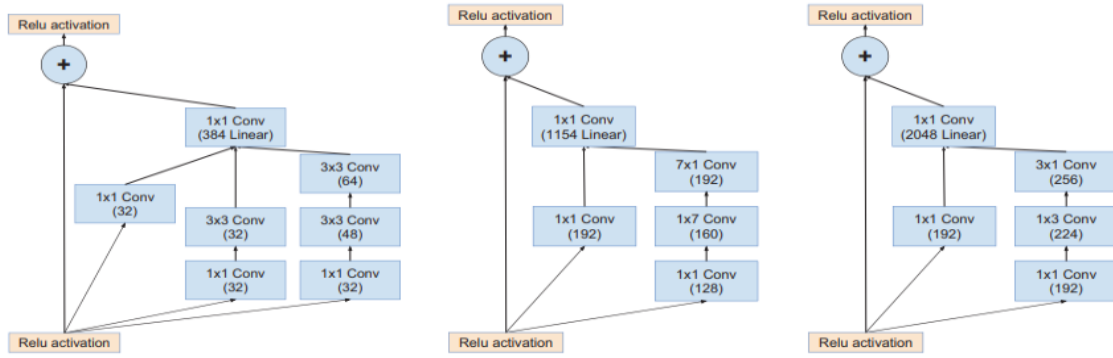
4.4.2. InceptionResnetV2 Modeli

Inception mimarisi, derin sinir ağıları için etkili bir yapı sağlamak amacıyla tasarlanmıştır. İlk olarak Inception V1 olarak adlandırılan model, paralel olarak çalışan farklı filtre boyutlarının kullanılmasıyla dikkat çekmektedir. Bu modelde, farklı boyutlardaki evrişim filtreleri aynı anda uygulanarak özellik haritalarının çıkarılması hedeflenmektedir. Böylece ağı farklı ölçeklerdeki özellikleri daha iyi yakalayabilmesi sağlanmaktadır. Inception modeli, ImageNet veri setinde sınıflandırma doğruluğunu en yüksek ve hata oranını en düşük olacak şekilde hesaplayan bir CNN mimarisidir. Paralel filtre kullanımı ve bilinçli boyut azaltma stratejileri, Inception modelinin hem etkili bir özellik çıkarma yeteneğine sahip olmasını hem de parametre sayısını kontrol altında tutarak hesaplama ve bellek verimliliğini artırmasını sağlamaktadır. Bu özellikleriyle Inception mimarisi, derin öğrenme alanında önemli bir başarı elde etmiş ve çeşitli görevlerde etkileyici sonuçlar vermiştir (Bozkurt, 2021).

Inception V2, orijinal mimarinin geliştirilmiş bir versiyonudur. Bu versiyonda, ağı performansını artırmak ve hesaplama maliyetini azaltmak amacıyla çeşitli iyileştirmeler yapılmıştır. İncelemeler ve geri bildirimler temel alınarak Inception V1 modeline yapılan değişikliklerle oluşturulmuştur.

Inception V2'de, farklı modifikasyonlar ve optimizasyonlar gerçekleştirilmiştir. Örneğin, daha küçük boyutlu filtrelerin kullanımı sayesinde ağ daha derin hale getirilebilmekte ve kapasitesi artırılabilir. Bu da daha karmaşık özellikleri öğrenme yeteneğini artırırken, hesaplama kaynaklarının daha verimli kullanılmasını sağlamaktadır. Aynı zamanda, parametre sayısı azaltılarak sınırlı kaynaklara sahip cihazlarda da daha etkin bir şekilde kullanılabilir hale getirilmiştir.

Inception V2, geliştirilen modifikasyonlar ve optimizasyonlarla birlikte orijinal mimarinin performansını artırmış ve hesaplama maliyetini azaltmış bir modeldir. Yapılan değişiklikler, derin öğrenme alanında etkileyici sonuçlar elde etmek ve çeşitli uygulamalarda kullanılabilirliği artırmak için yapılmıştır. Inception V2 modeli, tipik olarak görsel sınıflandırma, nesne tespiti ve görüntü segmentasyonu gibi görsel görevlerde tercih edilir. Bu model, geliştirilmiş özellik çıkarma kabiliyetleri ve daha az hesaplama maliyeti gibi avantajlarıyla dikkat çekmektedir (Ashames, 2020). InceptionResNet-v2 modelinin mimari yapısı Şekil 11'de gösterilmektedir.

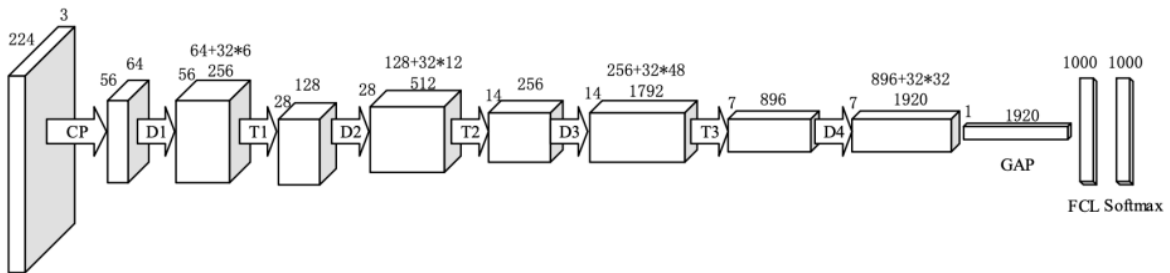


Şekil 11: InceptionResNet-v2 mimarisi (Szegedy vd., 2016)

4.4.3. DenseNet201 Modeli

Densenet201, derin öğrenme alanında önemli bir model olan Densenet'in bir varyasyonudur. Densenet201, Convolutional Neural Network (CNN) mimarisini temel alır ve yoğun bağlantılar aracılığıyla layerlar arasında bilgi akışını güçlendirmektedir. Bu model, daha önceki CNN modellerinden farklı olarak her layerda oluşan çıktıları doğrudan sonraki layerlara ileterek, daha derin bir ağ yapısına sahip olmayı hedeflemektedir. Bu bağlantılar, daha verimli bir bilgi paylaşımı sağlar ve ağın genel performansını artırmaktadır.

Densenet201, büyük bir veri kümesi üzerinde eğitilmiştir ve çeşitli görsel tanıma görevlerinde yüksek doğruluk oranları elde edebilmektedir. Yüksek yoğunluklu bağlantıları sayesinde, model, daha az parametre kullanarak daha fazla bilgi akışı sağlamakta, böylece daha hızlı ve daha etkili öğrenme gerçekleştirmektedir. Bu nedenle, Densenet201, görüntü sınıflandırma, nesne tespiti, segmentasyon gibi pek çok görsel işleme uygulamasında başarılı sonuçlar elde edebilen bir model olarak akademik araştırmalar ve uygulamalar için popüler bir seçenektir. Araştırmacılar, Densenet201'i derin öğrenme tabanlı projelerinde kullanarak daha keskin sonuçlar elde edebilmekte ve görsel veri analiziyle ilgili zorlu problemleri çözebilmektedir. DenseNet201 modelinin mimari yapısı Şekil 12'de gösterilmektedir.

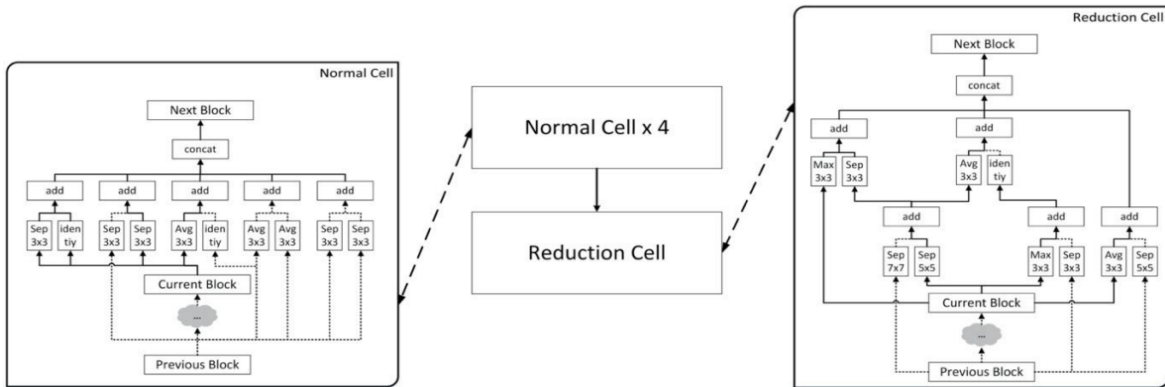


Şekil 12: DenseNet201 Modelinin Yapısı (Hayıt & Çınarer, 2022)

4.4.4. NasNetMobile Modeli

NasNetMobile, mobil cihazlar için özel olarak tasarlanmış olan NasNet mimarisinden türetilmiştir ve önceden eğitim için ImageNet veri tabanı kullanılmıştır (Herdian vd., 2019). NasNet, Nöral Mimari Arama (Neural Architecture Search) kavramının kısaltmasıdır. NasNet, birincil hücreleri içeren, destek eğitimiyle geliştirilen ayrılabilir evrişime sahip ve havuzlama gibi basit yapılardan oluşan ölçeklenebilir bir evrimsel sinir ağı mimarisidir. NasNet tabanlı mimariler, ağ kapasitelerine bağlı olarak bu basit blokların tekrarlanmasına dayalı bir yapıya sahiptir (Eryılmaz & Karacan, 2021). Model, 913 katmanlı bir yapıya ve 224x224 giriş görüntü boyutuna sahiptir (Kaçmaz, 2020).

NasNetMobile modelinin yapısı, 769 katman ve 224x224 giriş boyutunu kapsar. Bu model, 12 hücreden oluşmakta ve 5.4 milyon parametre kapasitesiyle 564 milyon çarpma kapasitesine sahiptir (Şekil 13). Bu bakış açısında, evrimsel olan ağ yapısının temel yapılandırma işlemi önceden tespit edilmektedir. Her biri aynı mimariye sahip ancak farklı ağırlıklara sahip birçok çoğaltılmış evrimsel birim bulunmaktadır. Ölçeklerin herhangi birindeki görüntüler için özellik eşlemesi yapmak için ölçeklendirilebilir bir mimari kullanılmaktadır. Evrimsel birimin iki önemli rolü vardır: aynı boyutta bir özellik haritası üretmek ve özellik haritasının yükseklik ve genişliğinin iki katı olan bir özellik haritası üretmek. Bu sayede basit bir şekilde ölçeklenebilir bir yapı elde edilmektedir (Ba Alawi & Qasem, 2021).



Şekil 13: NasNetMobile Modelinin Yapısı (Herdian vd., 2019)

4.4.5. EfficientNetV2S Modeli

2019 yılında Google Araştırma Ekibi tarafından geliştirilen bir CNN modelidir. Verimlilik ve doğruluk gibi alanlarda güzel geliştirmeler sunmaktadır. EfficientNetV2, EfficientNet serisinin devamı niteliğinde olan bir derin öğrenme modelidir. EfficientNetV2, önceki

versiyonlara kıyasla daha gelişmiş bir mimariye sahiptir ve daha yüksek performans sunmayı hedeflemektedir.

EfficientNetV2'nin temel özelliği, Compound Scaling adı verilen ölçeklendirme stratejisini kullanmasıdır. Bu strateji, ağın derinliği, genişliği ve çözünürlüğünü aynı anda optimize ederek daha iyi bir denge ve daha yüksek performans elde etmeyi amaçlamaktadır. Diğer CNN modellerinden farklı olarak, EfficientNet katman sayısı, çözünürlük derecesini veya derinlik ölçüsünü ayrı ayrı ayarlamaktansa, belirlediği bir oran ile tüm bu özelliklere aynı anda ayarlama yapmaktadır. Bu yöntem, daha az hesaplama gücüyle daha yüksek doğruluk elde etmeyi amaçlamaktadır. EfficientNet'in başarısı, çeşitli veri setleri üzerinde yapılan deneylerle kanıtlanmıştır. Model, daha az parametre kullanarak diğer CNN modellerinden daha iyi sonuçlar üretebilir ve ölçeklenebilirlik açısından avantaj sağlamaktadır. EfficientNet, derin öğrenme alanında önemli bir ilerleme olarak kabul edilmekte ve büyük veri setleriyle çalışan uygulamalarda etkileyici performans göstermektedir (Yavuz, 2020). EfficientNet, verimlilik ve doğruluk açısından optimize edilmiş bir yapıya sahip olup, performansı saniyede kayan nokta işlemleri (FLOPS) kullanılarak ölçülmektedir. Bu modelde, hesaplama gücünün etkin bir şekilde kullanılması hedeflenmiştir. EfficientNet, FLOPS metriği kullanılarak optimize edildiği için daha az kaynak kullanarak yüksek doğruluk elde etmek mümkün olmaktadır. Bu, veri işleme süresini ve enerji tüketimini azaltırken, aynı zamanda güçlü bir performans sunmayı sağlamaktadır. EfficientNet'in bu özellikleri, özellikle sınırlı kaynaklara sahip cihazlar veya uygulamalar için ideal bir seçenek haline gelmesini sağlamaktadır. Verimlilik ve doğruluğun dengesini sağlamak için saniyede kayan nokta işlemleri kullanımı, EfficientNet'in başarısının temel unsurlarından biridir (Hoang & Jo, 2021).

EfficientNetV2, görüntü sınıflandırması alanında daha iyi parametre verimliliği sağlayarak daha hızlı eğitim imkanı sunan bir modeldir. Modelin eğitim hızının ve boyut kapasitesinin optimize edilmesi için literatürde NAS olarak bilinen Sinir Mimari Araması (Neural Architecture Search-NAS) kullanılmıştır. EfficientNet'teki modelin son aşamasında bulunan parametre boyutu ve bellek erişim yükü, EfficientNetV2'de gerçekleştirilmektedir. Hatta, EfficientNetV2, 3x3'lük küçük bir yapı olan çekirdek boyutunu seçer ve bu tercihi, daha az alıcıya alan gerektirdiği için ekstra katmanların eklenmesini gerektirir. Bu özellikler, EfficientNetV2'nin daha etkili ve optimize edilmiş bir model olduğunu gösterir, daha az kaynak kullanarak yüksek performans sağlar ve eğitim süresini hızlandırır (Tan & Le, 2021; Yavuz, 2020).

EfficientNetV2-S, EfficientNetV2-M ve EfficientNetV2-L şeklinde ifade edilen farklı modellerinin, eğitim süreçlerinin ImageNet1K ve ImageNet21K veri setleri üzerinde yapılarak karşılaştırılması ile ilgili sonuç verileri aşağıdaki Tablo 1 ve Tablo 2'de sunulmuştur.

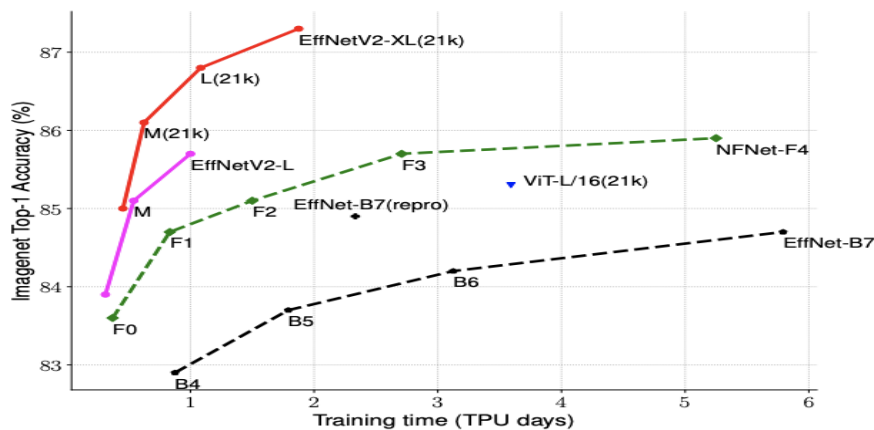
Tablo 1: ImageNet1K eğitim sonuçları (Tan, Le, 2021)

ImageNet1K Modelleri	Params	Top1 Acc.
EffNetV2-S	21.5M	83.9 %
EffNetV2-M	54.1M	85.2 %
EffNetV2-L	119.5M	85.7 %

Tablo 2: ImageNet21K önceden eğitilen ve ince ayarı yapılan modeller (Tan, Le, 2021)

ImageNet21K Modelleri	Finetuned ImageNet1K
EffNetV2-S	top1 = 84.9%
EffNetV2-M	top1 = 86.2%
EffNetV2-L	top1 = 86.9%

EfficientNetV2-S, EfficientNetV2-M ve EfficientNetV2-L olarak ifade edilen EfficientNetV2 ailesinin diğer modellerin, ImageNet üzerinde analiz edilen eğitim sonuçları incelendiğinde Şekil 14'teki gibi farklı doğruluk oranları elde ettikleri gözlemlenmektedir. Özellikle EfficientNetV2-M modelinde, şekillerle birlikte tablolar incelendiğinde, modelin az bir zaman içinde elde edilebilir doğruluk değerinin yüksek olduğu görülmektedir. Bu dengeli ilişki, eğitim süresi ile doğruluk arasındaki dengenin EfficientNetV2'nin etkileyici performansını göstermektedir. Modelin hızlı öğrenme yeteneği, verimli parametre kullanımı ve yüksek doğruluk elde etme potansiyeli, diğer EfficientNetV2 modellerinde de gözlemlenen özelliklerdir. Bu aile, çeşitli uygulamalarda yüksek performans ve verimlilik sunan güçlü bir görüntü sınıflandırma modelidir.



Şekil 14: EfficientNetV2 Modellerinin Eğitim Verimlilikleri (Tan & Le, 2021)

4.5. Başarım Ölçütleri

Analizlerde kullanılan modellerin performans durumlarının değerlendirilmesi, çeşitli performans olarak ifade edilen ölçütlerinin dikkate alınmasıyla gerçekleştirilmektedir. Bu performans ölçütleri, modelin ne kadar doğru sonuçlar ürettiğini belirlemek için kullanılmaktadır. En popüler performans ölçütleri arasında doğruluk, kesinlik, geri çağırma ve F1-skor bulunmaktadır. Doğruluk, modelin doğru sonuçları doğru bir şekilde sınıflandırma yeteneğini ölçmektedir. Doğruluk, toplam örneklerin içinde doğru sınıflandırılanların oranını temsil etmektedir. Yüksek bir doğruluk değeri, modelin genel olarak doğru sonuçlar ürettiği anlamına gelmektedir.

Kesinlik, modelin pozitif olarak sınıflandırdığı sonuçların ne kadarının gerçekten pozitif olduğunu belirlemektedir. Kesinlik, yanlış pozitif sonuçların yanlış bir şekilde pozitif olarak sınıflandırılma olasılığını değerlendirmektedir. Yüksek bir kesinlik değeri, modelin yanlış pozitif sonuçlar üretme olasılığının düşük olduğunu göstermektedir. Geri çağırma, gerçekten pozitif olan sonuçların ne kadarının doğru bir şekilde tespit edildiğini ölçmektedir. Geri çağırma, yanlış negatif sonuçların ne kadarını doğru bir şekilde pozitif olarak sınıflandırdığını değerlendirmektedir. Yüksek bir geri çağırma değeri, modelin yanlış negatif sonuçları minimize etme yeteneğine sahip olduğunu göstermektedir.

F1-skor, kesinlik ve geri çağırmanın dengeli bir kombinasyonunu temsil etmektedir. F1-skor, modelin hem yanlış pozitif sonuçları hem de yanlış negatif sonuçları minimize etme yeteneğini ölçmektedir. Yüksek bir F1-skor değeri, modelin hem kesinlik hem de geri çağırma açısından iyi bir performans sergilediğini gösterir. Bu performans ölçütleri, bir modelin sınıflandırma yeteneğini objektif bir şekilde değerlendirmek için kullanılmaktadır. Her bir ölçütün ayrı ayrı ele alınması ve değerlendirilmesi, modelin gücü ve zayıflıkları hakkında daha kapsamlı bir bilgi sağlamaktadır. Bu ölçütlerin kullanımı, modelin iyileştirilmesi ve optimize edilmesi sürecinde rehberlik sağlamaktadır (Zhang, 2021).

4.5.1. Doğruluk Ölçütü

Doğruluk, bir modelin tüm veri seti üzerinde doğru bir şekilde sınıflandırdığı örneklerin sayısını ifade etmektedir. Doğruluk, genellikle performans değerlendirmesinde yaygın olarak kullanılan bir ölçüttür. Ancak, doğruluğun kullanılması bazı durumlarda yetersiz olabilir, özellikle veri setinde sınıf dağılımı dengesizse. Eğer veri setindeki sınıflar arasında büyük bir dengesizlik varsa, yani bazı sınıflar diğerlerine göre daha fazla örneğe sahipse, doğruluk tek

başına performansın belirlenmesinde güvenilir bir ölçüt olmayabilir. Bu durumda, modelin çoğunluk sınıfına odaklanarak yanlış bir şekilde yüksek doğruluk elde edebileceği bir senaryo ortaya çıkabilmektedir.

Bu nedenle, doğruluk ölçütü yanında diğer performans metriklerinin de dikkate alınması önemlidir. Bunlar arasında kesinlik (precision), geri çağırma (recall) ve F1 skoru gibi ölçütler bulunur. Kesinlik, bir sınıf olarak tahmin edilen örnekler arasında gerçekten o sınıfa ait olanların oranını ifade eder. Geri çağırma, gerçekten o sınıfa ait olan örneklerin ne kadarının doğru bir şekilde tahmin edildiğini gösterir. F1 skoru ise kesinlik ve geri çağırmanın birleşimini temsil eder ve doğru sınıflandırma performansının bir ölçüsü olarak kullanılır. Bu ek performans ölçütleri, modelin sınıflandırma yeteneklerini daha ayrıntılı bir şekilde değerlendirmemizi sağlar. Özellikle dengesiz veri setlerinde, modelin daha az örneğe sahip olan sınıfları ne kadar doğru bir şekilde tahmin ettiğini belirlemek için bu metrikler kullanılır. Dolayısıyla, doğruluk dışında kesinlik, geri çağırma ve F1 skoru gibi ölçütlerin de performans analizinde göz önünde bulundurulması önemlidir. Bu metrikler, modelin performansını daha kapsamlı bir şekilde değerlendirmemize ve iyileştirmemize yardımcı olur.

Doğruluk değerinin hesaplaması Eşitlik 2’de ifade edilmiştir (Degadwala vd., 2021).

$$\text{doğruluk değeri} = \frac{TN + TP}{TN + TP + FP + FN} \quad (2)$$

4.5.2. F1 Skor Değeri

Geri çağırma ve kesinlik değerleri bulunduktan sonra hesaplanan F1-skor değerine, test kısmının doğruluğunun ölçüsü denir (Zhang, 2021). F1-skor, hem kesinlik hem de geri çağırma metriklerini birleştirerek bir sınıflandırma modelinin performansını değerlendirir. Yine diğer ölçütlerde olduğu gibi F1-skorun da 1’e yakın veya 1 olan değeri alması, modelin başarılı olduğunun göstergesidir. F1-skor değerinin en üst seviyede olması için ise yanlış-pozitif ve yanlış-negatif değerlerinin sıfır olması gerekir. F1 skor değerinin hesaplaması Eşitlik 3’te ifade edildiği gibidir.

$$F1 \text{ skor değeri} = 2 * \frac{\text{kesinlik} * \text{geri çağırma}}{\text{kesinlik} + \text{geri çağırma}} \quad (3)$$

F1-skor, hem kesinlik hem de geri çağırma değerlerini dikkate alarak bir denge noktası sağlamaktadır. Kesinlik, modelin pozitif olarak tahminlediği örneklerin gerçekten pozitif olma oranını ifade ederken, geri çağırma, gerçek pozitif örneklerin ne kadarının doğru bir şekilde saptandığını ölçmektedir.. F1-skor, bu iki metriği birleştirerek hem yanlış pozitifleri

hem de yanlış negatifleri minimize etmeye çalışmaktadır.

F1-skorun en yüksek değeri 1'dir ve bu, hem kesinlik hem de geri çağırma değerlerinin en yüksek seviyede olduğunu göstermektedir. Bu durumda, model hem pozitif örnekleri doğru bir şekilde tanımlamış hem de yanlış pozitifler ve yanlış negatifler en aza indirgenmiştir. F1-skor, sınıflandırma modelinin hassasiyetini ve kapsamını bir arada değerlendirerek modelin genel performansını belirlemektedir (Eşitlik 3).

4.5.3. Geri Çağırma Değeri

Sınıflandırıcı olan modelin performansını değerlendirmede önemli bir metrik olan geri çağırma (recall), tüm pozitif örneklerin doğru bir şekilde saptandığı durumu ifade etmektedir (Zhang, 2021). Geri çağırma değeri, Eşitlik 4'te yer alan formülü kullanarak hesaplanmaktadır. Bu formüle bakıldığında, kesinlik değerinde olduğu gibi geri çağırma değerinin de 1 veya 1'e yakın bir değer alması beklenmektedir. Dolayısıyla bunun için pay ve paydanın birbirine eşit olması gerekmektedir, bir başka ifadeyle yanlış-negatif değerinin 0'a yakın veya 0 olması gerekmektedir.

$$\text{Geri Çağırma Değeri} = \frac{TP}{TP + FN} \quad (4)$$

Geri çağırma, sınıflandırma modelinin gerçek pozitif örnekleri ne kadar başarılı bir şekilde tespit ettiğini göstermektedir. Bu metrik, sınıflandırma modelinin yanlış negatifleri minimize etme yeteneğini ölçmektedir. Yanlış negatifler, gerçekte pozitif olan örneklerin model tarafından yanlışlıkla negatif olarak tahmin edildiği durumları temsil etmektedir. Dolayısıyla, geri çağırma değeri yüksekse, modelin pozitif örnekleri doğru bir şekilde tespit etme yeteneği de yüksektir.

Geri çağırma değerinin 1 veya 1'e yakın olması, modelin pozitif örnekleri kaçırmadan doğru bir şekilde tanımlayabildiğini göstermektedir. Yanlış negatiflerin olabildiğince az olduğu durumda geri çağırma değeri en yüksek seviyededir. Bu nedenle, sınıflandırma modelinin geri çağırma değerinin yüksek olması, iyi bir performans sergilediğinin bir göstergesidir.

4.5.4. Kesinlik Değeri

Sınıflandırmanın iyi derecede olması durumunda, kesinlik değeri yüksek çıkarsa, bu durum sınıf tahmininin gerçek pozitif örnekleri doğru bir şekilde belirlediği anlamını taşımaktadır. Kesinlik değeri, 1'e yakın en yüksek değeri alır. Bu durumu ifade etmek için Eşitlik 5'te yer alan denklemi kullanılabilir.

$$Kesinlik\ Değeri = \frac{TP}{TP + FP} \quad (5)$$

Bu denklemde, yanlış pozitif sonucu 0'a yakın veya 0 olursa kesinlik değerinin en çok değeri alma ihtimali oluşmaktadır. Yanlış pozitif değerleri, gerçekte negatif olan örneklerin yanlışlıkla pozitif olarak tahmin edildiği durumları temsil etmektedir. Bu durumda, kesinlik değeri yüksek olabilir. Ancak gerçek pozitifleri doğru bir şekilde tahmin etme yeteneği düşük olabilmektedir.

Kesinlik, sınıflandırma modelinin pozitif olarak tahmin ettiği değerlerin gerçek pozitiflerle ne kadar uyumlu olduğunu göstermektedir. Yani, pozitif olarak tahmin edilen örneklerin ne kadarının gerçek pozitif olduğunu belirlemektedir. Eğer model, gerçek pozitifleri doğru bir şekilde saptayabiliyor ve yanlış pozitifleri minimum seviyede tutabiliyorsa, kesinlik değeri yüksek olur. Bu nedenle, kesinlik değerinin yüksek olması, sınıf tahmininde yanlış pozitif örneklerinin en aza indirildiğini ve gerçek pozitif örneklerin doğru bir şekilde tespit edildiğini göstermektedir. Kesinlik değerinin en yüksek değere ulaşması, iyi bir sınıflandırma performansının bir göstergesidir.

4.5.5. Karışıklık Matrisi Ölçütü

Analiz modellerinin eğitimi sırasında seçilen algoritmaların performansını değerlendirmek amacıyla kullanılan bir yöntem ve karışıklık ya da hata matrisi şeklinde ifade edilen bir algoritmadır. Bu matris, makine öğrenmesi veya sınıflandırma problemlerinde modelin doğruluk, kesinlik, geri çağırma gibi performans ölçütlerini görsel bir şekilde sunmaktadır. Karışıklık matrisi, gerçek etiketlerin ve algoritmanın tahmin ettiği etiketlerin bilinmesini gerektirir. Bu nedenle, veri setinin gerçek etiketlerinin bilinmesi ve algoritmanın bu verilere nasıl etiketleme yaptığının bilinmesi önemlidir. Karışıklık matrisi, gerçek ve tahmin edilen değerlerin karşılaştırılmasını kolaylaştırarak modelin performansını daha ayrıntılı bir şekilde analiz etmemizi sağlamaktadır.

Karışıklık matrisi genellikle bir kare matris olarak temsil edilmekte olup, satırlar gerçek sınıfları temsil ederken sütunlar tahmin edilen sınıfları temsil etmektedir. Her bir hücre, gerçek sınıf ve tahmin edilen sınıf kombinasyonuna karşılık gelen örnek sayısını içermektedir. Bu matris üzerinden, doğru sınıflandırılan örnekler, yanlış sınıflandırılan örnekler, yanlış pozitifler, yanlış negatifler gibi değerler hesaplanabilmektedir. Bu değerler, modelin performansını daha ayrıntılı bir şekilde değerlendirmemizi sağlamaktadır. Karışıklık matrisi, modelin hangi sınıfları ne kadar doğru tahmin ettiğini, hangi sınıfları yanlış tahmin

ettiğini ve hangi sınıflar arasında karışıklık yaşandığını görsel olarak göstermektedir. Bu matris, modelin gücünü ve zayıflıklarını anlamak ve iyileştirmek için önemli bir araçtır. Performans değerlendirmesi ve modelin ayarlanması sürecinde karışıklık matrisi kullanılarak modelin performansı optimize edilebilmektedir (Akar, 2022).

		GERÇEK	
		Pozitif	Negatif
TAHMİN	Pozitif	TP	FP
	Negatif	FN	TN

Şekil 15: İkili Sınıflandırmaya Ait Karışıklık Matrisi

Şekil 15'teki karışıklık matrisi ölçütlerindeki değerler aşağıdaki şekilde ifade edilmektedir (Güngör vd., 2021).

Doğru-pozitif (TP): Modelin doğru tahmin ettiği sonuçlar, yani tahminlerin pozitif olmasıdır.

Yanlış-Pozitif (FP): Gerçekte pozitif olan tahmin sonuçlarının model aracılığıyla yanlış tahmin edildiği sonuçlar, yani yanlış negatif tahminlerdir.

Doğru-Negatif (TN): Gerçekte negatif olan tahmin sonuçlarının model aracılığıyla doğru tahmin edildiği sonuçlar, yani doğru negatif tahminlerdir.

Yanlış-Negatif (FN): Gerçekte negatif olan tahmin sonuçlarının model aracılığıyla da yanlış tahmin edilmesinin sonuçlarıdır.

4.6. Çalışmada Kullanılan Hiper Parametreler

4.6.1. Epoch Değeri

Modelin öğrenme sürecinin optimize edilmesi veya en üst seviyeye çıkarılması için belirli bir sayıda epoch kullanılmaktadır. Epoch, veri kümesinde yer alan görüntülerin birden fazla kez işlenmesini ifade etmektedir. Eğitim sürecinin sonunda modelin istenilen doğruluğa ulaşabilmesi için epoch sayısının uygun bir şekilde seçilmesi önemlidir. Yüksek bir epoch seçimi, modelin fazla öğrenme yapmasına ve eğitim verilerine aşırı uyum sağlamasına neden olabilir. Bu durumda, model eğitim verilerine aşırı adapte olurken, genelleme yeteneği

azalır ve yeni veriler üzerinde başarısız olabilir.

Diğer yandan, düşük bir epoch seçimi, modelin yetersiz öğrenme gerçekleştirmesine ve istenilen doğruluğa ulaşamamasına yol açabilir. Modelin karmaşıklığını ifade eden öğrenme süreci tamamlanmadığından, modelin optimum performansına ulaşması mümkün olmayabilir. Bu nedenle, epoch sayısı modelin eğitimi sırasında dikkatlice seçilmelidir. İstenilen doğruluğa ulaşabilmesi için yeterli sayıda epoch kullanılmalıdır, ancak aşırı öğrenme riskini minimize etmek için de dikkatli olunmalıdır (Çarkacı, 2018).

4.6.2. Analizlerde Kullanılan Model Eğitimi (Training)

Model eğitimi, denetimli veya denetimsiz öğrenme ile verinin özellikleri arasındaki ilişkinin en üst seviyede matematiksel betimlemesini oluşturarak gerçekleştirme amacını taşımaktadır. Bu süreçte, kullanılan eğitim verilerinin kalitesi ve uygulamanın gereklilikleri performansın sonucunu belirlemektedir.

Model eğitimi, genellikle bir hedef fonksiyonunu optimize etme amacıyla gerçekleştirilir. Denetimli öğrenme durumunda, eğitim verileri ile etiketler arasındaki ilişkiyi yakalamak için bir öğrenme algoritması kullanılmaktadır. Bu algoritma, veri özelliklerini analiz ederek, girdi verilerine karşılık gelen doğru çıktıları tahmin etmeyi öğrenmektedir. Öte yandan, denetimsiz öğrenme yöntemleri, veri içerisindeki desenleri ve yapıları keşfetmek için kullanılır. Bu yöntemler, etiketlenmemiş verilerden öğrenerek veri setindeki yapıları ve desenleri yakalamayı hedeflemektedir.

Eğitim sürecinin başarıyla tamamlanması, doğru algoritma seçimi, uygun hiperparametre ayarlamaları ve yeterli miktarda temsil edici ve çeşitlilik içeren eğitim verilerinin kullanılmasını gerektirir. Eğitim verilerinin kalitesi, veri kümesinin temsil ettiği gerçek dünyadaki durumu doğru şekilde yansıtması, etiketlerin doğru ve tutarlı olması gibi faktörlere bağlıdır. Ayrıca, eğitim sürecinde kullanılan uygulamanın gereklilikleri, hedeflenen performans metriklerine uygun bir şekilde modelin eğitilmesini sağlamaktadır.

Eğitim sonunda elde edilen performans, modelin veriler üzerinde ne kadar iyi bir şekilde genelleme yaptığını ve hedeflenen görevi ne derece başarıyla yerine getirdiğini yansıtmaktadır. Performans, genellikle doğruluk, kesinlik, geri çağırma, F1-skor gibi metriklerle değerlendirilir. Ancak, performansın sadece eğitim verileri üzerinde değil, gerçek dünya verileri üzerinde de iyi olması önemlidir. Bu nedenle, modelin gerçek dünya verileriyle test edilmesi ve validasyon süreçlerinin gerçekleştirilmesi önemlidir (c3.ai, 2022).

4.6.3. Batches

Kullanılan modelin sinir ağı yapısından veri setinin tamamının aynı anda geçirilmesi zor olduğu için, eğitim işleminin daha yönetilebilir hale getirilmesi için veri seti batch olarak adlandırılan daha küçük parçalara bölünür. Batch değeri, her bir eğitim adımında işlenen örneklerin sayısını belirtmektedir. Batch değeri seçimi önemlidir, çünkü doğru bir denge bulunmalıdır. Çok küçük bir batch değeri, eğitim sürecini yavaşlatabilir ve işlemci kaynaklarının verimli kullanılmasını engelleyebilmektedir. Öte yandan, çok büyük bir batch değeri, bellek ve hesaplama gücü açısından yüksek taleplerde bulunabilmektedir.

Batch değeri 32 olarak belirlenip tez çalışması gerçekleştirilmiştir. Bu, her bir eğitim adımında 32 örneğin işleneceği anlamına gelmektedir. Bu değer, eğitim sürecinin verimli bir şekilde ilerlemesini sağlamak için uygun bir denge sunmaktadır. Sinir ağı modelinin iyi bir şekilde öğrenmesini ve istenen performans düzeyine ulaşmasını desteklemektedir (Alkhder, 2021).

4.6.4. Öğrenme Oranı (Learning Rate)

Tahminleme işlemi yapılırken hata durumlarının minimize edilmesi için öğrenme oranı (Learning Rate) kritik bir faktördür ve modelin yapılandırılmasında önemli bir rol oynamaktadır. Öğrenme oranı, modelin minimum hatayı elde etmek ve öğrenme sürecini hızlandırmak için ayarlanır. Ancak, Learning Rate değerinin çok büyük veya çok küçük seçilmemesi önemlidir.

Eğer öğrenme oranı çok büyük seçilirse, eğitim süreci gerçekleşebilir, ancak modelin güvenilirliği azalmaktadır. Büyük bir öğrenme oranıyla model, minimum hataya hızla yaklaşabilir, ancak bu yaklaşım sırasında aşırı atlamalar ve istikrarsızlık yaşayabilir. Bu durumda, modelin optimum çözümü yakalayamama riski bulunmaktadır.

Öte yandan, öğrenme oranı çok küçük seçilirse, model aşırı ayrıntılı ve yavaş bir eğitim süreciyle karşı karşıya kalmaktadır. Küçük bir öğrenme oranı, modelin ağırlıklarını yavaşça güncellemesine neden olmakta, bu da modelin hızlı bir şekilde toparlanmasını engeller ve eğitim süresini uzatır. Çalışmalarda genellikle standart bir öğrenme oranı kullanılmaktadır. Bu oran, 0.0001 gibi bir değere denk gelmektedir. Bu değer, çoğu durumda başarılı sonuçlar veren ve genel olarak kabul gören bir değerdir. Bu nedenle, bizim çalışmamızda da bu standart öğrenme oranını kullanmış bulunuyoruz. Sonuç olarak, uygun bir öğrenme oranı seçimi, modelin optimize edilmesi ve hata miktarının minimize edilmesi için kritik öneme

sahiptir. Doğru bir denge bulunmalı ve öğrenme oranı, modelin hızlı ve güvenilir bir şekilde eğitilmesini sağlamalıdır (Alkhder, 2021).

4.6.5. Optimizasyon Algoritması

Optimizasyon, belirli sınırlamalara ve parametrelere bağlı bir maliyet-kayıp fonksiyonunu en üst düzeye çıkarmayı amaçlayan bir süreçtir. Derin öğrenme uygulamalarında, öğrenme süreci aslında bir optimizasyon problemini içerir. Doğrusal olmayan bu tür problemlerin çözümünde, en uygun değeri bulmak için optimizasyon yöntemleri kullanılır. Farklı performans ve hız özelliklerine sahip olan stochastic gradient descent, adagrad, adadelata, adam ve adamax algoritmaları, yaygın bir şekilde kullanılan optimizasyon tekniklerindendir (Atcılı, 2020). Bizim çalışmamızda da iyi sonuçları hızlı bir şekilde veren ve derin öğrenme konularında son zamanlarda sıklıkla kullanılan Adam optimizasyon algoritması kullanılmıştır.

4.7. Derin Öğrenme Kütüphaneleri

Derin öğrenme kütüphaneleri, derin sinir ağlarının bilimsel hesaplamalarını etkin bir şekilde gerçekleştiren yapılar olarak bilinir. Derin öğrenme algoritmalarının uygulanması için Caffè, Torch, Theano, TensorFlow, Keras gibi pek çok popüler yazılım çatısı ve kütüphanesi mevcuttur. Bu çalışmada, karmaşık hesaplamaları başarılı bir şekilde gerçekleştirebilen Keras kütüphanesi ile Google Assistant gibi birçok uygulamada kullanılan Tensorflow kütüphanesi kullanılmıştır.

4.7.1. Keras Kütüphanesi

Keras, Python dilinde geliştirilmiş bir derin öğrenme API'sidir ve TensorFlow gibi temel derin öğrenme kütüphaneleriyle uyumlu çalışmaktadır. Kullanımının kolaylığı, esnekliği ve kullanıcı dostu tasarımı sayesinde büyük popülerlik kazanmıştır (Singh vd., 2021). Keras, araştırma tasarımı olarak François Chollet tarafından geliştirilmiştir ve Mart 2015 itibarıyla ulaşılabilir kaynak şeklinde yayınlanmıştır (Géron, 2019).

Keras, her türlü sinir ağı modelinin oluşturulması, eğitimi ve değerlendirilmesi için geniş bir işlevselliğe sahiptir. Özellikle TensorFlow kütüphanesini desteklemesi sayesinde metin ve görüntü işleme gibi alanlarda çalışmayı daha da kolay bir hale getirmiştir. Keras'ın resmi websitesinde, TensorFlow destekli örnek kodlar ve eğitim için kullanılabilecek kaynaklar bulunmaktadır (Keras.io, 2022). Bu özellikleriyle Keras, derin öğrenme projelerinde

kullanıcıların yüksek düzeyde performans, esneklik ve geniş bir ekosistem sunarak kolaylıkla model oluşturma, eğitim, tahminleme ve modelin dağıtımı gibi bir dizi işlemi gerçekleştirmelerine olanak sağlamaktadır.

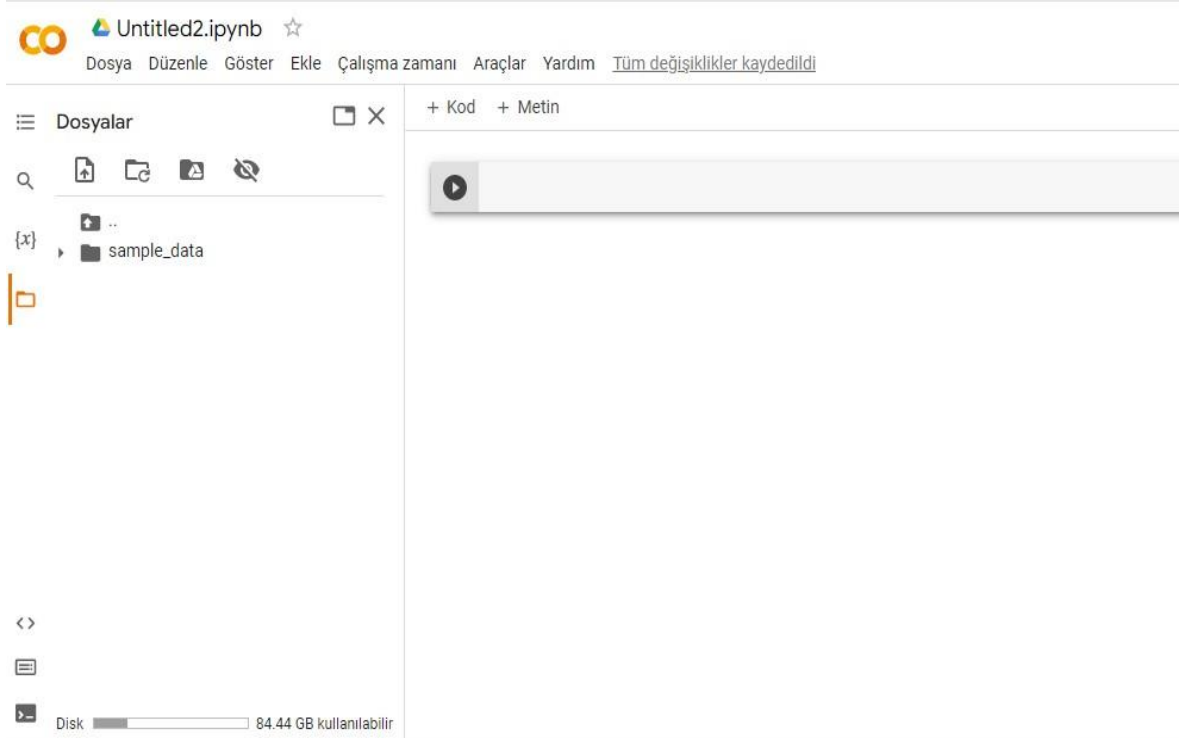
4.7.2. Tensorflow Kütüphanesi

Google Brain ekibi tarafından geliştirilen ve büyük ölçekli makine öğrenimine yatkın olan TensorFlow, sayısal bilgi işlem kütüphanesinde güçlü bir yapıya sahiptir. Bu ücretsiz ulaşılabilir bir kaynaktır. Google Cloud, kütüphane, Google Arama, Google Fotoğraflar gibi birçok Google imkanlarını desteklemektedir. Öneri sistemleri, doğal dil işleme, görüntü sınıflandırma gibi birçok alanda faydalanılan Tensorflow, son dönemin popüler ve yaygın kullanılan kitaplıklarından biridir. Yapay zeka ve derin öğrenme projelerinde kullanıcılara yüksek performans, esneklik ve geniş bir ekosistem sunmaktadır (Cerit, 2020; Singh vd., 2021; Géron, 2019). TensorFlow, kullanıcıların model oluşturma, eğitim, tahminleme ve modelin dağıtımı gibi bir dizi işlemi kolaylıkla gerçekleştirmelerini sağlayarak makine öğrenimi uygulamalarının geliştirilmesine büyük katkı sağlamaktadır.

Bu kütüphane, Python programlama dili üzerinde kullanılabilen ve eğitim süreci bir ortamda gerçekleştirildikten sonra elde edilen modeller, diğer ortamlarda da çalıştırılabilir hale getirilebilir (Géron, 2019). Çalışmanızda kullanmış olduğunuz Tensorflow, çeşitli komutlar ve özellikler sunmaktadır. Ön işleme ve veri yükleme işlemleri için `tf.io` ve `tf.data`, görüntü işleme işlemlerinde `tf.image` ve Keras kütüphanesi kullanımı için ise `tf.keras` gibi komutlar yaygın olarak kullanılmaktadır. Bununla birlikte, farklı özellikler ve ihtiyaçlara yönelik olarak farklı komutlar da mevcuttur. Tensorflow, geniş bir işlevselliğe sahip olup kullanıcıya esneklik sağlayarak çeşitli makine öğrenimi projelerinin geliştirilmesini desteklemektedir.

5. BULGULAR

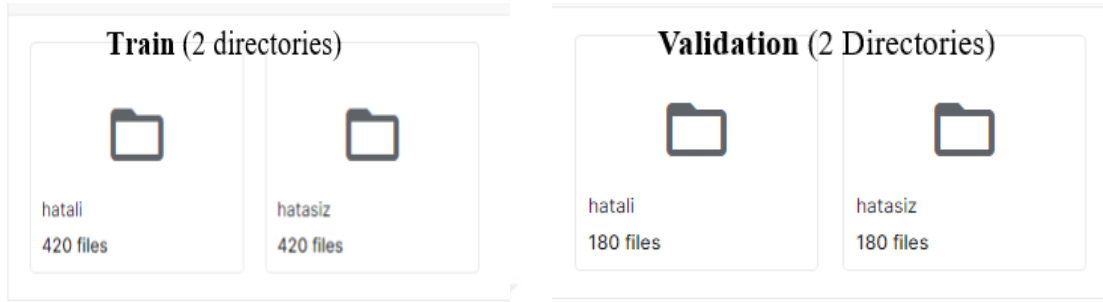
Derin öğrenme model eğitimi ve çıkarımı genellikle yoğun hesaplama gerektirir. GPU'lar, paralel işlem yetenekleri sayesinde bu hesaplamaları hızlandırır. Bu çalışmamızda elimizde güçlü bir makinenin bulunmaması, sadece internet tarayıcısı kullanılarak erişebilmesi sebebiyle kullanıcıların derin öğrenme projeleri üzerinde hızlı bir şekilde çalışmasını sağlaması, makine öğrenimi projeleri için GPU desteği sağlaması ve Python dilini kullanarak birçok popüler veri işleme ve makine öğrenimi kütüphanesini içermesinden dolayı Google Colab kullanılmıştır. Çalışmamızda, hatalı ürünlerin tespiti için pedilen ayak görüntülerinin analizi gerçekleştirildi ve bu analizler, Python programlaştırma dilinden yararlanılarak Google Colab üzerindeki Jupyter not defterleri kullanılarak gerçekleştirilmiştir (Şekil 16).



Şekil 16: Jupyter Notebook Görselinin Google Colab Ortamındaki Görünümü

Çalışmada yararlanılan model çeşitlerinin eğitimi için veri setinin görüntüleri pedilen ayak üretimi yapan firmadan temin edilmiştir. Firmadan temin edilen pedilen ayak görüntülerine model eğitiminden önce Python kodları aracılığıyla döndürme, parlaklık derecesini artırma, filtreleme ve yeniden boyutlandırma gibi görüntü çoğaltma işlemleri uygulanması sonucu 1200 görüntüden oluşan veri seti elde edilmiştir. Birçok veri setinin yer aldığı Kaggle sitesine 1200 resimden oluşan veri setimiz “Dataset10 Hatalı Hatasız 1200 Resim” ismiyle yüklenmesi sonucu train ve test dosyaları oluşturularak hazır hale getirilmiş ve kullanılmıştır.

Veri setimizde hatalı ve hatasız olarak işleme alınan ve iki sınıf yapıya sahip bu çalışmamızda, eğitim aşamasında 420 hatalı, 420 hatasız olmak üzere toplamda 840 görüntü ile test aşamasında ise 180 hatalı, 180 hatasız olmak üzere toplamda 360 görüntü yer almaktadır. (Şekil 17). Veri setimizdeki görüntülerin %30'u test aşamasında, %70'i ise eğitim aşamasında kullanılmıştır.



Şekil 17: Veri Kümesi Dağılımının Görseli

Xception, InceptionResNetV2, DenseNet201, NasNetMobile ve EfficientNetV2S gibi derin öğrenme modellerimizin nesne tespiti için doğru sonuçlar elde etmek için kullandığımız parametreler Tablo 3'te aşağıda belirtilmiştir.

Tablo 3: Modelleri Eğitiminde Kullanılan Parametreler

Performans Ölçütü	Değerler
Öğrenme Oranı	0.00001
Epochs (Eğitim Sayısı)	10 / 20
Batch Size Değeri	32

Çalışmamızda Tablo 3'te belirtilen değerler kullanılarak yapılan analizler sonucunda, başlangıç öğrenme oranı 0.00001, epoch sayısı 20, batch size değeri 32 olarak belirlendiğinde önemli ölçüde başarılı sonuçlar elde edildiği gözlemlenmiştir.

5.1. Xception Modeli Analiz Sonuçları

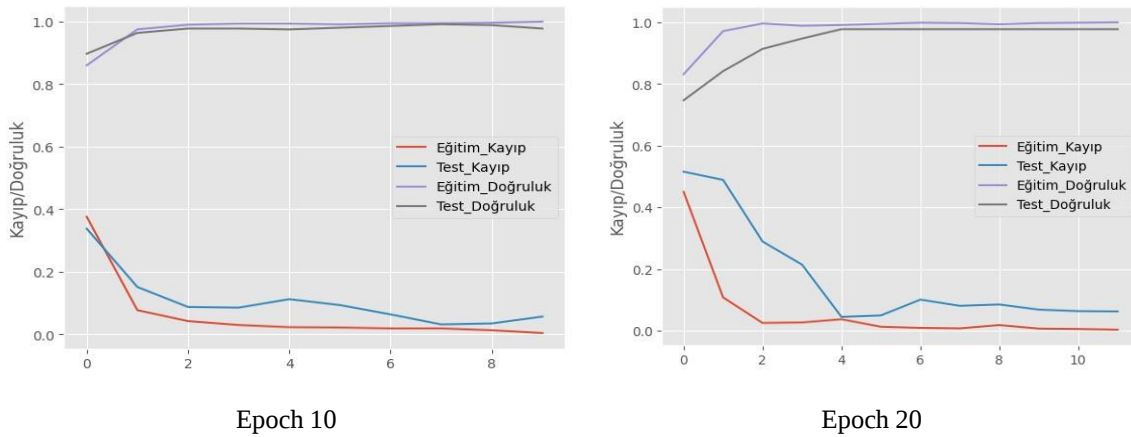
Çalışmamızın bu aşamasında CNN sinir ağı türünün çeşitli derin öğrenme modellerinden olan Xception modeli ile veri setimizdeki değişkenler arasındaki ilişkileri anlamak ve önemli bulguları ortaya çıkarmak için analiz gerçekleştirilmiştir.

Tablo 4: Xception Modeli Ölçüm Parametre Sonuçları

Epoch	(Accuracy) Doğruluk Değeri	(Loss) Kayıp Değeri	Eğitim İçin Geçen Süre
10	0.9889	0.0272	2 saat 7 dk 45.03 sn
20 (epoch12'de eğitim bitmiştir)	0.9778	0.0450	3 saat 2 dk 41.17 sn

Tablo 4’te görüldüğü gibi, Xception modeli kullanılarak gerçekleştirilen eğitimlerde; epoch parametresi 10 olarak seçildiğinde eğitimin toplamda 2 saat 7 dakika 45.03 saniyede tamamlandığı, bu süre içerisinde, kayıp oranının 0.0272 olarak hesaplandığı ve %98.89 oranında doğruluk elde edildiği, epoch parametresi 20 olarak seçildiğinde ise eğitimin toplamda 3 saat 2 dakika 41.17 saniyede epoch 12 de erken sonlanarak tamamlandığı, bu süre içerisinde, kayıp oranının 0.0450 olarak hesaplandığı ve %97.78 oranında doğruluk elde edildiği anlaşılmıştır. Dolayısıyla modelimizin hatalı hatasız tespitinde epoch parametresi 10 seçildiğinde yaklaşık iki saatlik sürede ve epoch parametresi 20 seçildiğinde ise epoch 12’de erken durması sonucu yaklaşık üç saatlik sürede tamamlanan eğitim sonuçlarının başarılı olarak ifade edilebilecek seviyede olduğu görülmüştür. Analiz sonucu ile ilgili eğitim ve test verilerine göre kayıp ve doğruluk değerlerinin değişiminin görüldüğü Şekil 18’den de anlaşılacağı üzere, epoch parametresinin 10 ve 20 seçildiği her iki durumda da eğitim verilerine ait sonuçların test verilerine ait sonuçlara göre kaybının daha düşük seviyede olduğu, doğruluk değerinin ise daha yüksek seviyede olduğu anlaşılmaktadır. Değerlerin eğitim sürecinde artış ve azalış durumlarında dalgalanmalar olsa da epoch parametresinin 10 ve 20 seçildiği her iki durumda da düzenli bir şekilde gerçekleştiği görülmektedir.

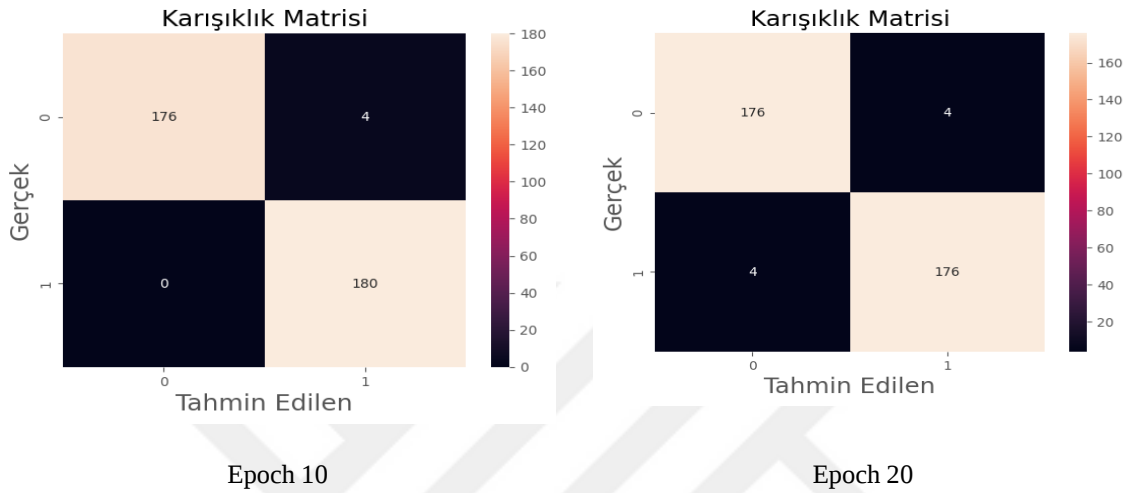
Xception



Şekil 18: Xception Modeli Eğitim Grafikleri

Şekil 19’da Xception modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek gerçekleştirilen eğitimler sonunda oluşan 180 hatalı, 180 hatasız test verilerine ait karışıklık matrisleri yer almaktadır. Xception modelinin epoch parametresi farklı seçilerek gerçekleştirilen eğitimlerine ait Karışıklık Matrislerine göre; epoch parametresi 10 seçildiğinde 180 hatasız görüntüsünün 180’ini hatasız olarak sınıflandırdığı, 180 hatalı görüntüsünün 176’sını hatalı ve 4’ünü hatasız olarak sınıflandırdığı ve epoch parametresi 20 seçildiğinde ise 180 hatasız görüntüsünün 176’sını hatasız ve 4’ünü hatalı olarak

sınıflandırdığı, 180 hatalı görüntüsünün 176'sını hatalı ve 4'ünü hatasız olarak sınıflandırdığı anlaşılmaktadır. Modelin; epoch parametresi 10 seçildiğinde hatasız görüntü verilerini tespit etmede başarılı olduğu, hatalı görüntü verilerini tespit etmede az da olsa yanlış olduğu, epoch parametresi 20 seçildiğinde ise hatalı ve hatasız görüntü verilerini tespit etmede az da olsa aynı oranda yanlış tahminde bulunduğu görülmektedir.



Şekil 19: Xception Modeli Karışıklık Matrisleri

Xception modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek gerçekleştirilen eğitimlerin sonuçlarıyla ilgili Tablo 5’de test ve eğitim verilerinin hatalı ve hatasız olacak şekilde sınıflandırılması analizinin f1-skor, geri çağırma ve kesinlik değerleri yer almaktadır.

Tablo 5: Xception Modeli Sınıflama Raporları

		<i>Kesinlik</i>	<i>Geri Çağırma</i>	<i>F1-score</i>	<i>Görüntü Sayısı</i>
Epoch 10	<i>0</i>	1.00	0.98	0.99	180
	<i>1</i>	0.98	1.00	0.99	180
	<i>Makro Ort.</i>	0.99	0.99	0.99	360
	<i>Ağırlıklı Ort.</i>	0.99	0.99	0.99	360
Epoch 20	<i>0</i>	0.98	0.98	0.98	180
	<i>1</i>	0.98	0.98	0.98	180
	<i>Makro Ort.</i>	0.98	0.98	0.98	360
	<i>Ağırlıklı Ort.</i>	0.98	0.98	0.98	360

Tablo 5 incelendiğinde eğitim veri kümesi için, epoch parametresi 10 seçildiğinde kesinlik ve geri çağırma değerlerinde %97’lik, f1-skor değerinde %99’luk bir oran, epoch parametresi 20 seçildiğinde ise f1-skor, geri çağırma ve kesinlik değerlerinin hepsinin oranı %98’dir. Test veri kümesi için epoch parametresi 10 seçildiğinde hatalı (0) görüntülere ait f1-skor değerinin %99, geri çağırma değerinin %98 ve kesinlik değerinin ise %100 olduğu, hatasız (1) veriler için ise kesinlik değeri %98, geri çağırma değeri %100 ve f1-skor değerinin ise %99 olduğu

görülmektedir. Yine test verileri için bu sonuçlar epoch parametresi 20 seçildiğinde ise hatalı (0) görüntülere ait kesinlik değeri %98, geri çağırma değeri %98 ve f1-skor değerinin ise %98 olduğu, hatasız (1) görüntülere ait ise f1-skor değerinin %98, geri çağırma değerinin %98 ve kesinlik değerinin ise %98 olduğu anlaşılmaktadır.

5.2. InceptionResNetV2 Modeli Analiz Sonuçları

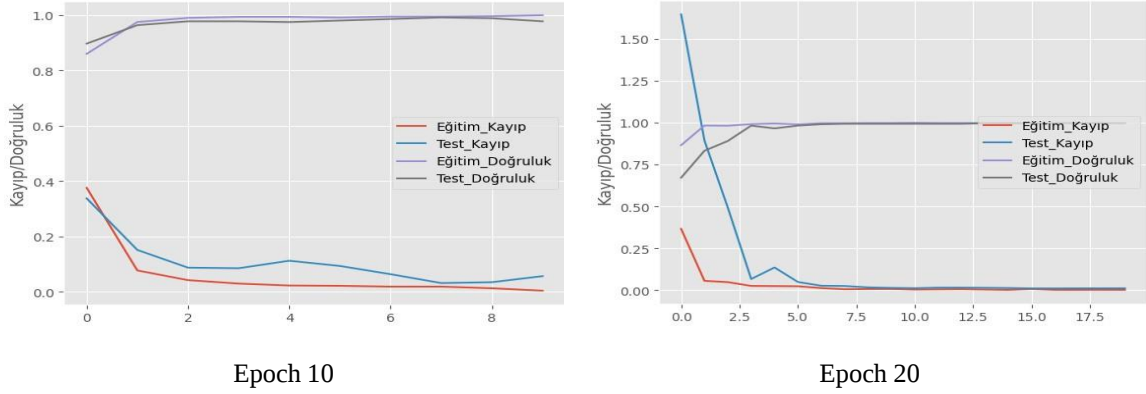
Çalışmamızın bu aşamasında CNN sinir ağı türünün çeşitli derin öğrenme modellerinden olan InceptionResNetV2 modeli ile veri setimizdeki değişkenler arasındaki ilişkileri anlamak ve önemli bulguları ortaya çıkarmak için analiz gerçekleştirilmiştir.

Tablo 6: InceptionResNetV2 Modeli Ölçüm Parametre Sonuçları

Epoch	(Accuracy) Doğruluk Değeri	(Loss) Kayıp Değeri	Eğitim İçin Geçen Süre
10	0.9778	0.0566	2 saat 49 dk 19.54 sn
20	0.9972	0.0120	5 saat 11 dk 39.93 sn

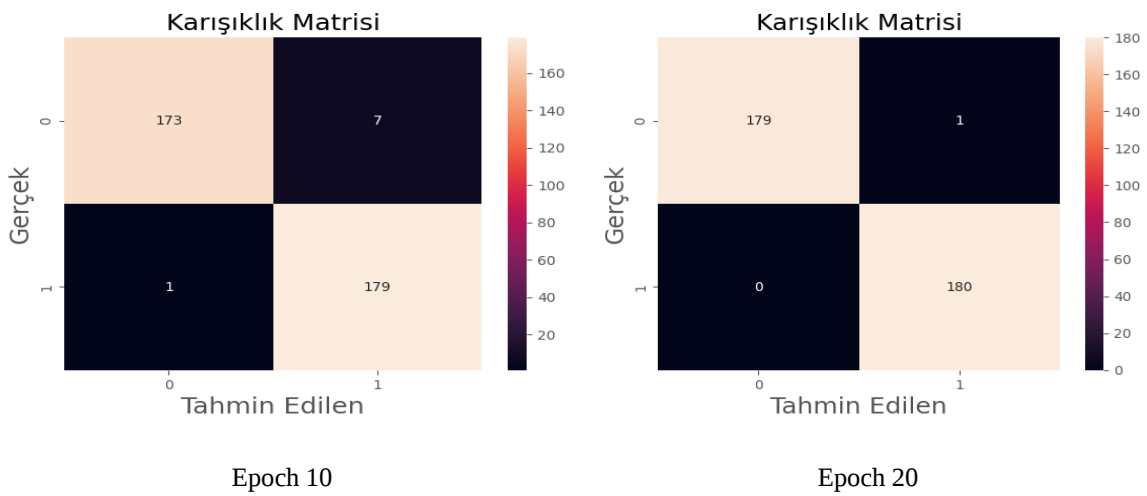
Tablo 6’da görüldüğü gibi, InceptionResNetV2 modeli kullanılarak gerçekleştirilen eğitimlerde; epoch parametresi 10 olarak seçildiğinde eğitimin toplamda 2 saat 49 dakika 19.54 saniyede tamamlandığı, bu süre içerisinde, kayıp oranının 0.0566 olarak hesaplandığı ve %97.78 oranında doğruluk elde edildiği, epoch parametresi 20 olarak seçildiğinde ise eğitimin toplamda 5 saat 11 dakika 39.93 saniyede tamamlandığı, bu süre içerisinde, kayıp oranının 0.0120 olarak hesaplandığı ve %99.72 oranında doğruluk elde edildiği anlaşılmıştır. Dolayısıyla modelimizin hatalı hatasız tespitinde epoch parametresi 10 seçildiğinde yaklaşık üç saatlik sürede ve epoch parametresi 20 seçildiğinde ise yaklaşık beş saatlik sürede tamamlanan eğitim sonuçlarının başarılı olarak ifade edilebilecek seviyede olduğu görülmüştür. Analiz sonucu ile ilgili eğitim ve test verilerine göre kayıp ve doğruluk değerlerinin değişiminin görüldüğü Şekil 20’den de anlaşılabileceği üzere, epoch parametresinin 10 seçildiği durumda eğitim verilerine ait sonuçların test verilerine ait sonuçlara göre kaybının daha düşük seviyede olduğu, doğruluk değerinin ise daha yüksek seviyede olduğu, epoch parametresinin 20 seçildiği durumda ise eğitim verilerinde test verilerindeki kaybın eğitim verilerine göre başlarda daha yüksek olmasına karşın eğitim sonunda eğitim ve test verilerindeki kaybın eşitlendiği, doğruluk değerinin de aynı şekilde eşitlendiği anlaşılmaktadır. Değerlerin eğitim sürecinde artış ve azalış durumlarında dalgalanmalar olsa da epoch parametresinin 10 ve 20 seçildiği her iki durumda da düzenli bir şekilde gerçekleştiği görülmektedir.

InceptionResNetV2



Şekil 20: InceptionResNetV2 Modeli Eğitim Grafikleri

Şekil 21’de InceptionResNetV2 modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek gerçekleştirilen eğitimler sonunda oluşan 180 hatalı, 180 hatasız test verilerine ait karışıklık matrisleri yer almaktadır. InceptionResNetV2 modelinin epoch parametresi farklı seçilerek gerçekleştirilen eğitimlerine ait Karışıklık Matrislerine göre; epoch parametresi 10 seçildiğinde 180 hatasız görüntüsünün 179’unu hatasız ve 1’ini hatalı olarak sınıflandırdığı, 180 hatalı görüntüsünün 173’ünü hatalı ve 7’sini hatasız olarak sınıflandırdığı ve epoch parametresi 20 seçildiğinde ise 180 hatasız görüntüsünün 180’ini de hatasız olarak sınıflandırdığı, 180 hatalı görüntüsünün 179’unu hatalı ve 1’ini hatasız olarak sınıflandırdığı anlaşılmaktadır. Modelin; epoch parametresi 10 seçildiğinde hatasız görüntü verilerini tespit etmede başarılı olduğu, hatalı görüntü verilerini tespit etmede az da olsa yanlış olduğu, epoch parametresi 20 seçildiğinde ise hem hatalı ve hem de hatasız görüntüleri ayırt etmede kayda değer oranda başarılı olduğu görülmektedir.



Şekil 21: InceptionResNetV2 Modeli Karışıklık Matrisleri

InceptionResNetV2 modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek eğitimlerin sonuçlarıyla ilgili Tablo 7’de test ve eğitim verilerinin hatalı ve hatasız olacak şekilde sınıflandırılması analizinin f1-skor, geri çağırma ve kesinlik değerleri yer almaktadır.

Tablo 7: InceptionResNetV2 Modeli Sınıflama Raporları

		<i>Kesinlik</i>	<i>Geri Çağırma</i>	<i>F1-score</i>	<i>Görüntü Sayısı</i>
Epoch 10	<i>0</i>	0.99	0.96	0.98	180
	<i>1</i>	0.96	0.99	0.98	180
	<i>Makro Ort.</i>	0.98	0.98	0.98	360
	<i>Ağırlıklı Ort.</i>	0.98	0.98	0.98	360
		<i>Kesinlik</i>	<i>Geri Çağırma</i>	<i>F1-score</i>	<i>Görüntü Sayısı</i>
Epoch 20	<i>0</i>	1.00	0.99	1.00	180
	<i>1</i>	0.99	1.00	1.00	180
	<i>Makro Ort.</i>	1.00	1.00	1.00	360
	<i>Ağırlıklı Ort.</i>	1.00	1.00	1.00	360

Tablo 7 incelendiğinde eğitim veri kümesi için, epoch parametresi 10 seçildiğinde kesinlik ve geri çağırma değerlerinde %96’lık, f1-skor değerinde %98’lik bir oran, epoch parametresi 20 seçildiğinde ise kesinlik ve geri çağırma değerlerinin oranı %99, f1-skor değerinin oranı ise %100’dür. Test veri kümesi için epoch parametresi 10 seçildiğinde hatalı (0) görüntülere ait f1-skor değerinin ise %98, geri çağırma değeri %96 ve kesinlik değeri %99 olduğu, hatasız (1) veriler için ise f1-skor değerinin ise %98, geri çağırma değerinin %99 ve kesinlik değerinin %96 olduğu görülmektedir. Yine test verileri için bu sonuçlar epoch parametresi 20 seçildiğinde ise hatalı (0) görüntülere ait f1-skor değerinin ise %100, geri çağırma değerinin %99 ve kesinlik değerinin %100 olduğu, hatasız (1) veriler için ise f1-skor değerinin ise %100, geri çağırma değerinin %100 ve kesinlik değerinin %99 olduğu görülmektedir.

5.3. DenseNet201 Modeli Analiz Sonuçları

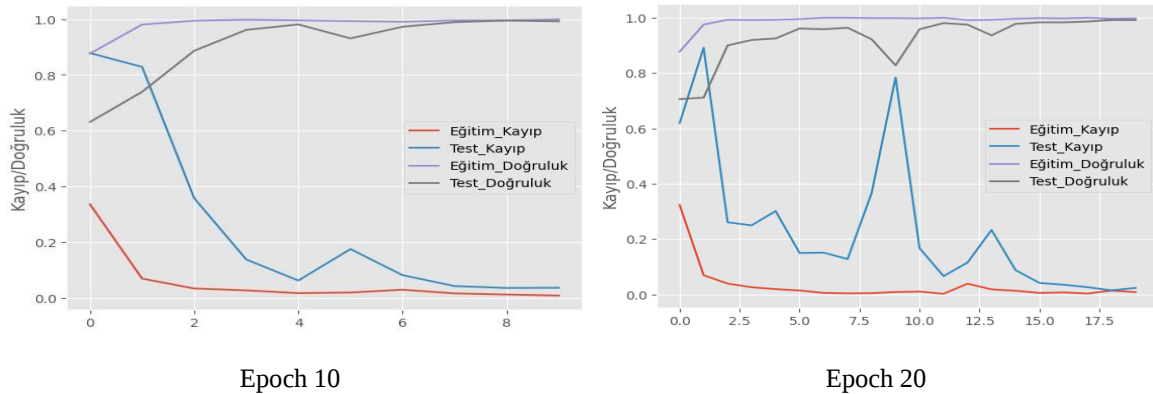
Çalışmamızın bu aşamasında CNN sinir ağı türünün çeşitli derin öğrenme modellerinden olan DenseNet201 modeli ile veri setimizdeki değişkenler arasındaki ilişkileri anlamak ve önemli bulguları ortaya çıkarmak için analiz gerçekleştirilmiştir.

Tablo 8: DenseNet201 Modeli Ölçüm Parametre Sonuçları

Epoch	(Accuracy) Doğruluk Değeri	(Loss) Kayıp Değeri	Eğitim İçin Geçen Süre
10	0.9917	0.0357	1 saat 45 dk 29.32 sn
20	0.9917	0.232	4 saat 12 dk 9.17 sn

Tablo 8’de görüldüğü gibi, DenseNet201 modeli kullanılarak yapılan eğitimlerde; epoch parametresi 10 olarak seçildiğinde eğitimin toplamda 1 saat 45 dakika 29.32 saniyede tamamlandığı, bu süre içerisinde, kayıp oranının 0.0357 olarak hesaplandığı ve %99.17 oranında doğruluk elde edildiği, epoch parametresi 20 olarak seçildiğinde ise eğitimin toplamda 4 saat 12 dakika 9.17 saniyede tamamlandığı, bu süre içerisinde, kayıp oranının 0.232 olarak hesaplandığı ve %99.17 oranında doğruluk elde edildiği anlaşılmıştır. Dolayısıyla modelimizin hatalı hatasız tespitinde epoch parametresi 10 seçildiğinde yaklaşık bir saat 45 dakikalık sürede ve epoch parametresi 20 seçildiğinde ise yaklaşık dört saatlik sürede eğitim sonuçlarının başarılı olarak ifade edilebilecek seviyede olduğu görülmüştür. Analiz sonucu ile ilgili eğitim ve test verilerine göre kayıp ve doğruluk değerlerinin değişiminin görüldüğü Şekil 22’den de anlaşılabacağı üzere, epoch parametresinin 10 ve 20 seçildiği her iki durumda da eğitim verilerine ait sonuçların test verilerine ait sonuçlara göre kaybının daha düşük seviyede olduğu, doğruluk değerinin ise daha yüksek seviyede olduğu anlaşılmaktadır. Değerlerin eğitim sürecinde artış ve azalış durumlarında dalgalanmalar olsa da epoch parametresinin 10 ve 20 seçildiği her iki durumda da düzenli bir şekilde gerçekleştiği görülmektedir.

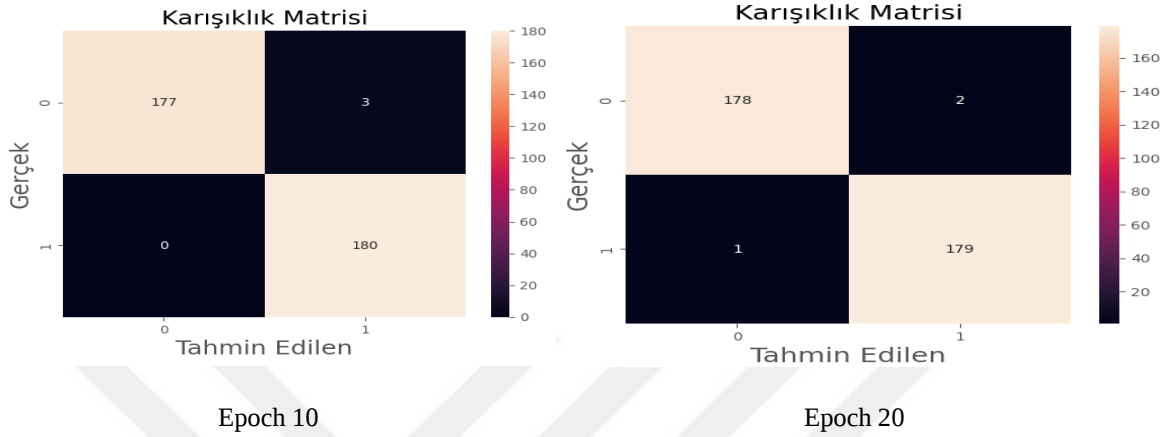
DenseNet201



Şekil 22: DenseNet201 Modeli Eğitim Grafikleri

Şekil 23’te DenseNet201 modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek gerçekleştirilen eğitimler sonunda oluşan 180 hatalı, 180 hatasız test verilerine ait karışıklık matrisleri yer almaktadır. DenseNet201 modelinin epoch parametresi farklı seçilerek gerçekleştirilen eğitimlerine ait Karışıklık Matrislerine göre; epoch parametresi 10 seçildiğinde 180 hatasız görüntüsünün 180’ini de hatasız olarak sınıflandırdığı, 180 hatalı görüntüsünün 177’sini hatalı ve 3’ünü hatasız olarak sınıflandırdığı ve epoch parametresi 20 seçildiğinde ise 180 hatasız görüntüsünün 179’unun hatalı ve 1’ini hatasız olarak

sınıflandırdığı, 180 hatalı görüntüsünün 178'ini hatalı ve 2'sini hatasız olarak sınıflandırdığı anlaşılmaktadır. Modelin; epoch parametresi 10 veya 20 seçildiğinde de hatasız görüntü verilerini tespit etmede başarılı olduğu, hatalı görüntü verilerini tespit etmede az da olsa yanlışlığı görülmektedir.



Şekil 23: DenseNet201Modeli Karışıklık Matrisleri

DenseNet201 modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek gerçekleştirilen eğitimlerin sonuçlarıyla ilgili Tablo 9'da test ve eğitim verilerinin hatalı ve hatasız olacak şekilde sınıflandırılması analizinin f1-skor, geri çağırma ve kesinlik değerleri yer almaktadır.

Tablo 9: DenseNet201 Modeli Sınıflama Raporları

		<i>Kesinlik</i>	<i>Geri Çağırma</i>	<i>F1-score</i>	<i>Görüntü Sayısı</i>
Epoch 10	0	1.00	0.98	0.99	180
	1	0.98	1.00	0.99	180
	Makro Ort.	0.99	0.99	0.99	360
	Ağırlıklı Ort.	0.99	0.99	0.99	360
Epoch 20	0	0.99	0.99	0.99	180
	1	0.99	0.99	0.99	180
	Makro Ort.	0.99	0.99	0.99	360
	Ağırlıklı Ort.	0.99	0.99	0.99	360

Tablo 9 incelendiğinde eğitim veri kümesi için, epoch parametresi 10 seçildiğinde kesinlik ve geri çağırma değerlerinde %98'lik, f1-skor değerinde %99'luk bir oran, epoch parametresi 20 seçildiğinde ise kesinlik, geri çağırma ve f1-skor değerlerinin hepsinin oranının %99 olduğu anlaşılmaktadır. Test veri kümesi için epoch parametresi 10 seçildiğinde hatalı (0) görüntülere ait f1-skor değerinin %99, geri çağırma değerinin %98 ve kesinlik değerinin %100 olduğu, hatasız (1) veriler için ise f1-skor değerinin %99, geri çağırma değeri %100 ve kesinlik değeri %98 olduğu görülmektedir. Yine test veri kümesi için bu sonuçlar epoch parametresi 20 seçildiğinde ise hatalı (0) görüntülere ait f1-skor değerinin %99, geri çağırma

değeri %99 ve kesinlik değeri %99 olduğu, hatasız (1) veriler için ise f1-skor değerinin %99, geri çağırma değeri %99 ve kesinlik değeri %99 olduğu görülmektedir.

5.4. NasNetMobile Modeli Analiz Sonuçları

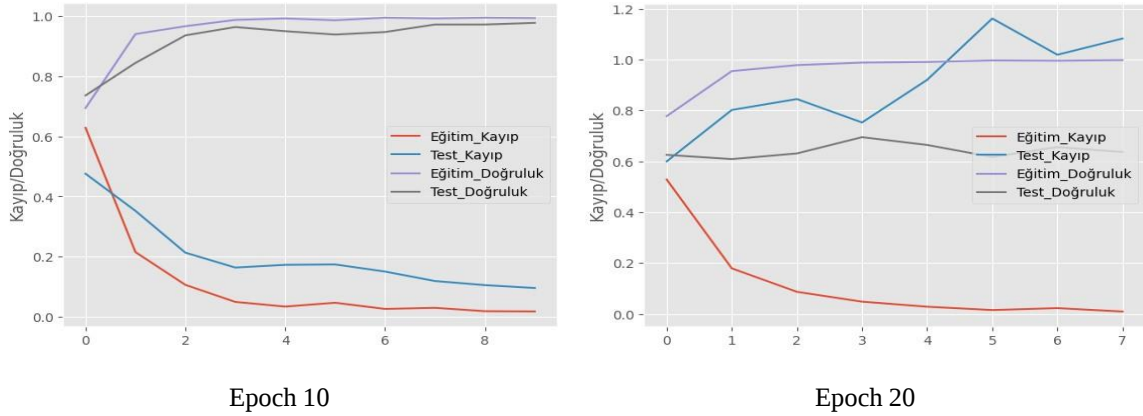
Çalışmamızın bu aşamasında CNN sinir ağı türünün çeşitli derin öğrenme modellerinden olan NasNetMobile modeli ile veri setimizdeki değişkenler arasındaki ilişkileri anlamak ve önemli bulguları ortaya çıkarmak için analiz gerçekleştirilmiştir.

Tablo 10: NasNetMobile Modeli Ölçüm Parametre Sonuçları

Epoch	(Accuracy) Doğruluk Değeri	(Loss) Kayıp Değeri	Eğitim İçin Geçen Süre
10	0.9778	0.0952	1 saat 1 dk 2.51 sn
20 (epoch8'de eğitim bitmiştir)	0.6361	1.0823	47 dk 51.54 sn

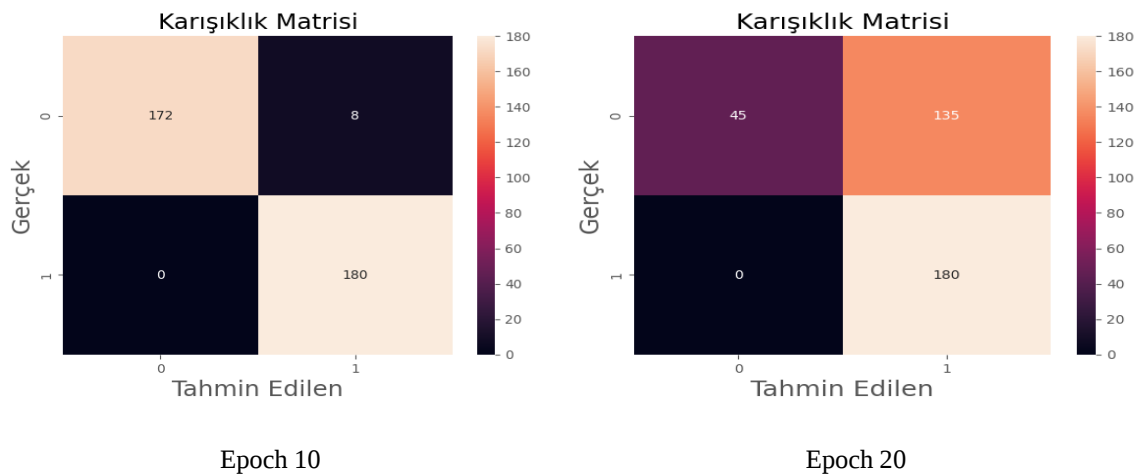
Tablo 10'da görüldüğü gibi, NasNetMobile modeli kullanılarak gerçekleştirilen eğitimlerde; epoch parametresi 10 olarak seçildiğinde eğitimin toplamda 1 saat 1 dakika 2.51 saniyede tamamlandığı, bu süre içerisinde, kayıp oranının 0.0952 olarak hesaplandığı ve %97.78 oranında doğruluk elde edildiği, epoch parametresi 20 olarak seçildiğinde ise eğitimin toplamda 47 dakika 51.54 saniyede epoch 8'de erken sonlanarak tamamlandığı, bu süre içerisinde, kayıp oranının 1.0823 olarak hesaplandığı ve %63.61 oranında kayda değer olmayacak kadar düşük doğruluk elde edildiği anlaşılmıştır. Dolayısıyla modelimizin hatalı hatasız tespitinde epoch parametresi 10 seçildiğinde yaklaşık bir saatlik sürede tamamlanan eğitimin sonucunun başarı olarak ifade edilebileceği ve epoch parametresi 20 seçildiğinde ise epoch 8'de erken durması sonucu yaklaşık bir saatlik sürede tamamlanan eğitimin sonucunun başarısız olarak ifade edilebilecek seviyede olduğu anlaşılmıştır. Analiz sonucu ile ilgili eğitim ve test verilerine göre kayıp ve doğruluk değerlerinin değişiminin görüldüğü Şekil 24'ten de anlaşılabacağı üzere, epoch parametresinin 10 seçildiği durumda eğitim verilerine ait sonuçların test verilerine ait sonuçlara göre kaybın daha düşük olduğu, doğruluk değerinin ise daha yüksek olduğu ve epoch parametresinin 20 seçildiği durumda ise eğitim verilerinde test verilerine göre kayıp ve doğruluk değerlerinin ise istenilen kayda değer sonuçlar olmadığı anlaşılmaktadır. Değerlerin eğitim sürecinde artış ve azalış durumlarında dalgalanmalar olsa da epoch parametresinin 10 seçildiği durumda düzenli bir şekilde gerçekleştiği, ancak epoch parametresi 20 seçildiği durumda aynı şekilde ifade edilemeyeceği görülmektedir.

NasNetMobile



Şekil 24: NasNetMobile Modeli Eğitim Grafikleri

Şekil 25'te NasNetMobile modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek gerçekleştirilen eğitimler sonunda oluşan 180 hatalı, 180 hatasız test veri kümesine ait karışıklık matrisleri görülmektedir. NasNetMobile modelinin epoch parametresi farklı seçilerek gerçekleştirilen eğitimlerine ait Karışıklık Matrislerine göre; epoch parametresi 10 seçildiğinde 180 hatasız görüntüsünün 180'ini hatasız olarak sınıflandırdığı, 180 hatalı görüntüsünün 172'sini hatalı ve 8'ini hatasız olarak sınıflandırdığı ve epoch parametresi 20 seçildiğinde ise 180 hatasız görüntüsünün 180'ini hatasız olarak sınıflandırdığı, 180 hatalı görüntüsünün 135'ini hatalı ve 45'ini hatasız olarak sınıflandırdığı anlaşılmaktadır. Modelin; epoch parametresi 10 seçildiğinde hatasız görüntü verilerini tespit etmede başarılı olduğu, hatalı görüntü verilerini tespit etmede az da olsa yanlış olduğu, epoch parametresi 20 seçildiğinde ise yine hatasız görüntüleri ayırt etmede başarılı olduğu, ancak hatalı görüntüleri ayırt etmede başarılı olamadığı görülmektedir.



Şekil 25: NasNetMobile Modeli Karışıklık Matrisleri

NasNetMobile modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek eğitimlerin sonuçlarıyla ilgili Tablo 11’de test ve eğitim verilerinin hatalı ve hatasız olacak şekilde sınıflandırılması analizinin f1-skor, geri çağırma ve kesinlik değerleri yer almaktadır.

Tablo 11: NasNetMobile Modeli Sınıflama Raporları

		<i>Kesinlik</i>	<i>Geri Çağırma</i>	<i>F1-score</i>	<i>Görüntü Sayısı</i>
Epoch 10	<i>0</i>	1.00	0.96	0.98	180
	<i>1</i>	0.96	1.00	0.98	180
	<i>Makro Ort.</i>	0.98	0.98	0.98	360
	<i>Ağırlıklı Ort.</i>	0.98	0.98	0.98	360
		<i>Kesinlik</i>	<i>Geri Çağırma</i>	<i>F1-score</i>	<i>Görüntü Sayısı</i>
Epoch 20	<i>0</i>	1.00	0.25	0.40	180
	<i>1</i>	0.57	1.00	0.73	180
	<i>Makro Ort.</i>	0.79	0.62	0.56	360
	<i>Ağırlıklı Ort.</i>	0.79	0.62	0.56	360

Tablo 11’de görüldüğü gibi eğitim verileri için, epoch parametresi 10 seçildiğinde kesinlik ve geri çağırma değerlerinde %96’lık, f1-skor değerinde %98’lik bir oran, epoch parametresi 20 seçildiğinde ise f1-skor, geri çağırma ve kesinlik değerlerinin hepsinde çok düşük bir oran elde edilmiştir. Test verileri için bu sonuçlar epoch parametresi 10 seçildiğinde hatalı (0) görüntülere ait f1-skor değerinin %98, geri çağırma değerinin %96 ve kesinlik değerinin %100 olduğu, hatasız (1) veriler için ise f1-skor değerinin %98, geri çağırma değerinin %100 ve kesinlik değerinin %96 olduğu görülmektedir. Yine test verileri için bu sonuçlar epoch parametresi 20 seçildiğinde ise hatalı (0) görüntülere ait kesinlik değeri %100, geri çağırma değeri %25 ve f1-skor değerinin ise %40 olduğu, hatasız (1) veriler için ise kesinlik değeri %57, geri çağırma değeri %100 ve f1-skor değerinin ise %73 olduğu görülmektedir.

5.5. EfficientNetV2S Modeli Analiz Sonuçları

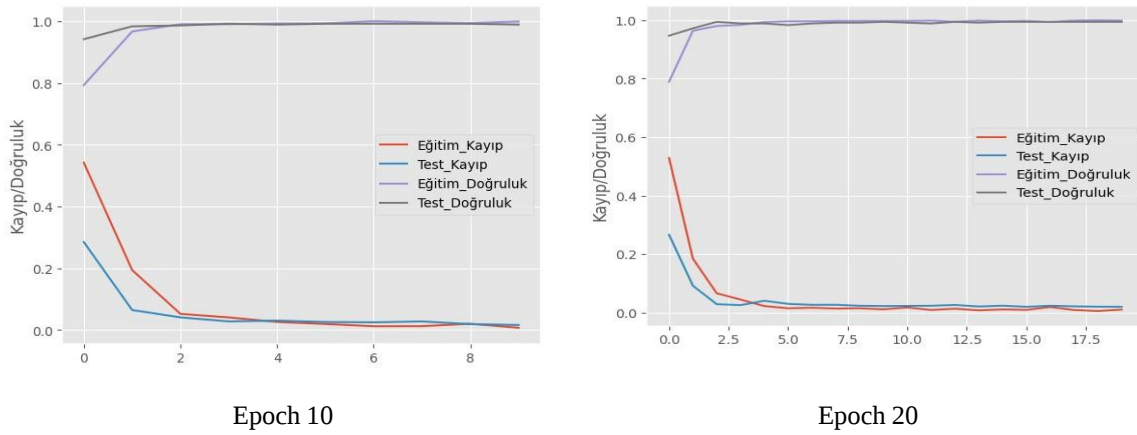
Çalışmamızın bu aşamasında CNN sinir ağı türünün çeşitli derin öğrenme modellerinden olan EfficientNetV2S modeli ile veri setimizdeki değişkenler arasındaki ilişkileri anlamak ve önemli bulguları ortaya çıkarmak için analiz gerçekleştirilmiştir.

Tablo 12: EfficientNetV2S Modeli Ölçüm Parametre Sonuçları

Epoch	(Accuracy) Doğruluk Değeri	(Loss) Kayıp Değeri	Eğitim İçin Geçen Süre
10	0.9889	0.0164	1 saat 6 dk 31.79 sn
20	0.9944	0.0193	3 saat 29 dk 17.49 sn

Tablo 12’de görüldüğü gibi, EfficientNetV2S modeli kullanılarak gerçekleştirilen eğitimlerde; epoch parametresi 10 olarak seçildiğinde eğitimin toplamda 1 saat 6 dakika 31.79 saniyede tamamlandığı, bu süre içerisinde, kayıp oranının 0.0164 olarak hesaplandığı ve %98.89 oranında doğruluk elde edildiği, epoch parametresi 20 olarak seçildiğinde ise eğitimin toplamda 3 saat 29 dakika 17.49 saniyede epoch 12 de tamamlandığı, bu süre içerisinde, kayıp oranının 0.0193 olarak hesaplandığı ve %99.44 oranında doğruluk elde edildiği anlaşılmıştır. Dolayısıyla modelimizin hatalı hatasız tespitinde epoch parametresi 10 seçildiğinde yaklaşık bir saatlik sürede ve epoch parametresi 20 seçildiğinde ise yaklaşık üç buçuk saatlik sürede tamamlanan eğitimlerin sonuçlarının başarılı olarak ifade edilebilecek seviyede olduğu görülmüştür. Analiz sonucu ile ilgili eğitim ve test verilerine göre kayıp ve doğruluk değerlerinin değişiminin görüldüğü Şekil 26’dan da anlaşılabacağı üzere, epoch parametresinin 10 ve 20 seçildiği her iki durumda da eğitim verilerine ait sonuçların test verilerine ait sonuçlara göre kaybının daha düşük seviyede olduğu, doğruluk değerinin ise daha yüksek seviyede olduğu anlaşılmaktadır. Değerlerin eğitim sürecinde azalış ve artış durumlarında az dalgalanmalar olsa bile epoch parametresinin 10 ve 20 seçildiği her iki durumda da düzenli bir şekilde gerçekleştiği görülmektedir.

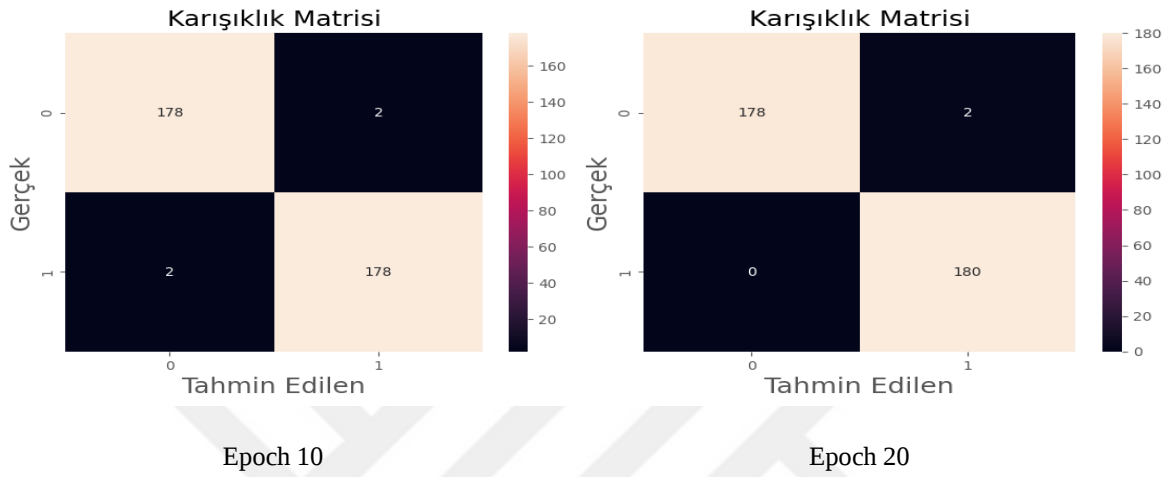
EfficientNetV2S



Şekil 26: EfficientNetV2S Modeli Eğitim Grafikleri

Şekil 27’de EfficientNetV2S modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek gerçekleştirilen eğitimler sonunda oluşan 180 hatalı, 180 hatasız test verilerine ait karışıklık matrisleri yer almaktadır. EfficientNetV2S modelinin epoch parametresi farklı seçilerek gerçekleştirilen eğitimlerine ait Karışıklık Matrislerine göre; epoch parametresi 10 seçildiğinde 180 hatasız görüntüsünün 178’ini hatasız ve 2’sini hatalı olarak sınıflandırdığı, 180 hatalı görüntüsünün 178’ini hatalı ve 2’sini hatasız olarak sınıflandırdığı ve epoch parametresi 20 seçildiğinde ise 180 hatasız görüntüsünün 180’ini hatasız olarak

sınıflandırdığı, 180 hatalı görüntüsünün 178'ini hatalı ve 2'sini hatasız olarak sınıflandırdığı anlaşılmaktadır. Modelin; epoch parametresi 10 seçildiğinde hatalı ve hatasız görüntüleri ayırt etmede az da olsa yanlış olduğu, epoch parametresi 20 seçildiğinde ise hatasız görüntü verilerini tespit etmede başarılı olduğu, hatalı görüntü verilerini ayırt etmede az da olsa yanlış olduğu görülmektedir.



Şekil 27: EfficientNetV2S Modeli Karışıklık Matrisleri

EfficientNetV2S modeli kullanılarak epoch parametresi 10 ya da 20 seçilerek eğitimlerin sonuçlarıyla ilgili Tablo 13'de test ve eğitim verilerinin hatalı ve hatasız olacak şekilde sınıflandırılması analizinin f1-skor, geri çağırma ve kesinlik değerleri yer almaktadır.

Tablo 13: EfficientNetV2S Modeli Sınıflama Raporları

		<i>Kesinlik</i>	<i>Geri Çağırma</i>	<i>F1-score</i>	<i>Görüntü Sayısı</i>
Epoch 10	<i>0</i>	0.99	0.99	0.99	180
	<i>1</i>	0.99	0.99	0.99	180
	<i>Makro Ort.</i>	0.99	0.99	0.99	360
	<i>Ağırlıklı Ort.</i>	0.99	0.99	0.99	360
Epoch 20	<i>0</i>	1.00	0.99	0.99	180
	<i>1</i>	0.99	1.00	0.99	180
	<i>Makro Ort.</i>	0.99	0.99	0.99	360
	<i>Ağırlıklı Ort.</i>	0.99	0.99	0.99	360

Tablo 13'de görüldüğü gibi eğitim verileri için, epoch parametresi 10 seçildiğinde f1-skor, geri çağırma ve kesinlik, değerlerinin hepsinde %99'luk bir oran, epoch parametresi 20 seçildiğinde ise f1-skor, geri çağırma ve kesinlik değerlerinin hepsinde %99'luk bir oran elde edilmiştir. Test verileri için bu sonuçlar epoch parametresi 10 seçildiğinde hatalı (0) görüntülere ait f1-skor, geri çağırma ve kesinlik değerlerinin hepsinin %99 olduğu, hatasız (1) veriler için de f1-skor, geri çağırma ve kesinlik değerlerinin hepsinin %99 olduğu

görülmektedir. Yine test verileri için bu sonuçlar epoch parametresi 20 seçildiğinde ise hatalı (0) görüntülere ait f1-skor değerinin %99, geri çağırma değeri %99 ve kesinlik değeri %100 olduğu, hatasız (1) veriler için ise f1-skor değerinin ise %99, geri çağırma değeri %100 ve kesinlik değeri %99 olduğu görülmektedir.

5.6. Analizde Kullanılan Modellerin Performans Karşılaştırması

Çalışmamızın bu aşamasında görüntü işleme, görüntü sınıflandırma ve nesne tespiti gibi görsel verilerle çalışmak için etkili bir yöntem olarak kabul edilen CNN sinir ağı türünün çeşitli derin öğrenme modellerinden olan Xception, InceptionResNetV2, DenseNet201, NasNetMobile ve EfficientNetV2S modelleri kullanılarak gerçekleştirilen analizlerin sonuçları en doğru hatalı ürün tespiti yapan model ortaya çıkarmak amacıyla karşılaştırılmıştır.

Tablo 14: Modellerin Başarım Ölçütleri ve Eğitim Süresi Değerlerinin Karşılaştırılması

Model Adı	Epoch	Doğruluk	Kesinlik	Geri Çağırma	F1-Skor	Kayıp Değeri	Eğitim Süresi
Xception	10	0.9889	0.97	0.97	0.99	0.0272	2 saat 7 dk 45.03 sn
	20	0.9778	0.98	0.98	0.98	0.0450	3 saat 2 dk 41.17 sn
InceptionResNetV2	10	0.9778	0.96	0.96	0.99	0.0566	2 saat 49 dk 19.54 sn
	20	0.9972	0.99	0.99	1.00	0.0120	5 saat 11 dk 39.93 sn
DenseNet201	10	0.9917	0.98	0.98	0.99	0.0357	1 saat 45 dk 29.32 sn
	20	0.9917	0.99	0.99	0.99	0.2320	4 saat 12 dk 9.17 sn
NasNetMobile	10	0.9778	0.96	0.96	0.98	0.0952	1 saat 1 dk 2.51 sn
	20	0.6361	0.57	0.25	0.73	1.0823	47 dk 51.54 sn
EfficientNetV2S	10	0.9889	0.97	0.97	0.97	0.0164	1 saat 6 dk 31.79 sn
	20	0.9944	0.97	0.97	0.97	0.0193	3 saat 29 dk 17.49 sn

Analiz yapılan modellerin Tablo 14'te bulunan eğitim süre değerleri incelendiğinde, epoch parametresi 10 seçildiğinde NasNetMobile modeli 1 saat 1 dakika 2.51 saniye ile en az sürede eğitimi tamamlamışken, InceptionResNetV2 modelinin 2 saat 49 dakika 19.54 saniye ile en uzun eğitim süresine sahip olduğu ve epoch parametresi 20 seçildiğinde NasNetMobile modeli epoch 8'de sonlanarak 47 dakika 51.54 saniye ile en az sürede eğitimi tamamlamışken, InceptionResNetV2 modelinin 5 saat 11 dakika 39.93 saniye ile en uzun eğitim süresine sahip olduğu görülmektedir. Epoch parametresi 10 olarak belirlendiğinde, Xception modelinin eğitim süresi 2 saat 7 dakika 45.03 saniye, DenseNet201 modelinin eğitim süresi 1 saat 45 dakika 29.32 saniye ve EfficientNetV2S modelinin eğitim süresi 1 saat 6 dakika 31.79 saniye olarak kaydedilmiştir. Modellerin eğitim sürelerine göre sıralandığında, en kısa süreye sahip olan model NasNetMobile, ardından EfficientNetV2S, DenseNet201, Xception ve InceptionResNetV2 olarak belirlenmiştir.

Epoch parametresi 20 olarak belirlendiğinde ise, Xception modelinin eğitim süresi 3 saat 2 dakika 41.17 saniye, DenseNet201 modelinin eğitim süresi 4 saat 12 dakika 9.17 saniye ve EfficientNetV2S modelinin eğitim süresi 3 saat 29 dakika 17.49 saniye olarak gözlemlenmiştir. Modellerin eğitim sürelerine göre sıralandığında, en kısa süreye sahip olan model NasNetMobile, ardından Xception, EfficientNetV2S, DenseNet201 ve InceptionResNetV2 olarak sıralanmaktadır. Tablo 14'te yer alan doğruluk oranlarına göre kullanılan modeller değerlendirildiğinde en yüksek doğruluk oranının epoch parametresi 20 seçilerek gerçekleşen eğitim sonunda %99.35'lik oranla InceptionResNetV2 modeliyle elde edilmişken, en düşük doğruluğun ise epoch parametresi 20 seçilerek gerçekleşen eğitim sonunda %63.61'lik oranla NasNetMobile modeline ait olduğu gözlemlenmiştir. Sonuçlar incelendiğinde en az başarılı modelden en başarılı modele doğru sıralamanın NasNetMobile (epoch 20), NasNetMobile (epoch 10), InceptionResNetV2 (epoch 10), Xception (epoch 20), Xception (epoch 10), EfficientNetV2S (epoch 10), DenseNet201 (epoch10), DenseNet201 (epoch 20), EfficientNetV2S (epoch 20) ve InceptionResNetV2 (epoch 20) olduğu söylenebilir.

Tablo 14'te yer alan kesinlik, geri çağırma, f1-skor ve kayıp değerleri açısından kullanılan modeller değerlendirildiğinde; InceptionResNetV2 (epoch 20) ve DenseNet201 (epoch 20) modellerinde %99 oranı ile kesinlik değeri en yüksek olduğu, geri çağırma değerlerinin eşit olduğu, f1-skor değerlerinin ise InceptionResNetV2 (epoch 20) modelinde %100, DenseNet201 (epoch 20) modelinde ise %99 olduğu ve en düşük kesinlik, geri çağırma, f1-skor oranlarını NasNetMobile (epoch 20) modelinde olduğu görülmektedir. Modellerin kayıp

deęeri en az olanı 0.0120 oranıyla epoch parametresi 20 seęilerek kullanılan InceptionResNetV2 ile elde edilirken, en yksek kayıp deęeri 1.0823 oranı ile epoch deęeri 20 seęilerek eęitilen NasNetMobile ile elde edilmiřtir.



6. SONUÇ VE ÖNERİLER

Endüstrideki üretim süreçlerinde hatalı ürün tespiti, kalite kontrolünün kritik bir parçasıdır. Hatalı ürünlerin erken tespit edilmesi, maliyetleri azaltabilir, müşteri memnuniyetini artırabilir ve marka itibarını koruyabilir. Günümüzde görüntü işleme ve derin öğrenme gibi teknolojiler, hatalı ürün tespiti için etkili araçlar sunmaktadır. Diz altı amputelerin protez bileşenleri arasında ayak, büyük bir öneme sahiptir, zira bu bileşen, ampute bireylerin hareketlilik ve denge sağlamalarında kritik bir rol oynamaktadır. Ayak, protez kullanıcısının yürüyüş sürecinde vücut ağırlığını taşıyarak doğal bir adım atma hareketini taklit etmeyi amaçlayan bir yapıdır. Protez ayaklar, bir bacağı veya her iki bacağı eksik olan kişilere hareketlilik sağlamak için kullanılan yapay uzuvlardır. Pedilen ayak, özellikle aktif yaşam tarzına sahip bireyler için tasarlanmış bir protez ayak çeşididir. Pedilen ayakların üretiminde kullanılan hammaddelerden biri poliüretandır. Poliüretan, organik üniteler zinciri üzerinde karbamat bağlantılarıyla birleştirilen polimerik bir malzemedir.

Pedilen ayak üretimi yapan bir firmadan temin edilen veri seti üzerinde derin öğrenme yöntemlerinin kullanılarak hatalı ürün tespiti konusunu hedeflenmektedir. Veri seti, pedilen ayakların görüntülerinin çoğaltılmasıyla elde edilen toplamda 1200 görüntüden oluşmaktadır. Kalite kontrol uzmanları tarafından yapılan değerlendirmeler sonucunda, temin edilen pedilen ayakların hatalı veya hatasız olduğu tespit edilmiş ve hatalı olarak belirlenen pedilen ayakların çukur hataları veya kabarcık hatalarından hangi tür hatalara sahip olduğu belirlenmiştir. Çalışmamızda 300 çukur hatalı, 300 kabarcık hatalı olmak üzere toplam 600 hatalı ve 600 hatasız pedilen ayak resimlerinden oluşan veri kümesi kullanılmıştır. Veri kümesinin %30'u test ve %70'i eğitim için ayrılmıştır. Veri kümesinde 210 çukur hatalı, 210 kabarcık hatalı olmak üzere toplam 420 hatalı ve 420 hatasız pedilen ayak görüntüleri eğitim için, 90 çukur hatalı 90 kabarcık hatalı olmak üzere toplam 180 hatalı ve 180 hatasız pedilen ayak görüntüleri test için yer almaktadır. Makine öğrenimi projeleri için GPU desteği sağlaması ve Python dilini kullanarak birçok popüler veri işleme ve makine öğrenimi kütüphanesini içermesinden dolayı Google Colab kullanılarak CNN sinir ağı türü modellerinden Xception, InceptionResNetV2, DenseNet201, NasNetMobile ve EfficientNetV2S modelleri uygulanarak elde edilen analiz sonuçları karşılaştırılmıştır. Eğitim sürecinin sonunda analizi yapılan modelin istenilen doğruluğa ulaşabilmesi için epoch sayısının uygun bir şekilde seçilmesi önemli olduğundan her model için epoch değeri 10 ve ya 20 seçilerek iki farklı eğitim uygulanmıştır. Analizler en yüksek doğruluk oranı dikkate

alınarak incelendiğinde epoch parametresi 20 seçilerek gerçekleşen eğitim sonunda %99.35’lik oranla InceptionResNetV2 modeliyle elde edilmiştir. Kesinlik, geri çağırma, f1-skor ve kayıp değerleri açısından kullanılan modeller değerlendirildiğinde ise; InceptionResNetV2 (epoch 20) ve DenseNet201 (epoch 20) modellerinde %99 oranı ile kesinlik değeri en yüksek olduğu, geri çağırma değerlerinin eşit olduğu, f1-skor değerlerinin ise InceptionResNetV2 (epoch 20) modelinde %100, DenseNet201 (epoch 20) modelinde ise %99 olduğu ve en düşük kesinlik, geri çağırma, f1-skor oranlarını NasNetMobile (epoch 20) modelinde olduğu görülmektedir. Modellerin kayıp değeri en az olanı 0.0120 oranıyla epoch parametresi 20 seçilerek kullanılan InceptionResNetV2 ile elde edilirken, en yüksek kayıp değeri 1.0823 oranı ile epoch değeri 20 seçilerek eğitilen NasNetMobile ile elde edilmiştir. Deneysel değerlendirmeler neticesinde, InceptionResNetV2 modelinde en üstün başarı oranlarına ulaşmıştır. 20 epoch değeri kullanıldığında, %99.72 doğruluk, %99 geri çağırma, %99 kesinlik ve %100 f1 skoru elde edilmiştir. Elde edilen sonuçlar, pedilen ayak görüntülerindeki hatalı ürünlerin etkili bir şekilde tespit edilebildiğini göstermektedir. Görüntü işleme ve derin öğrenme yöntemleri sayesinde yüksek doğruluk oranları elde edilmiş ve hatalı ürünlerin saptanması için güvenilir bir sistem oluşturulmuştur. Bu çalışmanın sonuçları, üretim süreçlerinde hatalı ürünlerin erken tespit edilmesine yardımcı olabilecek önemli bir adımı temsil etmektedir. Hatalı ürünlerin zamanında tespit edilmesi, maliyetleri azaltabilmekte, müşteri memnuniyetini artırabilmekte ve marka itibarını koruyabilmektedir. Ayrıca, üretim süreçlerinin verimliliğini artırmak için hatalı ürünlerin sebep olduğu atıkların azaltılmasına katkı sağlayabilmektedir.

Bu çalışmanın önerisi, endüstrideki üretim süreçlerinde hatalı ürün tespiti için görüntü işleme ve derin öğrenme yöntemlerinin kullanılmasının yaygınlaştırılmasıdır. Firmalar, ayak görüntüleri gibi ürünleri veya bileşenleriyle ilgili olarak bu teknikleri uygulayarak hatalı ürünleri erken aşamada tespit edebilir ve önlemler alabilirler. Ayrıca, derin öğrenme modellerinin eğitimi için daha fazla veri toplanması ve etiketlenmesi, modelin performansının daha da iyileştirilmesine yardımcı olabilir. Sonuç olarak, bu çalışma, endüstrideki üretim süreçlerinde hatalı ürün tespitinde görüntü işleme ve derin öğrenme yöntemlerinin etkin bir şekilde kullanılabileceğini göstermektedir ve bu yöntemlerin daha yaygın olarak benimsenmesi önerilmektedir.

7. KAYNAKÇA

- A. Şeker, KA Peker, AG Yüksek ve E. Delibaş, (2016) "*Derin öğrenmeyi kullanarak kumaş kusur tespiti*", 24. Sinyal İşleme ve İletişim Uygulama Konferansı (SIU) , 2016, s. 1437-1440, doi: 10.1109/SIU.2016.7496020.
- Adem, Kemal. (2013). *Tetra Brik Aseptik Kartonlardaki Şerit Dikişlerin Sayısal Görüntüleme Tabanlı Kalite Kontrolü*, Yüksek Lisans Tezi, Gaziosmanpaşa Üniversitesi, Fen Bilimleri Enstitüsü, Tokat.
- Adem, K. (2022). *Impact of activation functions and number of layers on detection of exudates using circular Hough transform and convolutional neural networks*. Expert Systems with Applications, 203, 117583.
- Adem, K., Ozguven, M. M., & Altas, Z. (2023). *A sugar beet leaf disease classification method based on image processing and deep learning*. Multimedia Tools and Applications, 82(8), 12577-12594.
- Akar, M. (2022). *Covid-19'a Karşı İnsansız Hava Aracı ile Derin Öğrenme Tabanlı Maske Tespiti*. (Yüksek Lisans Tezi), Karamanoğlu Mehmet Bey Üniversitesi Fen Bilimleri Fakültesi, Karaman.
- Akdoğan, S. (2002). *Diz Altı Amputelerde Farklı Ayakların Yürüyüş ve Ambulasyon Üzerine Etkisi*, Hacettepe Üniversitesi Sağlık Bilimleri Enstitüsü, Protez-Ortez ve Biomekanik Programı, Ankara.
- Aktaş, H. (2022). *Antep fıstığının derin öğrenme ile dış kabuk rengine göre sınıflandırılması*. Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi, 11(3), 461-466. 10(2), 409-445.
- Aldhaheri A.R., Edirisinghe E.A., (2014), *Detection and Classification of a Moving Object in a Video Stream*, In: Proc. of the Intl. Conf. on Advances in Computing and Information Technology-ACIT.
- Alkhder, H. (2021). *Face Mask Detection*. (Yüksek Lisans Tezi), Başakşehir Üniversitesi Bilgisayar Mühendisliği Ana Bilim Dalı, İstanbul.

- Alqadiri, Y. (2022). *Face Mask Detection Using Deep Learning Methods*. (Yüksek Lisans Tezi), Bahçeşehir Üniversitesi Bilgisayar Mühendisliği Ana Bilim Dalı, İstanbul.
- Artan, Cihat Tolga (2019). *Car Damage Analysis for Insurance Market Using Convolutional Neural Networks*, Istanbul Technical University, Graduate School of Science Engineering and Technology.
- Arvas, M. (2022). *Derin öğrenme algoritmaları kullanılarak döner kanatlı insansız hava aracı görüntülerinden haşhaş, kenendir ve tütün bitkilerinin tespiti*. Yüksek Lisans. Van: Van Yüzüncü Yıl Üniversitesi, Fen Bilimleri Enstitüsü
- Ashames, M. M. A. (2020). *Derin öğrenme yöntemleri kullanılarak osteoporozun belirlenmesi*. (Yüksek Lisans Tezi), Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü, Konya.
- Atcılı, A. (2020). *Yapay Sinir Ağlarında Kullanılan Optimizasyon Algoritmaları*. *Machine Learning Türkiye*. 30.12.2022 tarihinde <https://medium.com/machine-learning-t%C3%BCrkiye/yapay-sinir-a%C4%B1nda-kullan%C4%B1lan-optimizasyon-algoritmalar%C4%B1-3e87cd738cb5> sayfasından erişilmiştir.
- Ba Alawi, A. E., & Qasem, A. M. (2021). *Lightweight CNN-based Models for Masked Face Recognition*. 2021 International Congress of Advanced Technology and Engineering (ICOTEN), 1-5. <https://doi.org/10.1109/ICOTEN52080.2021.9493424>
- Bakshi, A. (2019). *What is Deep Learning? Getting Started with Deep Learning*. Retrieved October 28, 2019, from <https://www.edureka.co/blog/whatis-deep-learning>.
- Balaji S.R., Karthikeyan S. (2017), *A survey on moving object tracking using image processing*, 2017 11th International Conference on Intelligent Systems and Control (ISCO), IEEE, pp. 469–474.
- Bowker, J.H., Goldberg, B, Poonekar, P.D., (1992), Transtibial amputation, *Atlas of Limb Prosthetics: Surgical Prosthetic and Rehabilitation Principles*, Bowker, J.H., Michael, J.W.(eds), Mosby Year Book. St. Louis, 429-452.

- Bozkurt, F. (2021). *Derin Öğrenme Tekniklerini Kullanarak Akciğer X-Ray Görüntülerinden COVID-19 Tespiti*. Avrupa Bilim ve Teknoloji Dergisi, Ejosat Özel Sayı 2021 (ARACONF) , 149-156 .
- C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, (2017), “*Inception-v4, Inception-ResNet and the impact of residual connections on learning*,” in Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI’17), San Francisco, California USA, Feb. 4-9, pp. 4278-4284
- c3.ai. (2022). *Model Training*. 29.12.2022 tarihinde <https://c3.ai/glossary/data-science/model-training/> sayfasından erişilmiştir.
- Çağır, G. & Yıldırım, B. (2020). *Bir Montaj Parçasının Derin Öğrenme ve Görüntü İşleme ile Tespiti*. Journal of Intelligent Systems: Theory and Applications, 3 (2) , 31-37. DOI: 10.38016/jista.710144.
- Cerit, B. (2020). *Gerçek Zamanlı Öğrenci Takip Sistemi ve Derin Öğrenme Yaklaşımı İle Maske Tespiti*. (Yüksek Lisans Tezi), Karabük Üniversitesi Mekatronik Mühendisliği Ana Bilim Dalı, Karabük.
- Chollet, F. (2017). *Xception: Deep Learning With Depthwise Separable Convolutions*. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 1251-1258).
- Culham, E.G., Peat, M., Newell, E. (1986), *Below-knee amputation: comparison of the effect of the SACH foot and single axis foot on electromyographic patterns during locomotion*, *Prosthetics and Orthotics International*, 10, 15-22.
- Czemiecki, J.M., Gitter, A., Munro, C., (1991) *Joint moment and muscle power output characteristics of below knee amputees during running: the influence of energy storing prosthetic feet*. *Journal of Biomechanics*, 24(1), 63-75.
- Çarkacı, N. (2018). *Derin Öğrenme Uygulamalarında En Sık kullanılan Hiper-parametreler*. 30.12.2022 tarihinde <https://medium.com/deep-learning-turkiye/derin-ogrenme-uygulamalarinda-en-sik-kullanilan-hiper-parametreler-ece8e9125c4> sayfasından erişilmiştir.

- Çayırılı, M., (2006), *Yanma olayının modellenmesi ve görüntü işleme yoluyla yanma performansının optimizasyonu*, Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü, Isparta
- Çoşkun, Hüseyin (2022). *Endüstriyel Karo Üretiminde Kalite Kontrol Sürecinin Yapay Görme ve Derin Öğrenme Teknikleri ile Dijitalleştirilmesi*, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı.
- Danaei Mehr, Homay (2017). *Classification of Lung Ct Images Using Deep Convolutional Neural Network*, The Graduate School of Natural and Applied Sciences of Gazi University.
- Daşgın, Aslıhan. (2023). *COVID19 yayılımını azaltmak için yüz maskesinin evrişimsel sinir ağı modelleri ile tespiti*, Yüksek Lisans Tezi, Aksaray Üniversitesi, Lisansüstü Eğitim Enstitüsü, Isparta.
- Degadwala, S., Vyas, D., Chakraborty, U., Dider, A. R., & Biswas, H. (2021). *Yolo-v4 Deep Learning Model for Medical Face Mask Detection. 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, 209-213. <https://doi.org/10.1109/ICAIS50930.2021.9395857>
- Devrim, Mehmet Osman (2019). *Farklı Derin Evrişimsel Sinir Ağlarının Yüz İfadesi Tanıma İşlemindeki Karşılaştırılması*, Muğla Sıtkı Koçman Üniversitesi, Fen Bilimleri Enstitüsü Bilişim Sistemleri Mühendisliği Anabilim Dalı.
- 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.
- Doğan, Ö. (2020). *CNN (Convolutional Neural Networks) Nedir?*. 28.12.2022 tarihinde <https://teknoloji.org/cnn-convolutional-neural-networks-nedir/> sayfasından erişilmiştir.
- Edizer, E., (2006), *Sayısal görüntü işleme yöntemi ile tane boyut dağılımı analizi*, Yüksek Lisans Tezi, Çukurova Üniversitesi, Fen Bilimleri Enstitüsü, Adana.

- Eryılmaz, F., & Karacan, H. (2021). *Akciğer X-Ray Görüntülerinden COVID-19 Tespitinde Hafif ve Geleneksel Evrimsel Sinir Ağ Mimarilerinin Karşılaştırılması*. Düzce Üniversitesi Bilim ve Teknoloji Dergisi, ICAIAME 2021, 26-39.
- Géron, A. (2019). 14. *Deep Computer Vision Using Convolutional Neural Networks*. 26.12.2022 tarihinde <https://learning.oreilly.com/library/view/hands-on-machine-learning/9781492032632/ch14.html> sayfasından erişilmiştir.
- Güngör, S., Kaya, M., & Alhajj, R. (2021). *Renklendirilmiş BT Görüntülerinden Covid-19 Tespiti İçin Derin Öğrenmeye Dayalı Bir Yöntem*. Computer Science, 5th International Artificial Intelligence and Data Processing symposium, 391-399. <https://doi.org/10.53070/bbd.990750>
- Gürgen, Gökhan. (2019). *Derin Öğrenme İle Tekstil Tipi Tanıma*, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Hanbay, K., Üzen, H. (2017). *Nesne tespit ve takip metotları: Kapsamlı bir derleme*. Tr. Doğa ve Fen Dergisi.
- Hardas A., Vibha M., (2015), *Moving Object Detection using Background Subtraction Shadow Removal and Post Processing*, Int. J. Comput. Appl., 975–8887.
- Harriat Christa, G., J, J., K, A., & Sagayam, K. M. (2021). *CNN-based Mask Detection System Using OpenCV and MobileNetV2*. 2021 3rd International Conference on Signal Processing and Communication (ICPS), 115-119. <https://doi.org/10.1109/ICSPC51351.2021.9451688>
- Hayıt, T. ve Çınarer, G. (2022). *X-ray görüntülerini kullanarak glcm ve derin özniteliklerin birleşimine dayalı covid-19 sınıflandırılması*. İnönü Üniversitesi Sağlık Hizmetleri Meslek Yüksekokulu Dergisi, 10(1), 313-325.
- Hekim, M., Cömert, O., & Adem, K. (2020). *A hybrid model based on the convolutional neural network model and artificial bee colony or particle swarm optimization-based iterative thresholding for the detection of bruised apples*. Turkish Journal of Electrical Engineering and Computer Sciences, 28(1), 61-79.

- Herdian, Putra, G., & -. (2019). *Classification of C2C e-Commerce Product Images using Deep Learning Algorithm*. International Journal of Advanced Computer Science and Applications, 10(9). <https://doi.org/10.14569/IJACSA.2019.0100925>
- Hoang, V.-T., & Jo, K.-H. (2021). *Practical Analysis on Architecture of EfficientNet*. 2021 14th International Conference on Human System Interaction (HSI), 1-4. <https://doi.org/10.1109/HSI52170.2021.9538782>
- Jasim, A. A. (2022). *Gerçek Zamanlı Yüz Maskesi Algılama Uyarı Sistemi*. (Yüksek Lisans Tezi), Kırşehir Ahi Evran Üniversitesi İleri Teknolojiler Ana Bilim Dalı, Kırşehir.
- Kaçmaz, R. N. (2020). *Mining Colonoscopy Images For Abnormality Detection*. (Doktora Tezi), Abdullah Gül Üniversitesi Elektrik ve Bilgisayar Mühendisliği Ana Bilim Dalı, Kayseri.
- Kapp, S., Cummings, D. (1992), Prosthetic management, "Atlas of Limb Prosthetics: Surgical prosthetic and rehabilitation principles", Bowker, J.H., Michael, J.W.(eds), Mosby Year Book. St. Louis, 453-478.
- Karaköse, M., Baygın, M., Aydın, İ., Sarımaden, A., Akın, E. (2016). *Endüstriyel Sistemlerde Arka Plan Çıkarımı Tabanlı Hareketli Nesne Tespiti ve Sayılması İçin Yeni Bir Yaklaşım*. MSU Fen Bilimleri Dergisi, c. 4, sayı. 2, ss. 373-381. <https://doi.org/10.18586/msufbd.27543>
- Kassani, S. H., Kassani, P. H., Khazaeinezhad, R., Wesolowski, M. J., Schneider, K. A. ve Deters, R., (2019), *Diabetic Retinopathy Classification Using a Modified Xception Architecture*, 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 1-6.
- Kaya, B., & Önal, M. (2021). *COVID-19 Tespiti için Akciğer BT Görüntülerinin Bölütlenmesi*. Avrupa Bilim ve Teknoloji Dergisi, Ejosat Özel Sayı 2021 (ICAENS), 1296-1303 . <https://doi.org/10.31590/ejosat.1015061>
- Keras.io. (2022). *Keras documentation: The Model class*. 29.12.2022 tarihinde <https://keras.io/api/models/model/> sayfasından erişilmiştir.

- Kılıçarslan, S., Adem, K., & Çelik, M. (2021). *An overview of the activation functions used in deep learning algorithms*. Journal of New Results in Science, 10(3), 75-88.
- Koç, Yücel. (2022). *Hasarlı güneş paneli hücrelerinin derin öğrenme yöntemleriyle sınıflandırılması*, Yüksek Lisans Tezi, Amasya Üniversitesi, Fen Bilimleri Enstitüsü, Amasya.
- Konca, M. S. (2020). *Evrişimli Sinir Ağları (Convolutional Neural Networks—CNN)*. 29.12.2022 tarihinde <https://mustafaserdarkonca.medium.com/evri%C5%9Fimli-sinir-a%C4%9Flar%C4%B1-convolutional-neural-networks-cnn-74b3d4a567f9> sayfasından erişilmiştir.
- L. Deng and D. Yu, (2014), “*Deep Learning: Methods and Applications*,” Found. Trends® Signal Process., vol. 7, no. 3–4, pp. 197–387.
- Lecun, Y., Bottou, L., Bengio, Y., & 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>
- Lehmann, J.F., Price, R., Boswell-Bessette, S., Dralle, A., Questad K., Lateur, B.J. (1993), *Comprehensive analysis of energy storing prosthetic feet: flex foot and seattle foot versus standart SACH foot*. Archives of Physical Medicine and Rehabilitation, 74, 1217-1223.
- Luo W., Xing J., Milan A., Zhang X., (2014), *Multiple Object Tracking: A Literature Review*, arXiv Prepr. arXiv1409.7618.
- Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013, June). *Rectifier nonlinearities improve neural network acoustic models*. In Proc. icml (Vol. 30, No. 1, p. 3).
- May B.J. (1996), *A case study approach, Amputation and Prosthetics* FA. Davis Company. Philadelphia.
- Mehedi Shamrat, F. M. J., Chakraborty, S., Billah, Md. M., Jubair, Md. A., Islam, M. S., & Ranjan, R. (2021). *Face Mask Detection using Convolutional Neural Network (CNN) to reduce the spread of Covid-19*. 2021 5th International Conference on

Trends in Electronics and Informatics (ICOEI), 1231-1237.
<https://doi.org/10.1109/ICOEI51242.2021.9452836>

Militante, S. V., & Dionisio, N. V. (2020). *Deep Learning Implementation of Facemask and Physical Distancing Detection with Alarm Systems*. 2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE), 1-5.
<https://doi.org/10.1109/ICVEE50212.2020.9243183>

Militante, S. V., & Dionisio, N. V. (2020). *Real-Time Facemask Recognition with Alarm System using Deep Learning*. 2020 11th IEEE Control and System Graduate Research Colloquium (ICSGR), 106-110.
<https://doi.org/10.1109/ICSGRC49013.2020.9232610>

Özaydın, Kaan Çağhan (2022). *Evrişimli Sinir Ağları Yöntemlerini Kullanarak Bitki Tohum ve Fidelerini Sınıflandırma*, Ankara Üniversitesi, Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı.

Özbay, E., & Altunbey Özbay, F. (2020). *Derin Öğrenme ve Sınıflandırma Yaklaşımları ile BT görüntülerinden Covid-19 Tespiti*. Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi, c. 12, sayı. 2, ss. 211-219.
<https://doi.org/10.24012/dumf.812810>

Prince, F., Winter, D.A., Sjonnesson, G., Powell, C., Wheeldon, R.K., (1998) *Mechanical efficiency during gait of adults with trans tibial amputation: a pilot study comparing the SACH, Seattle and Golden Ankle prosthetic feet*. Journal of Rehabilitation Research and Development, 35(2), 177-185.

Rahimzadeh, M. ve Attar, A., (2020), *A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2*, Inform Med Unlocked, 19, 100360.

Ramakrishna, S., Mayer, J., Wintermantel, E., Leong, K.W., (2001), *Biomedical applications of polymer-composite materials: a review*, Composite Science and Technology 61, 1189-1224, 36p.

- Rihs, D., Polizzi, I., (1998) *Prosthetic foot design*. Department of Mechanical Engineering Victoria University of Technology. Australia.
- Risha K.P., Kumar A.C., (2016), *Novel Method of Detecting Moving Object in Video*, Procedia Technology, 24:1055–1060.
- Robert, D, Virdil, McA., Faulkner, V., (1996) "*Physical Medicine and Rehabilitation*", W.B. Saunders Company, Philadelphia, 289-320.
- Rubik's Code. (2020). *Introduction to Convolutional Neural Networks*, [Eriřim: 25.02.2021, <https://rubikscodene.net/2018/02/26/introduction-to-convolutional-neural-networks/>].
- S, T. R., Thomas, S. A., Kamath, V. ve N. K, B., (2019), *Hybrid Xception Model for Human Protein Atlas Image Classification*, 2019 IEEE 16th India Council International Conference (INDICON), 1-4
- Simonyan, K., & Zisserman, A. (2014). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv 1409.1556.
- Sin. S.W., Chow, D.H.K., Cheng, J. C. Y., (2001), *Significance of non-level walking on prosthesis fitting with particular reference to the effects of anteriorposterior alignment*. Journal of Rehabilitation Research and Development, 380, 1-6.
- Singh, K. R., Kamble, S. D., Kalbande, S. M., & Fulzele, P. (2021). *A Review on COVID-19 Face Mask Detection using CNN*. Journal of Pharmaceutical Research International, 39-45. <https://doi.org/10.9734/jpri/2021/v33i35B31896>
- Sorgun, Özcan. (2022). *X-ray görüntülerinde faster R-CNN kullanılarak yasaklı nesne tespiti*, Yüksek Lisans Tezi, Sakarya Uygulamalı Bilimler Üniversitesi, Lisansüstü Eğitim Enstitüsü, Sakarya.
- Sutaji, D. ve Yıldız, O., (2022), *LEMOXINET: Lite ensemble MobileNetV2 and Xception models to predict plant disease*, Ecological Informatics, 70, 101698.

- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. CVPR, 2818–2826.
- Şener, G, Erbahçeci, F., (2001), *Protezler*, Hacettepe Üniversitesi Fizik Tedavi ve Rehabilitasyon Yüksekokulu Yayınları, Ankara.
- Tan, F. G., Yüksel, A. S., Aydemir, E., Ersoy, M., (2021). *Derin Öğrenme Teknikleri İle Nesne Tespiti Ve Takibi Üzerine Bir İnceleme*. Avrupa Bilim ve Teknoloji Dergisi, sayı. 25, ss. 159-171. <https://doi.org/10.31590/ejosat.878552>
- Tan, M., & Le, Q. V. (2021). *EfficientNetV2: Smaller Models and Faster Training* (arXiv:2104.00298). arXiv. <http://arxiv.org/abs/2104.00298>
- Tang, P. CY., Ravji, K., Key, J. J., Mahler, D. B., Blume, P. A., Sumpio, B., (2008), *Current Prosthesis Options for Leg and Foot Amputees*, *J Am Coll Surg* Vol. 206, No. 3, 548-560, 13p.
- Tao J., Wang H., Zhang X., Li X. and Yang H., (2017) "An object detection system based on YOLO in traffic scene", 2017 6th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 2017, pp. 315-319.
- Taşçı, M., (2011), *FPGA kontrollü robotik göz*, Yüksek Lisans Tezi, Balıkesir Üniversitesi, Fen Bilimleri Enstitüsü, Balıkesir.
- Topuz S, Ülger Ö, Şener G., (2010), *Diz altı amputelerde Türkiye’de sıklıkla kullanılan protez ayakların yürüyüş ve ambulasyon aktiviteleri üzerine etkisi*. Fizyoterapi Rehabilitasyon. 2010;21(1):27-34.
- Türkoğlu, M., Hanbay, K., Saraç Sivrikaya, I., & Hanbay, D. (2020). *Derin Evrimsel Sinir Ağı Kullanılarak Kayısı Hastalıklarının Sınıflandırılması*. Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, 9(1), 334-345. <https://doi.org/10.17798/bitlisfen.562101>
- Uras, A., (2006), *Pullu uç demirlerindeki aşınmaların belirlenmesinde görüntü işleme tekniğinden yararlanma olanakları*, Yüksek Lisans Tezi, Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Bursa.

- Uysal, Neşe. (2020). *Zeytin yaprağındaki hastalıkların derin öğrenme teknikleri kullanılarak sınıflandırılması*, Yüksek Lisans Tezi, Isparta Uygulamalı Bilimler Üniversitesi, Sosyal Bilimler Enstitüsü, Isparta.
- Versluys, R., Beyl, P., Van Damme, M., Desomer, A., Van Ham, R., Lefeber, D. (2009) *Prosthetic feet: State-of-the-art review and the importance of mimicking human ankle-foot biomechanics*. Disability and Rehabilitation: Assistive Technology, 4 (2), 65-75.
- Yavuz, M. (2020). *Derin Öğrenme Modellerinin Hücre Veri Seti Üzerinde Eğitilerek Kıyaslanması ve Mobil Ortama Uyarlanması*. (Yüksek Lisans Tezi), Sakarya Uygulamalı Bilimler Üniversitesi Biyomedikal Mühendisliği Ana Bilim Dalı, Sakarya.
- Yılmaz, A., (2007), *Kamera kullanılarak görüntü işleme yoluyla gerçek zamanlı güvenli uygulaması*, Yüksek Lisans Tezi, Haliç Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Yılmaz, B. (2009), *Karbon Fiber Kompozit Ayak Protezi Tasarım Ve İmalatı*, Yüksek Lisans Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir.
- Yılmaz, E. K., Adem, K., Kılıçarslan, S., & Aydın, H. A. (2023). *Classification of lemon quality using hybrid model based on Stacked AutoEncoder and convolutional neural network*. European Food Research and Technology, 1-13.
- Yurdusev, A. A., Adem, K., & Hekim, M. (2023). *Detection and classification of microcalcifications in mammograms images using difference filter and Yolov4 deep learning model*. Biomedical Signal Processing and Control, 80, 104360.
- Zhang, E. (2021). *A Real-Time Deep Transfer Learning Model for Facial Mask Detection*. 2021 Integrated Communications Navigation and Surveillance Conference (ICNS), 1-7. <https://doi.org/10.1109/ICNS52807.2021.9441582>

EKLER



Aksaray Üniversitesi Tez Değerlendirme Formu		
Öğrencinin Adı Soyadı:		Evet
Kapak		
1	Tez Başlığı tutanaktaki başlıkla aynı mı?	
2	Kapakdaki ay ve yıl savunmaya girilen tarihle tutarlı mı?	
3	Kapak format kılavuzdaki kapak formatına uygun mu?	
4	Kapakta yazılan tüm yazılar doğru olarak verilmiş mi?	
İçindekiler		
5	Sayfa numaraları tam verilmiş mi?	
6	Şekil, Çizelge vb. listeleri verilmiş mi? Sıralaması doğru mu?	
7	Özet, Abstract, Giriş, Sonuçlar vb. bölümler var mı?	
8	Yazım hataları kontrolü yapıldı mı?	
Giriş		
9	Hazırlanan tezin önemini anlatıyor mu?	
10	İkinci ve Üçüncü dereceden başlık içermemeli kuralına uyuldu mu?	
Özet/Abstract		
11	Kılavuza uygun mu?	
12	Ay ve yıl savunmaya girilen tarihle tutarlı mı?	
13	Özet; tek sayfa, tek aralık, tek paragraf kuralına uygun olarak yazıldı mı?	
14	Bilim kodu, sayfa adedi, anahtar kelimeler ve tez danışmanı yazıldı mı?	
Kaynakça		
15	Kaynakların tamamına metin içinde atıf yapıldı mı?	
16	Kaynak formatı Kılavuzdaki kaynak formatına uygun olarak hazırlanmış mı?	
17	Atıf formatı kılavuzdaki atıf formatına uygun mu?	
Atıf Yöntemi APA 6 ☐ CMS ☐ İSNAD ☐		
Genel Değerlendirme		
18	Etik Beyan açıklaması okundu, uyuldu ve imzalandı mı?	
19	Kabul/Onay sayfası kılavuzdaki formata uygun olarak düzenlenmiş mi?	
20	Kabul /Onay sayfasında belirtilen oy birliği/oy çokluğu seçeneklerinden uygun olanı savunmayla tutarlı olacak şekilde belirlenmiş mi?	
21	Sayfa kenar boşluklar ve sayfa numaraları kılavuzdaki formatına uygun mu?	
22	Paragraf boşlukları ve metin satır aralığı kılavuza uygun olacak şekilde düzenlenmiş mi?	
23	Başlık yazımları kılavuzdaki başlık formatlarına uygun mu?	
24	Yazı tipi ve boyutu kılavuzdaki yazı tipi ve boyutu formatına uygun mu?	
25	Şekil, Çizelge vb. açıklama ve numaralandırmaları kılavuzdaki formata uygun olarak yazılmış mı?	

Bu tezin tarafımdan “Tez yazım kuralları” okunarak dikkatlice hazırlanmış olduğunu ve doğabilecek her türlü olumsuzluktan sorumlu olacağımı kabul ederim.

Öğrencinin imzası