

**BAŐKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĐİ ANABİLİM DALI
ELEKTRİK ELEKTRONİK MÜHENDİSLİĐİ TEZLİ YÜKSEK
LİSANS PROGRAMI**

**KAHVE ÇEKİRDEKLERİNE AİT GÖRÜNTÜLERİN
SINIFLANDIRILMASI**

HAZIRLAYAN

CAN ÜNAL

YÜKSEK LİSANS TEZİ

ANKARA – 2025

**BAŐKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĐİ ANABİLİM DALI
ELEKTRİK ELEKTRONİK MÜHENDİSLİĐİ TEZLİ YÜKSEK
LİSANS PROGRAMI**

**KAHVE ÇEKİRDEKLERİNE AİT GÖRÜNTÜLERİN
SINIFLANDIRILMASI**

HAZIRLAYAN

CAN ÜNAL

YÜKSEK LİSANS TEZİ

TEZ DANIŐMANI

DR. ÖĐR. ÜYESİ DENİZ KARAÇOR

ANKARA – 2025

BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Elektrik Elektronik Mühendisliği Anabilim Dalı Elektrik Elektronik Mühendisliği Tezli Yüksek Lisans Programı çerçevesinde Can ÜNAL tarafından hazırlanan bu çalışma, aşağıdaki jüri tarafından Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi: 14 / 04 / 2025

Tez Adı: Kahve Çekirdeklerine Ait Görüntülerin Sınıflandırılması

Tez Jüri Üyeleri (Unvanı, Adı - Soyadı, Kurumu)

İmza

Dr. Öğr. Üyesi Deniz KARAÇOR, Başkent Üniversitesi

.....

Prof. Dr. Mehmet Emin TAĞLUK, İnönü Üniversitesi

.....

Prof. Dr. Hamit ERDEM, Başkent Üniversitesi

.....

ONAY

Prof. Dr. Dilek ÇÖKELİLER SERDAROĞLU

Fen Bilimleri Enstitüsü Müdürü

Tarih : ... / ... /

BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
YÜKSEK LİSANS TEZ ÇALIŞMASI ORJİNALLİK RAPORU

Tarih: 14 / 04 / 2025

Öğrencinin Adı, Soyadı : Can ÜNAL

Öğrencinin Numarası : 22210225

Anabilim Dalı : Elektrik Elektronik Mühendisliği

Programı : Tezli Yüksek Lisans Programı

Danışmanın Unvanı/Adı, Soyadı : Dr. Öğr. Üyesi Deniz KARAÇOR

Tez Başlığı : Kahve Çekirdeklerine Ait Görüntülerin Sınıflandırılması

Yukarıda başlığı belirtilen Yüksek Lisans/Doktora tez çalışmamın; Giriş, Ana Bölümler ve Sonuç Bölümünden oluşan, toplam 124 sayfalık kısmına ilişkin, 25 / 04 / 2024 tarihinde şahsım/tez danışmanım tarafından Turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı %11'dir. Uygulanan filtrelemeler:

1, Kaynakça hariç

2, Alıntılar hariç

3, Beş (5) kelimedenden daha az örtüşme içeren metin kısımları hariç "Başkent Üniversitesi Enstitüleri Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Usul ve Esaslarını" inceledim ve bu uygulama esaslarında belirtilen azami benzerlik oranlarına tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Öğrenci İmzası:.....

ONAY

Tarih: ... / ... / 20...

Öğrenci Danışmanı:

Dr. Öğr. Üyesi Deniz Karaçor

.....

TEŐEKKÜR

Tez süresi boyunca, bana danışmanlık ederek yolumu aydınlatan, yardımlarını esirgemedен bana rehberlik eden danışman hocam Dr. Öğretim Üyesi Deniz KARAÇOR'a sonsuz teşekkür ederim.

Yüksek lisans eğitim hayatım boyunca, yardıma ihtiyaç duyduğum her an yanımda olan, sorunlarımı dinleyip her zaman çözüm üretmek için çaba sarf eden Volkan HAMURCU'ya teşekkür ederim.

Doğduğum günden bu yana sevgi ve desteklerini her zaman hissettiğim, zor zamanlarda bana cesaret veren ve her adımda beni teşvik eden aileme teşekkür ederim.

ÖZET

Can ÜNAL

KAHVE ÇEKİRDEKLERİNE AİT GÖRÜNTÜLERİN SINIFLANDIRILMASI

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Elektrik/Elektronik Mühendisliği Anabilim Dalı

2025

Kahve ağaçlarından toplanan yeşil kahve çekirdeklerinin farklı türlere sahip olmasından dolayı tatların birbirine karışmaması ve tüketiciye sunulan kahvenin belirli bir standartta olması kahvenin ticareti açısından büyük önem taşımaktadır. Kahvenin ayıklanma ve sınıflandırma süreçleri genellikle insan gözüyle gerçekleştirildiğinden zaman alıcı, maliyetli ve hatalara açık olabilmektedir. Bu nedenle görüntü işleme, makine öğrenmesi ve derin öğrenme yöntemlerinin kullanılması kahve çekirdeklerinin otomatik olarak sınıflandırılmasını sağlayarak süreçleri hızlandırmakta ve doğruluğu artırmaktadır. Makine öğrenmesi teknikleri, kahve çekirdeklerinin boyut, renk, yoğunluk gibi fiziksel özelliklerini analiz ederek kalite kontrol süreçlerini optimize etmeye yardımcı olmaktadır.

Bu tez çalışmasında yeşil çekirdeğin dört farklı türüne ait 8000 veriden oluşan USK-Coffee veri seti kullanılmıştır. Veri setine MobileNetV2, ResNet18, VGG16 ve DenseNet201 modelleri özellik çıkarım yöntemi olarak veri setine uygulanarak K-En Yakın Komşu (KNN), Naive Bayes, Karar Ağacı ve Destek Vektör Makinesi (SVM) gibi farklı makine öğrenmesi algoritmaları kullanılarak sınıflandırılmıştır. Sınıflandırma modellerinin performansındaki değişiklikleri incelemek amacıyla, Temel Bileşen Analizi (PCA) ve Yönlendirilmiş Gradyanların Histogramı (HoG) gibi ek özellik çıkarma ve boyut indirgeme yöntemleri de kullanılarak sonuçlar karşılaştırılmıştır.

Tüm uygulamalar karşılaştırıldığı zaman %91,99 doğruluk oranı ile DenseNet201 ve PCA ile çıkarılan özelliklere uygulanan SVM algoritmasının en yüksek doğruluk oranına ulaştığı görülmüştür.

ANAHTAR KELİMELER: Kahve çekirdeği sınıflandırması, USK-coffee veri seti, MobileNetV2, ResNet18, VGG16, DenseNet201, K-NN, Naive Bayes, Karar Ağacı, SVM, PCA, HoG

ABSTRACT

Can ÜNAL

CLASSIFICATION OF COFFEE BEAN IMAGES

Başkent University Institute of Science and Engineering

Department of Electrical and Electronics Engineering

2025

The variation in green coffee beans collected from coffee trees significantly impacts coffee trade, as it affects flavor consistency and quality standards. Traditional sorting and classification processes are typically performed manually, making them time-consuming, costly, and prone to errors. Therefore, utilizing image processing, machine learning, and deep learning techniques enables the automatic classification of coffee beans, accelerating processes and enhancing accuracy. Machine learning techniques help optimize quality control by analyzing physical characteristics such as size, color, and density.

In this thesis, the USK-Coffee dataset, consisting of 8,000 samples from four different green coffee bean types, was used. Feature extraction was performed using MobileNetV2, ResNet18, VGG16, and DenseNet201 models, followed by classification with machine learning algorithms such as K-Nearest Neighbors (KNN), Naive Bayes, Decision Tree, and Support Vector Machine (SVM). To analyze classification performance variations, additional feature extraction and dimensionality reduction techniques such as Principal Component Analysis (PCA) and Histogram of Oriented Gradients (HoG) were applied, and the results were compared.

Among all implementations, the highest accuracy rate of 91.99% was achieved using the DenseNet201 model combined with PCA-extracted features applied to the SVM algorithm.

KEYWORDS: Coffee bean classification, USK-coffee dataset, MobileNetV2, ResNet18, VGG16, DenseNet201, K-NN, Naive Bayes, Decision Tree, SVM, PCA, HoG

İÇİNDEKİLER

TEŞEKKÜR.....	i
ÖZET.....	ii
ABSTRACT.....	iii
İÇİNDEKİLER.....	iv
TABLolar LİSTESİ.....	vi
ŞEKİLLER LİSTESİ.....	xi
SİMGELER VE KISALTMALAR LİSTESİ.....	xiv
1. GİRİŞ.....	1
2. VERİ SETİ.....	7
3. YÖNTEM.....	9
3.1. Derin Öğrenme Modelleri.....	12
3.1.1. Konvolüsyonel Sinir Ağları (CNN).....	12
3.2. Temel Bileşenler Analizi (PCA).....	18
3.3. Yönlü Gradyanlar Histogramı (HoG).....	20
3.4. Makine Öğrenmesi Modelleri.....	22
3.4.1. K-En Yakın Komşu (KNN) Algoritması.....	22
3.4.2. Naive Bayes Algoritması.....	23
3.4.3. Karar Ağacı Algoritması.....	24
3.4.4. SVM Algoritması.....	25
3.5. Performans Metrikleri.....	27
3.5.1. Karmaşıklık Matrisi (Confusion Matrix).....	27
3.5.2. Doğruluk (Accuracy).....	28
3.5.3. Hassasiyet (Precision).....	28
3.5.4. Duyarlılık (Recall).....	28
3.5.5. F1 Skoru.....	29
3.6. Çapraz Doğrulama (Cross Validation).....	29
4. DENEY SONUÇLARI.....	31
4.1. CNN Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması.....	33
4.1.1. MobileNetV2 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması.....	33
4.1.2. ResNet18 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması.....	40

4.1.3. VGG16 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	47
4.1.4. DenseNet201 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması.....	54
4.2. CNN ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	61
4.2.1. MobileNetV2 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	61
4.2.2. ResNet18 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	68
4.2.3. VGG16 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	75
4.2.4. DenseNet201 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	82
4.3. CNN ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	89
4.3.1. MobileNetV2 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	89
4.3.2. ResNet18 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	95
4.3.3. VGG16 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	102
4.3.4. DenseNet201 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması	108
5. TARTIŞMA VE SONUÇ.....	116
KAYNAKLAR.....	119

TABLULAR LİSTESİ

Sayfa

Tablo 4.1 MobileNetV2+KNN Modelinin Ortalama Performans Metrikleri	34
Tablo 4.2 MobileNetV2+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7).....	35
Tablo 4.3 MobileNetV2+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7).....	35
Tablo 4.4 MobileNetV2+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	36
Tablo 4.5 MobileNetV2+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	36
Tablo 4.6 MobileNetV2+Naive Bayes Modelinin Ortalama Performans Metrikleri.....	36
Tablo 4.7 MobileNetV2+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi.....	37
Tablo 4.8 MobileNetV2+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri.....	37
Tablo 4.9 MobileNetV2+Karar Ağacı Modelinin Ortalama Performans Metrikleri.....	38
Tablo 4.10 MobileNetV2+SVM ile Elde Edilen Karmaşıklık Matrisi.....	39
Tablo 4.11 MobileNetV2+SVM Modelinin Sınıf Bazında Performans Metrikleri	39
Tablo 4.12 MobileNetV2+SVM Modelinin Ortalama Performans Metrikleri	39
Tablo 4.13 ResNet18+KNN Modelinin Ortalama Performans Metrikleri.....	41
Tablo 4.14 ResNet18+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)	41
Tablo 4.15 ResNet18+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)	41
Tablo 4.16 ResNet18+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	42
Tablo 4.17 ResNet18+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	43
Tablo 4.18 ResNet18+Naive Bayes Modelinin Ortalama Performans Metrikleri	43
Tablo 4.19 ResNet18+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	44
Tablo 4.20 ResNet18+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	44
Tablo 4.21 ResNet18+Karar Ağacı Modelinin Ortalama Performans Metrikleri	44
Tablo 4.22 ResNet18+SVM ile Elde Edilen Karmaşıklık Matrisi.....	45
Tablo 4.23 ResNet18+SVM Modelinin Sınıf Bazında Performans Metrikleri.....	45
Tablo 4.24 ResNet18+SVM Modelinin Ortalama Performans Metrikleri.....	46
Tablo 4.25 VGG16+KNN Modelinin Ortalama Performans Metrikleri.....	47
Tablo 4.26 VGG16+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)	48
Tablo 4.27 VGG16+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)	48
Tablo 4.28 VGG16+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	49
Tablo 4.29 VGG16+Naive Bayes Modeli Sınıf Bazında Performans Metrikleri	49
Tablo 4.30 VGG16+Naive Bayes Modeli Ortalama Performans Metrikleri	50
Tablo 4.31 VGG16+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	51
Tablo 4.32 VGG16+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	51
Tablo 4.33 VGG16+Karar Ağacı Modelinin Ortalama Performans Metrikleri	51

Tablo 4.34 VGG16+SVM ile Elde Edilen Karmaşıklık Matrisi	52
Tablo 4.35 VGG16+SVM Modelinin Sınıf Bazında Performans Metrikleri	53
Tablo 4.36 VGG16+SVM Modelinin Ortalama Performans Metrikleri	53
Tablo 4.37 DenseNet201+KNN Modelinin Ortalama Performans Metrikleri	55
Tablo 4.38 DenseNet201+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)	55
Tablo 4.39 DenseNet201+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)	55
Tablo 4.40 DenseNet201+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	56
Tablo 4.41 DenseNet201+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	57
Tablo 4.42 DenseNet201+Naive Bayes Modelinin Ortalama Performans Metrikleri	57
Tablo 4.43 DenseNet201+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	58
Tablo 4.44 DenseNet201+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	58
Tablo 4.45 DenseNet201+Karar Ağacı Modelinin Ortalama Performans Metrikleri	59
Tablo 4.46 DenseNet201+SVM ile Elde Edilen Karmaşıklık Matrisi	60
Tablo 4.47 DenseNet201+SVM Modelinin Sınıf Bazında Performans Metrikleri	60
Tablo 4.48 DenseNet201+SVM Modelinin Ortalama Performans Metrikleri	60
Tablo 4.49 MobileNetV2+PCA+KNN Modelinin Ortalama Performans Metrikleri	62
Tablo 4.50 MobileNetV2+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7).....	62
Tablo 4.51 MobileNetV2+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7).....	63
Tablo 4.52 MobileNetV2+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi.....	64
Tablo 4.53 MobileNetV2+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri....	64
Tablo 4.54 MobileNetV2+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri.....	64
Tablo 4.55 MobileNetV2+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi.....	65
Tablo 4.56 MobileNetV2+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri.....	66
Tablo 4.57 MobileNetV2+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri.....	66
Tablo 4.58 MobileNetV2+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi.....	67
Tablo 4.59 MobileNetV2+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri.....	67
Tablo 4.60 MobileNetV2+PCA+SVM Modelinin Ortalama Performans Metrikleri	67
Tablo 4.61 ResNet18+PCA+KNN Modelinin Ortalama Performans Metrikleri	69
Tablo 4.62 ResNet18+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)	69
Tablo 4.63 ResNet18+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)	70
Tablo 4.64 ResNet18+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	71
Tablo 4.65 ResNet18+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	71
Tablo 4.66 ResNet18+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri	71
Tablo 4.67 ResNet18+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	72
Tablo 4.68 ResNet18+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	72
Tablo 4.69 ResNet18+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri	73

Tablo 4.70 ResNet18+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi	74
Tablo 4.71 ResNet18+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri.....	74
Tablo 4.72 ResNet18+PCA+SVM Modelinin Ortalama Performans Metrikleri.....	74
Tablo 4.73 VGG16+PCA+KNN Modelinin Ortalama Performans Metrikleri	76
Tablo 4.74 VGG16+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)	76
Tablo 4.75 VGG16+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)	76
Tablo 4.76 VGG16+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	77
Tablo 4.77 VGG16+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	78
Tablo 4.78 VGG16+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri	78
Tablo 4.79 VGG16+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	79
Tablo 4.80 VGG16+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	79
Tablo 4.81 VGG16+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri	80
Tablo 4.82 VGG16+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi	81
Tablo 4.83 VGG16+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri.....	81
Tablo 4.84 VGG16+PCA+SVM Modelinin Ortalama Performans Metrikleri	81
Tablo 4.85 DenseNet201+PCA+KNN Modelinin Ortalama Performans Metrikleri.....	83
Tablo 4.86 DenseNet201+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi (k=5)	83
Tablo 4.87 DenseNet201+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=5)	83
Tablo 4.88 DenseNet201+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	84
Tablo 4.89 DenseNet201+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	85
Tablo 4.90 DenseNet201+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri	85
Tablo 4.91 DenseNet201+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	86
Tablo 4.92 DenseNet201+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	86
Tablo 4.93 DenseNet201+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri	86
Tablo 4.94 DenseNet201+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi	87
Tablo 4.95 DenseNet201+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri	87
Tablo 4.96 DenseNet201+PCA+SVM Modelinin Ortalama Performans Metrikleri.....	88
Tablo 4.97 MobileNetV2+HoG+KNN Modelinin Ortalama Performans Metrikleri	89
Tablo 4.98 MobileNetV2+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7).....	90
Tablo 4.99 MobileNetV2+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7).....	90
Tablo 4.100 MobileNetV2+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	91
Tablo 4.101 MobileNetV2+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri ..	91
Tablo 4.102 MobileNetV2+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri	92
Tablo 4.103 MobileNetV2+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi.....	93
Tablo 4.104 MobileNetV2+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri...	93
Tablo 4.105 MobileNetV2+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri	93

Tablo 4.106 MobileNetV2+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi.....	94
Tablo 4.107 MobileNetV2+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri	94
Tablo 4.108 MobileNetV2+HoG+SVM Modelinin Ortalama Performans Metrikleri	95
Tablo 4.109 ResNet18+HoG+KNN Modelinin Ortalama Performans Metrikleri.....	96
Tablo 4.110 ResNet18+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)	97
Tablo 4.111 ResNet18+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7).....	97
Tablo 4.112 ResNet18+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	98
Tablo 4.113 ResNet18+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	98
Tablo 4.114 ResNet18+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri	98
Tablo 4.115 ResNet18+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	99
Tablo 4.116 ResNet18+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	100
Tablo 4.117 ResNet18+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri.....	100
Tablo 4.118 ResNet18+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi.....	101
Tablo 4.119 ResNet18+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri.....	101
Tablo 4.120 ResNet18+HoG+SVM Modelinin Ortalama Performans Metrikleri.....	101
Tablo 4.121 VGG16+HoG+KNN Modelinin Ortalama Performans Metrikleri.....	103
Tablo 4.122 VGG16+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)	103
Tablo 4.123 VGG16+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)	103
Tablo 4.124 VGG16+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	104
Tablo 4.125 VGG16+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	105
Tablo 4.126 VGG16+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri	105
Tablo 4.127 VGG16+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	106
Tablo 4.128 VGG16+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri	106
Tablo 4.129 VGG16+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri.....	106
Tablo 4.130 VGG16+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi	107
Tablo 4.131 VGG16+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri.....	107
Tablo 4.132 VGG16+HoG+SVM Modelinin Ortalama Performans Metrikleri	108
Tablo 4.133 DenseNet201+HoG+KNN Modelinin Ortalama Performans Metrikleri.....	109
Tablo 4.134 DenseNet201+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi (k=5)	110
Tablo 4.135 DenseNet201+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=5)..	110
Tablo 4.136 DenseNet201+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi	111
Tablo 4.137 DenseNet201+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri	111
Tablo 4.138 DenseNet201+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri	111
Tablo 4.139 DenseNet201+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi	112
Tablo 4.140 DenseNet201+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri .	113
Tablo 4.141 DenseNet201+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri.....	113

Tablo 4.142 DenseNet201+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi	114
Tablo 4.143 DenseNet201+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri	114
Tablo 4.144 DenseNet201+HoG+SVM Modelinin Ortalama Performans Metrikleri	114
Tablo 5.1 VGG16 ile Uygulanan Algoritmalara Ait Doğruluk Oranları	116
Tablo 5.2 Densenet201 ile Uygulanan Algoritmalara Ait Doğruluk Oranları	116
Tablo 5. 3 MobileNetV2 ile Uygulanan Algoritmalara Ait Doğruluk Oranları.....	116
Tablo 5. 4 ResNet18 ile Uygulanan Algoritmalara Ait Doğruluk Oranları	117

ŞEKİLLER LİSTESİ

Sayfa

Şekil 2.1 Longberry Kahve Çekirdeği Görüntüleri	7
Şekil 2.2 Peaberry Kahve Çekirdeği Görüntüleri.....	8
Şekil 2.3 Premium Kahve Çekirdeği Görüntüleri	8
Şekil 2.4 Defect Kahve Çekirdeği Görüntüleri	8
Şekil 3.1 Uygulama-1 Blok Şeması	9
Şekil 3.2 Uygulama-2 Blok Şeması	10
Şekil 3.3 Uygulama-3 Blok Şeması	11
Şekil 3.4 Basit Konvolüsyonel Sinir Ağı (CNN) Mimarisi.....	12
Şekil 3.5 VGG16 Mimarisi	15
Şekil 3.6 DenseNet201 Mimarisi	16
Şekil 3.7 MobileNetV2 Mimarisi.....	17
Şekil 3.8 ResNet18 Mimarisi	18
Şekil 3.9 KNN Algoritması Örneği.....	23
Şekil 3.10 Karar Ağacının Yapısı	24
Şekil 3.11 SVM Algoritmasının Yapısı	27
Şekil 3.12 Karmaşıklık Matrisi	28
Şekil 3.13 5 Katmanlı Çapraz Doğrulama Örneği	30
Şekil 4.1 MobileNetV2+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)..	35
Şekil 4.2 MobileNetV2+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	37
Şekil 4.3 MobileNetV2+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği.	38
Şekil 4.4 MobileNetV2+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği.....	40
Şekil 4.5 ResNet18+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	42
Şekil 4.6 ResNet18+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	43
Şekil 4.7 ResNet18+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	45
Şekil 4.8 ResNet18+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	46
Şekil 4.9 VGG16+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	49
Şekil 4.10 VGG16+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği.....	50
Şekil 4.11 VGG16+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	52
Şekil 4.12 VGG16+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	54
Şekil 4.13 DenseNet201+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	56
Şekil 4.14 DenseNet201+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	58

Şekil 4.15 DenseNet201+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	59
Şekil 4.16 DenseNet201+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	61
Şekil 4.17 MobileNetV2+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	63
Şekil 4.18 MobileNetV2+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	65
Şekil 4.19 MobileNetV2+PCA+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	66
Şekil 4.20 MobileNetV2+PCA+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	68
Şekil 4.21 ResNet18+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	70
Şekil 4.22 ResNet18+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	72
Şekil 4.23 ResNet18+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	73
Şekil 4.24 ResNet18+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	75
Şekil 4.25 VGG16+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	77
Şekil 4.26 VGG16+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	79
Şekil 4.27 VGG16+PCA+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	80
Şekil 4.28 VGG16+PCA+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	82
Şekil 4.29 DenseNet201+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=5)	84
Şekil 4.30 DenseNet201+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	85
Şekil 4.31 DenseNet201+PCA+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	87
Şekil 4.32 DenseNet201+PCA+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	88
Şekil 4.33 MobileNetV2+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	91
Şekil 4.34 MobileNetV2+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	92
Şekil 4.35 MobileNetV2+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	94

Şekil 4.36 MobileNetV2+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	95
Şekil 4.37 ResNet18+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	97
Şekil 4.38 ResNet18+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	99
Şekil 4.39 ResNet18+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	100
Şekil 4.40 ResNet18+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	102
Şekil 4.41 VGG16+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)	104
Şekil 4.42 VGG16+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	105
Şekil 4.43 VGG16+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	107
Şekil 4.44 VGG16+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	108
Şekil 4.45 DenseNet201+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=5)	110
Şekil 4.46 DenseNet201+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	112
Şekil 4.47 DenseNet201+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	113
Şekil 4.48 DenseNet201+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği	115

SİMGELER VE KISALTMALAR LİSTESİ

ANN	Artificial Neural Network
CIE	Commission Internationale d'Eclairage
CNN	Convolutional Neural Network
GLCM	Gray Level Co-occurrence Matrix
HoG	Histogram of Oriented Gradients
KNN	K-Nearest Neighbors
MLP	Multi-Layer Perceptron
PCA	Principal Component Analysis
RGB	Red Green Blue
SCAA	Specialty Coffee Association of America
SGD	Stochastic Gradient Descent
SVM	Support Vector Machine

1. GİRİŞ

Sudan sonra en çok tüketilen ikinci içecek olan kahve, 70'ten fazla ülkede yetiştirilen, dünya çapında milyarlarca insanın hayatıyla yakından ilgili olan ve petrolden sonra dünya çapında ticareti en fazla yapılan ikinci şey haline gelmiştir [1]. Ticari açıdan önemli bir yere sahip olan kahvenin ne zaman ve nerede keşfedildiği hakkında net bir bilgi bulunmamaktadır. Etiyopya'nın yüksek rakımlı bölgelerine has olduğu düşünülmektedir.

Kahve, genellikle bol yağışlı ve sıcak bir iklime sahip tropikal bölgelerde yetişmektedir. Bu bölgelerde yetiştirilen kahve ağaçlarından belirli dönemlerde kahve kirazı olarak adlandırılan meyveler toplanır. Daha sonrasında toplanan meyveler güneş ışığında veya kurutma yataklarında kurutulur. Kuruyan meyvelerin içerisindeki çekirdekler ayrılır ve kahvenin kavrulmadan önceki hali olan yeşil çekirdek formuna ulaşılmış olur.

Kahve tüketicileri için en önemli şey kahvenin tadıdır [2]. Kahvenin kavrulması konusu, kahveyi keyifli hale getiren karmaşık tatların oluşmasında muhtemelen en önemli faktördür. Kavurma işlemi sırasında kahve çekirdekleri birçok karmaşık ve tam olarak tanımlanmamış kimyasal reaksiyonlara maruz kalır. Bu da önemli fiziksel değişikliklere ve kahvenin duyuşal niteliklerinden sorumlu maddelerin oluşumuna yol açar [3]. Kavrulmuş kahve çekirdeğinin rengi onu tanımlamanın en iyi yoludur [4]. Hafifçe kavrulmuş kahve çekirdekleri en yüksek asit oranına ve kafeine sahiptir. Orta kavrulmuş kahve çekirdeği daha koyu kahverengidir. Kavrulmuş kahve çekirdeğinin rengi, tadının, aromasının ve asitliğinin en iyi göstergesidir [5]. Çok koyu kavrulmuş kahve yassı, acı ve çok güçlü dumansı bir tada sahiptir. Kavrulmuş kahve çekirdeğinin renginin çok koyu olmaması gerekir çünkü kahve çekirdeğinin özelliklerini yok eder. Kahve çekirdeklerinin kavrulma derecesine göre sınıflandırılması, renk ve doku gibi fiziksel kalite özelliklerinin incelenmesine dayanır [6].

Kahvenin tadını etkileyen bir diğer etmen ise kahvenin çekirdeğinin türüdür. Toplanan meyvelerden elde edilen her yeşil çekirdek aynı özelliklere sahip olmaz. Bazı çekirdekler uzun ve büyük bir yapıya sahip olurken bazıları daha küçük ve yuvarlak bir yapıya sahip olabilir. Ayrıca toplanan yeşil kahve çekirdeklerinin içerisinde yanlış işlendiği veya böcekler tarafından yendiği için hatalı olan kahve çekirdekleri de bulunabilmektedir. Yeşil kahve çekirdeklerinin sahip olduğu bu özelliklere göre farklı isimleri bulunmaktadır. İsimlerinin farklılığı dışında bir başka konu ise her bir yeşil kahve çekirdeği türünün farklı kavrulma

özelliğine sahip olmasıdır. Aynı anda kavrulursa beklenmeyen bir tada sahip olacak farklı yeşil çekirdeklerin kavrulma işlemi öncesinde türlerine göre sınıflandırılıp ayrıştırılması gerekmektedir. Genellikle bu ayıklama işlemi çiftçiler veya tedarikçiler tarafından çekirdeklerin tek tek incelenerek sınıflara ayrılması şeklinde yapılmaktadır. Kahve çekirdeklerinin sınıflandırılması işleminin insan gözüyle yapılmasının hem yavaş hem de yanlış yapmaya müsait bir yöntem olmasından kaynaklı olarak başka çözümler bulunması gerekmektedir.

Jun Noel C. Sarino ve arkadaşları [4] tarafından yapılan çalışmada; bir kahve çekirdeğinin farklı kavrulma dereceleri görüntü işleme ve ANN (Artificial Neural Network, Yapay Sinir Ağları) kullanılarak sınıflandırılmıştır. Bu çalışmada kahve çekirdeklerine ait R, G ve B bileşenleri gibi özellikler çıkarılmıştır. Daha sonrasında ANN’de girdi olarak RGB (Red Green Blue, Kırmızı Yeşil Mavi) değerlerini kullanarak kahve çekirdeklerinin kavrulma derecesini hafif kavrulmuş, orta kavrulmuş ve çok koyu kavrulmuş olarak sınıflandırmak için ANN yöntemi kullanılmıştır. 60’ı hafif kavrulmuş, 60’ı orta kavrulmuş ve 60’ı çok koyu kavrulmuş olmak üzere toplam 180 tane kahve çekirdeğinin girdi olarak alındığı bu yöntem kahve çekirdeklerinin kavrulma derecesini %97,22 doğrulukla tanımlayabilmektedir.

Edwin R. Arboleda ve arkadaşları [7] tarafından yapılan çalışmada; siyah çekirdek adı verilen kusurlu ve kalitesiz türünün, kaliteli kahve çekirdeklerinden ayrıştırılması için bir metod sunmaktadır. MATLAB kullanılarak normal ve kalitesiz olarak nitelendirilen kahve çekirdeklerinin RGB değer aralıkları çıkarılmıştır. Bu değer aralıkları kullanılarak sadece normal kahve çekirdeklerinin gösterildiği, kalitesiz kahve çekirdeklerinin gözükmediği bir algoritma geliştirilmiştir. Bu çalışmada 70 normal ve 50 siyah kahve çekirdeği eğitim için kullanılırken, 35 normal ve 25 siyah kahve çekirdeği test için kullanılmıştır. Sonuç olarak, test görüntülerindeki siyah kahve çekirdeklerini ortadan kaldırmak için %100'lük bir doğruluk oranı elde edilmiştir.

Edwin R. Arboleda ve arkadaşları [8] tarafından yapılan bu çalışmada; Robusta, Liberica ve Excelsa adı verilen 3 farklı kahve türünün sınıflandırılması için ANN ve KNN (K-Nearest Neighbors, K-En Yakın Komşu) yöntemleri kullanılmıştır. Öncelikle çekirdeğin alanı, çevresi, eşdeğer çapı ve yuvarlaklık yüzdesi gibi morfolojiye dayalı önemli kahve çekirdeği özellikleri, 195 eğitim görüntüsünden ve 60 test görüntüsünden çıkarılmıştır. Daha sonrasında kahve çekirdeklerini otomatik olarak sınıflandırmak için ANN ve KNN

kullanılmıştır. ANN kullanıldığında %96,66 doğruluk oranı elde edilirken, 'K-NN kullanıldığında en yüksek doğruluk oranı, k=1 iken, %84,12 olarak bulunmuştur.

T H Nasution ve U Andayani [9] tarafından yapılan bu çalışmada; görüntüyü işleyerek ve geri yayımlı sinir ağı ile sınıflandırarak görüntülerin kahve çekirdeklerinin kavrulma düzeyini dijital olarak tanıyabilecek bir yöntem geliştirilmiştir. Bu çalışmada 16 farklı kavrulma derecesine ait, toplam 160 tane veri kullanılmıştır. Görüntüler siyah beyaza çevrilerek, özelliklerinin çıkarılması için GLCM (Gray Level Co-occurrence Matrix, Gri Seviye Eşzamanlılık Matrisi) yöntemi uygulanmıştır. Daha sonrasında normalleştirme tekniği kullanılarak veri kümesinin değer niteliğini belirli bir aralıkta tutulması amaçlanmıştır. En sonunda ise kavrulmuş kahve çekirdeğinin 16 seviyesinin görüntü sınıflandırmasını gerçekleştirmek için geri yayılım metodu kullanılmıştır. Sonuç olarak önerilen yöntemin kavrulmuş kahve çekirdeklerinin seviyesini %97,5 doğrulukla tanımlayabilmiştir.

Emanuelle Morais de Oliveira ve arkadaşları [10] tarafından yapılan bu çalışmada; yeşil kahve çekirdeklerinin CIE (Commission Internationale d'Eclairage) $L^*a^*b^*$ ölçümlerini veren ve bunları renklerine göre sınıflandıran bir bilgisayarlı görme sistemi oluşturulması amaçlanmıştır. Dönüşüm modeli olarak ANN seçilmiş olup ve kahve çekirdeklerini beyazımsı, kamış yeşili, yeşil ve mavimsi-yeşil olmak üzere dört gruba sınıflandırmak için Naive Bayes sınıflandırıcısı kullanılmıştır. ANN modelleri %1,15'lik bir genelleme hatası elde etmiştir ve Bayes sınıflandırıcısı tüm örnekleri beklenen sınıflara göre sınıflandırmıştır.

Yavuz Unal ve arkadaşları [11] tarafından yapılan bu çalışmada; SqueezeNet, Inception V3, VGG16 ve VGG19 olmak üzere 4 farklı CNN (Convolutional Neural Network, Konvolüsyonel Sinir Ağı) tabanlı modelden yararlanarak transfer öğrenme yöntemiyle 3 farklı kahve çekirdeğine (Espresso, Kenya ve Starbucks Pike Place) ait görüntülerin kullanılarak bunların sınıflandırılması amaçlanmıştır. Çalışmada kahve türlerine ait toplam 1554 adet kahve çekirdeği görüntüsü toplanmıştır ve elde edilen görüntüler ile model eğitimi ve model test işlemleri gerçekleştirilmiştir. Modelleri test etmek için çapraz doğrulama yöntemi kullanılmıştır. Yapılan testler sonucunda modellerin ortalama sınıflandırma başarısı SqueezeNet için %87,3, Inception V3 için %81,4, VGG16 için %78,2 ve VGG19 için %72,5 olarak belirlenmiştir.

Carlito Pinto ve arkadaşları [29] kahve üreticilerinin ürün kalitesini artırmak amacıyla bir otomatik kahve çekirdeği sıralama sistemi geliştirmiştir. Çalışmada kahve çekirdeklerinin ön ve arka yüzlerinin fotoğrafları çekilerek 6500 kahve çekirdeğinde 13000 görüntü elde edilmiştir. Görüntüler manuel olarak solgun, siyahi ekşi, kırık, peaberry ve kusursuz olarak sınıflara ayrılmıştır. Çalışmanın ilk aşamasında, yeşil kahve çekirdeklerinin görüntülerini işleyen bir sistem oluşturulmuş ve bu görüntülerdeki kusurlar CNN kullanılarak sınıflandırılmıştır. Bu yöntem, kahve çekirdeklerindeki çeşitli kusurları (örneğin, siyah, solmuş, ekşi, kırık ve peaberry) %72,4 ile %98,7 arasında bir doğruluk oranıyla ayırt edebilmiştir.

Birhanu Turi ve arkadaşları [30] Etiyopya'nın farklı bölgelerinden gelen kahve çekirdeklerinin kalite farklılıklarını belirlemek amacıyla geliştirilmiş bir bilgisayar algoritmasını incelemektedir. Araştırmanın temel hedefi, dört farklı kahve yetiştirme bölgesinden (Hararghe, Jimma, Wollega ve Yirgacheffe) gelen kahve çekirdeklerini otomatik olarak sınıflandıran bir algoritma geliştirmektir. Her bölgenin 40 görüntü olmak üzere toplam 160 adet kahve çekirdeği görüntüsü veri seti olarak kullanılmıştır. Bu veri setindeki görüntülerden renk, morfolojik (şekil ve boyut) ve doku özellikleri çıkarılmıştır. Daha sonrasında ANN kullanılarak kahve çekirdekleri dört bölgeye (Hararghe, Jimma, Wollega ve Yirgacheffe) göre sınıflandırılmıştır. Bu yöntem kahve çekirdeklerini %87,5 ile %100 arasında bir doğruluk oranıyla sınıflandırabilmiştir.

Nen-Fu Hang ve arkadaşları [31] yeşil kahve çekirdeklerinde Specialty Coffee Association of America (SCAA) tarafından belirlenen kusurlu kahveleri kusuru olmayan çekirdeklerden ayırmayı amaçlamışlardır. Veri seti olarak 1000 tane kusuru olmayan ve 1000 tane kusurlu olmak üzere toplam 2000 tane yeşil kahve çekirdeği görüntüsü kullanılmıştır. Görüntülerin öncelikle arka planı temizlenip, daha sonrasında aşırı öğrenme riskinin azaltılabilmesi için dönme ve aynan işlemleri uygulanarak verilerin miktarı artırılmıştır. Daha sonrasında görüntüler 180x180 piksel boyutuna küçültülerek veri setine konvolüsyonel sinir ağı uygulanmıştır. Sonucunda kullanılan model test veri setinde %94,68 doğruluk oranına ulaşmıştır.

Hira Lal Gope ve Hidekazu Fukai [32] normal yeşil kahve çekirdeklerinden farklı boyutta olan ve farklı kavrulma özelliklerine sahip olarak peaberry türünün görüntü işleme ve makine öğrenmesi yöntemleriyle sınıflandırılmasını amaçlamışlardır. Uygulamada toplam 3338 tane kahve çekirdeği görüntüsü kullanılmıştır ve bu görüntüler eğitim,

doğrulama ve test verisi olmak üzere üç gruba ayrılmıştır. Özellikle çıkarıp yöntemlerinden PCA (Principal Component Analysis, Temel Bileşenler Analizi) veri setine uygulanarak görüntüler SVM (Support Vector Machine, Destek Vektör Makinesi) ve CNN uygulanmıştır. CNN'e ait doğruluk oranı %98,5, SVM'e ait doğruluk oranı %96,71 olarak bulunmuştur.

Joao Paulo Lebarck Pizzaia ve arkadaşları [33] şekil, boyut ve renk gibi kriterler açısından iyi ve kötü olan kahve çekirdeklerinin MLP (Multilayer Perceptron, Çok Katmanlı Algılayıcı) sinir ağı kullanılarak sınıflandırılmasını amaçlamışlardır. Makalede 748 örnek eğitim verisi, 100 örnek ise doğrulama verisi olarak kullanılmıştır. Görüntülerde bulunan kahve çekirdekleri arka planlarından ayrıştırılarak görüntüye ait renk, şekil ve boyut gibi özellikler çıkarılmıştır. Daha sonrasında bu özellikler MLP sinir ağının eğitilmesi için kullanılmıştır. Sonuç olarak doğruluk oranı %94,10 olarak bulunmuştur.

Robby Janandi ve Tjeng Wawan Cenggoro [34] derin öğrenme temelli bir model kullanarak mobil telefon kamerası aracılığıyla kahve çekirdeklerinin kalitesini kahve belirleyebilecek bir mobil uygulama geliştirmeyi amaçlamaktadır. Kahve çekirdeklerinin kalitesini sınıflandırmak için veri setinde 160 adet fotoğraf kullanılmıştır. Fotoğraflar, Endonezya Ulusal Standartları'na (SNI) göre kalite sınıflarına (2, 3 ve 4) göre etiketlenmiştir. Veri setinde iki farklı büyüklükte (110 ve 160 görüntü) deneyler yapılmıştır. Veri setinin sınıflandırılmasında derin öğrenme algoritmalarından VGG16 ve ResNet-152 kullanılmıştır. 110 görüntü ile yapılan deneyde, ResNet-152 modelinin doğruluğu %62,3, VGG16 modelinin doğruluğu ise %58,7 olmuştur. 160 görüntü ile yapılan deneyde ise ResNet-152 modelinin doğruluğu %73,3, VGG16 modelinin doğruluğu ise %66,8 olmuştur.

Nen-Fu Huang ve arkadaşları [35] yeşil kahve çekirdeklerini iyi ve kötü olarak iki sınıfa ayırmaya amaçlamışlardır. Bu makalede 1000 adet iyi, 1000 adet kötü kahve çekirdeği olmak üzere toplam 2000 adet kahve çekirdeği görüntü kullanılmıştır. Görüntüler ilk olarak gri tonlamalı hale getirilerek her bir görüntünün arka planı temizlenmiştir. Daha sonrasında veri miktarını arttırmak amacıyla döndürme ve yansıtma işlemleri uygulanarak veriler 9 kat daha arttırılmıştır. Gri tonlamalı kahve çekirdeklerinin sınıflandırılması için CNN kullanılmıştır. Uygulama sonucunda %93,34 doğruluk oranına ulaşılmıştır.

Alifya Febriana ve arkadaşları [12] "USK-COFFEE" adlı bir veri seti oluşturmuşlar ve bu veri setini kullanarak derin öğrenme teknikleri ile 4 farklı yeşil Arabica kahve

çekirdeği türünü sınıflandırmak amacıyla bir çalışma yapmışlardır. Çalışmanın amacı, kahve çekirdeklerinin farklı kalite sınıflarına ayrılmasını sağlayacak bir model geliştirmektir. Bu çalışmada ResNet18 ve MobileNetV2 modelleri kullanılarak doğruluk, hassasiyet, duyarlılık değerleri hem sınıf bazında hem ortalama bazında hesaplanmıştır. Sonuç olarak, ResNet18'e ait doğruluk oranı %81,13 olarak bulunurken, MobileNetV2'ye ait doğruluk oranı 81,31 olarak bulunmuştur.

Bu tez çalışmasında yeşil çekirdeğin dört farklı türüne ait 8000 veriden oluşan USK-Coffee veri seti kullanılmıştır. Yeşil kahve çekirdekleri peaberry, longberry, premium ve defect olmak üzere dört sınıfa ayrılmıştır. USK-Coffee veri setinde ResNet18 ve MobileNetV2 modelleri direkt olarak veri setine uygulanmaktadır. Bu çalışmada farklı olarak veri seti üzerinde ilk olarak VGG16, DenseNet201, MobileNetV2 ve ResNet18 ile özellik çıkarımı yapılarak K-en yakın komşu (KNN), Naive Bayes, karar ağacı ve Destek Vektör Makineleri (SVM) algoritmaları uygulanmış ve performans metrikleri elde edilmiştir. Bu algoritmalar uygulanırken ayrıntılı bir değerlendirme olması açısından veri setine K-katmanlı çapraz doğrulama (K-fold cross validation) yöntemi uygulanarak k değeri 5 olarak seçilmiştir. Sonraki aşamada, VGG16, DenseNet201, MobileNetV2 ve ResNet18 ile çıkan özelliklere Temel Bileşenler Analizi (PCA) ve Yönlü Gradyanlar Histogramı (HoG) gibi ön işleme yöntemleri uygulanmış ve bu yöntemlerin ardından makine öğrenmesi algoritmaları yeniden çalıştırılmıştır. Elde edilen sonuçlar, algoritmaların veri seti üzerindeki performansını ve doğruluk oranlarını değerlendirmek ve karşılaştırmak amacıyla analiz edilmiştir. Bu çalışmadaki amaç hem USK-Coffee veri setine farklı yöntemler uygulayarak performans metriklerinde elde edilen sonuçları yükseltmek, hem de PCA ve HoG gibi farklı yöntemleri de kullanarak veri setinin performans metrikleri üzerinde nasıl bir etkisi olduğunu gözlemlemektir.

2. VERİ SETİ

Bu tezde, Alifya Febriana ve arkadaşları [12] tarafından üretilen USK-Coffee veri seti kullanılmıştır. Veri setinde bulunan kahve çekirdekleri 2021 yılında Endonezya, Banda Aceh'deki kahve çiftçilerinden doğrudan temin edilmiştir. Çiftçiler, kahve çekirdeklerini peaberry, longberry, premium ve defect olmak üzere dört kategoriye ayrılmasında yardımcı olmuştur. Ham veri Batı Endonezya saat dilimine göre saat 09,00 – 12,00 aralığında kaydedilmiştir. Her kahve çekirdeğinin ön ve arka tarafları beyaz bir A4 kâğıdı üzerinde fotoğraflanmıştır. Görüntüler, Canon EOS 60D ile ISO 800 ayarında, 1/80 saniye pozlama süresi, f/5,6 diyafram açıklığı ve otomatik modda netleme ile çekilmiştir. Kamera A4 kâğıdın 1 metre yukarısına yerleştirilmiştir ve kahve çekirdeklerinin renginin daha net ve keskin göstermek için çekirdeklerin ön ve arkasına bir aydınlatma cihazı yerleştirilmiştir [12].

Veri setinde toplamda 8000 görüntü bulunmaktadır. Bu görüntüler peaberry, longberry, premium ve defect olmak üzere 4 farklı yeşil Arabica kahve çekirdeği türüne göre sınıflandırılmıştır. Bu da her sınıf için 2000 görüntü bulunduğu anlamına gelmektedir. Her kahve çekirdeğinin görüntüsü 256x256 piksel boyutundadır.

Longberry kahve çekirdekleri yalnızca Endonezya'nın Açe bölgesine bağlı Takengon kasabasında bulunmaktadır. Bu çekirdekler uzun ve büyük bir şekle sahiptirler. Longberry kahve çekirdekleri diğer yeşil kahve çekirdeklerine kıyasla daha açık bir renge sahiptir [12]. Longberry kahve çekirdeklerine ait görüntü örnekleri Şekil 2.1'deki gibidir.



Şekil 2.1 Longberry Kahve Çekirdeği Görüntüleri

Peaberry kahve çekirdekleri, diğer kahve çekirdeklerine göre daha küçük, yuvarlak ve yoğun yapıdadır. Bu çekirdekler, hasat anında ağaçta yalnızca %3 – %5 oranında bulunması nedeniyle benzersiz ve çok değerli bir kahvedir. Hasat edilen her 200 kg kahve çekirdeğinde yalnızca 5 kg peaberry kahve çekirdeği elde edilmektedir [12]. Peaberry kahve çekirdeklerine ait görüntü örnekleri Şekil 2.2'deki gibidir.



Şekil 2.2 Peaberry Kahve Çekirdeği Görüntüleri

Premium kahve çekirdekleri daha yuvarlak ve daha büyük bir şekle sahiptir. Bu çekirdekler genellikle "yarı yıkanmış" kahve çekirdekleri olarak bilinir. Endonezya'nın Sumatra adasındaki kahve çiftçileri tarafından en yaygın kullanılan yöntem, hasat sonrası yarı yıkama işlemidir [12]. Premium kahve çekirdeklerine ait görüntü örnekleri **Şekil 2.3**'teki gibidir.



Şekil 2.3 Premium Kahve Çekirdeği Görüntüleri

Defect kahve çekirdekleri, yanlış işleme veya hayvanlar ya da böcekler tarafından yenilme sonucunda oluşur. Hatalı kahve çekirdekleri şekillerinden tanınabilir; bütün değildir, kırık görünür ve dengesiz ve uygun olmayan boyutta olduğu için içi boş gibi durur [12]. Defect kahve çekirdeklerine ait görüntü örnekleri **Şekil 2.4**'teki gibidir.



Şekil 2.4 Defect Kahve Çekirdeği Görüntüleri

3. YÖNTEM

Bu tezde veri seti üzerinde 3 farklı uygulama yapılmıştır.

Uygulama-1’de veri setine CNN algoritmaları ile özellik çıkarımı yapılarak makine öğrenmesi algoritmaları uygulanmıştır. Uygulama 1’in blok şeması Şekil 3.1’de gösterildiği gibidir.



Şekil 3.1 Uygulama-1 Blok Şeması

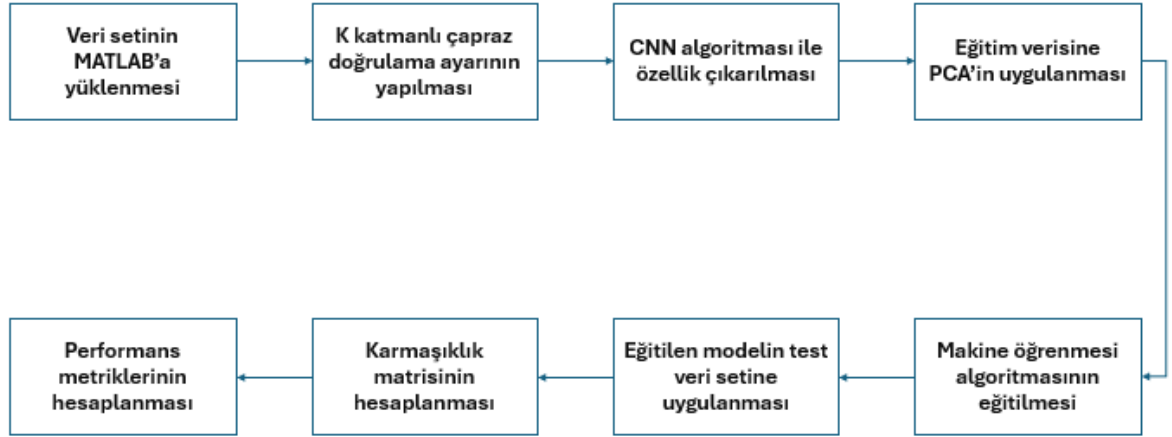
Kodun ilk kısmında veri setinin bulunduğu klasör tanımlanmıştır. Bu klasörün altında 4 sınıfa ait alt klasörler de bulunmaktadır. “imageDatastore” fonksiyonu kullanılarak veri setinde bulunan görüntülerin alt klasör adlarını etiket olarak kullanması sağlanmıştır. Daha sonra “cvpartition” fonksiyonu kullanılarak veri seti 5 katmanlı çapraz doğrulama yöntemi kullanılır. Daha sonrasında her katman için ortalama performans metriklerinin saklanacağı 5x1 boyutunda boş matrisler tanımlanır.

Konvolüsyonel sinir ağıyla özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyona ilk olarak önceden eğitilmiş konvolüsyonel sinir ağı modeli yüklenir. Daha sonrasında “activation” komutu kullanılarak sinir ağı modelinin son katmanlarından aktivasyonlar çıkartılır. Bu aktivasyonlar, modelin eğitimi için kullanılacak olan özellikleri oluşturmaktadır.

Daha sonrasında, çıkarılan özellikler doğrultusunda kullanılacak olan makine öğrenmesi algoritması eğitilir ve test verileri üzerinde tahminler yapılmaktadır. Modelin performansını ölçmek için “confusionmat” fonksiyonu kullanılarak karmaşıklık matrisi

bulunmaktadır. Karmaşıklık matrisinin bulunmasıyla doğruluk, hassasiyet, duyarlılık ve F1 skoru gibi performans metrikleri her sınıf için hesaplanmış ve ayrıca karmaşıklık matrisinden türetilen genel ortalama değerler elde edilmiştir.

Uygulama-2’de veri setine CNN algoritmaları ve PCA yöntemi ile özellik çıkarımı yapılarak makine öğrenmesi algoritmaları uygulanmıştır. Uygulama 2’nin blok şeması Şekil 3.2’de gösterildiği gibidir.



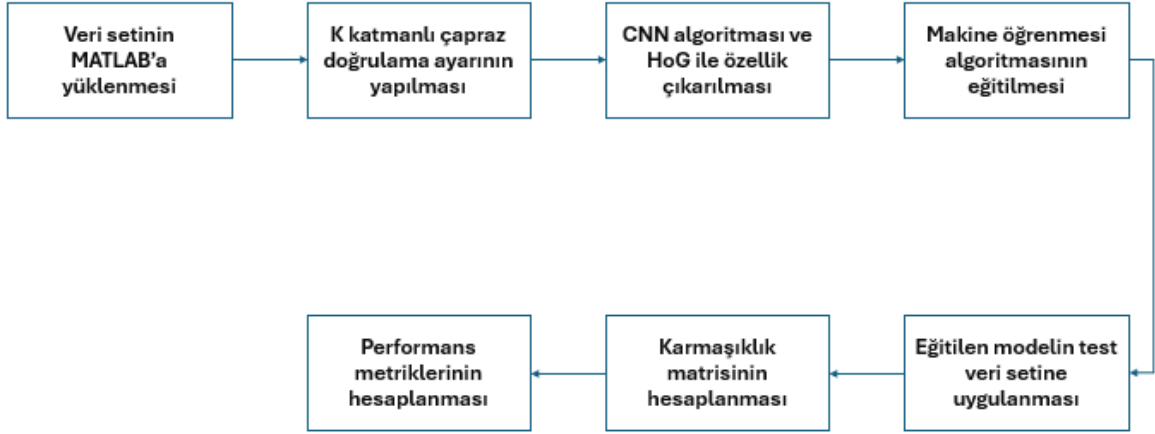
Şekil 3.2 Uygulama-2 Blok Şeması

Uygulama-2’de, Uygulama 1’de olduğu gibi veri setinin bulunduğu klasör tanımlanarak 5 katmanlı çapraz doğrulama yöntemi uygulanmaktadır. Daha sonra konvolüsyonel sinir ağıyla özellik çıkarımı yapmak için bir fonksiyon oluşturulur.

PCA’nın bir veri setine uygulanabilmesi için ilk adımda veri setinde bulunan özneliklerin standartlaştırılması gerekmektedir. İlk olarak veriler standartlaştırılarak her özelliğin ortalaması 0 ve standart sapması 1 olur. İkinci adımda standartlaştırılan eğitim verilerine MATLAB’da bulunan “pca” fonksiyonu uygulanır. Bu fonksiyon, bu standartlaştırılmış veri matrisinden kovaryans matrisini hesaplar ve bu matris yardımıyla özvektörler ve özdeğerler hesaplanmış olur. Özvektörler ve özdeğerler matrislere özdeğerler büyükten küçüğü doğru sıralı olacak şekilde yazdırılır. Bu sıralama sayesinde, ilk özvektör (en büyük özdeğere sahip olan) verinin en fazla varyansını açıklayan yönü temsil eder. Daha sonra belirli bir varyans yüzdesini belirten bir değişken tanımlanır. Daha sonra belirlenen bu varyans değerine göre kullanılacak ana bileşene sayısı belirlenir.

Daha sonrasında, karmaşıklık matrisi bulunarak doğruluk, hassasiyet, duyarlılık ve F1 skoru gibi performans metrikleri her sınıf için ve ortalama olarak hesaplanmıştır.

Uygulama-3'te veri setine CNN algoritmaları ve HoG yöntemi ile özellik çıkarımı yapılarak makine öğrenmesi algoritmaları uygulanmıştır. Uygulama 3'ÜN blok şeması Şekil 3.3'te gösterildiği gibidir.



Şekil 3.3 Uygulama-3 Blok Şeması

Uygulama-3'te, diğer uygulamalarda olduğu gibi veri setinin bulunduğu klasör tanımlanarak 5 katmanlı çapraz doğrulama yöntemi uygulanmaktadır. Daha sonra konvolüsyonel sinir ağıyla özellik çıkarımı yapmak için bir fonksiyon oluşturulur.

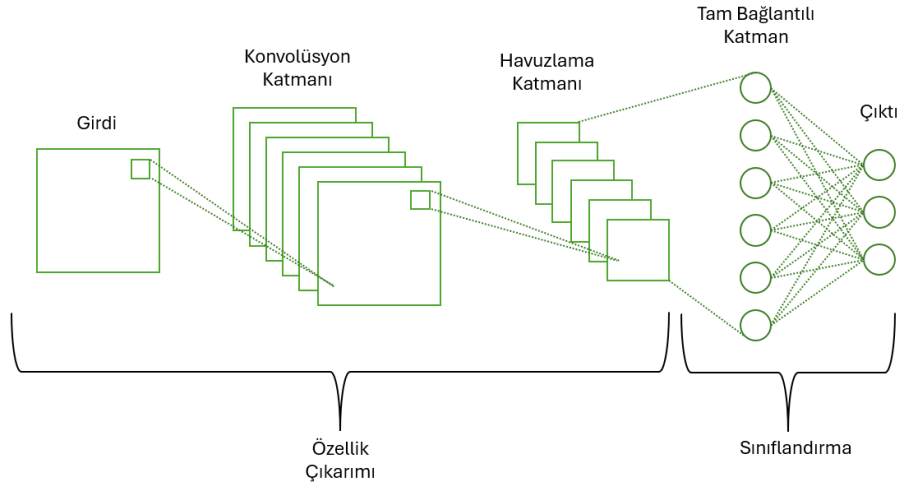
Daha sonra “extractHOGFeatures” komutu kullanılarak görüntülerin HoG özellikleri çıkartılır. Her görüntü 8x8 piksellik hücelere bölünerek her hücredeki gradyan büyüklükleri ve yönleri hesaplanmaktadır. Bu işlemde bir özellik haritası oluşturulmuş olur. Bu hücrelerdeki bilgileri birleştirilerek, 2x2 boyutundaki bloklar halinde gruplanmaktadır ve bu bloklardaki her hücrenin gradyan özellikleri normalleştirilir. Bu bloklar sayesinde HoG histogramları oluşturulur. Daha sonra uygulanacak olan CNN algoritmasıyla birleştirilebilmesi ve makine öğrenmesi algoritmalarının kullanabileceği bir formatta olabilmesi adına histogramlar tek bir vektör haline getirilir.

Daha sonrasında, karmaşıklık matrisi bulunarak doğruluk, hassasiyet, duyarlılık ve F1 skoru gibi performans metrikleri her sınıf için ve ortalama olarak hesaplanmıştır.

3.1. Derin Öğrenme Modelleri

3.1.1. Konvolüsyonel Sinir Ağları (CNN)

Kunihiko Fukushima tarafında 1979 yılında temel konvolüsyonel sinir ağları mimarisi geliştirilmiştir. Neocognitron adı verilen bu yapı konvolüsyonel ve aşağı örnekleme katmanlarına sahip bir yapıdır [39]. 1990'larda, LeCun ve ekibi CNN'lere gradyan tabanlı öğrenme algoritmasını uygulayarak el yazısıyla yazılmış rakam sınıflandırma problemlerinde başarılı sonuçlar elde etmiştir [37]. Bu gelişmelerden sonra, araştırmacılar CNN'leri daha da geliştirerek günümüzde birçok uygulamada kullanılan bir yapı haline getirmiştir.



Şekil 3.4 Basit Konvolüsyonel Sinir Ağı (CNN) Mimarisi

Şekil 3.4'te gözüktüğü gibi CNN mimarisi iki öznelik çıkarımı ve sınıflandırma olmak üzere iki ana başlıktan oluşmaktadır. Bu yapı konvolüsyon katmanı, havuzlama katmanı ve tam bağlantılı katman olmak üzere 3 tip katmanı içermektedir.

Girdiler ilk başta konvolüsyon katmanında özellik çıkarım işlemine tabi tutulur. Konvolüsyon katmanı, modelin girişten özellikleri çıkardığı katmandır. Bu katmanda, özellik haritaları daha önceki katmanlardan alınan çıktılarla öğrenilebilir çekirdekler (kernel) aracılığıyla konvolüsyon işlemi uygulanarak hesaplanır. Bu işlem, her bir özellik haritasını farklı çekirdeklerle kaydırarak özellikleri çıkarmayı sağlar [37]. Genel olarak bu ifade aşağıdaki formül ile ifade edilmektedir.

$$x_j^l = f \left(\sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_j^l \right) \quad (3.1)$$

- x_j^l : Katmandaki çıktı özellik haritası
- x_i^{l-1} : Bir önceki katmandan gelen giriş özellik haritası
- k_{ij}^l : Katmanda kullanılan çekirdek matrisi
- b_j^l : Çıktı için eklenen bias terimi
- M_j : Girdi haritaları kümesi
- f : Aktivasyon fonksiyonu

Çıkış haritaları ağı doğrusal olmayan ilişkileri öğrenmesini sağlamak için ReLu, Softmax, hiperbolik tanjant gibi bir aktivasyon fonksiyonundan geçerler.

Havuzlama katmanı, boyut azaltma işlemini gerçekleştirir. Bu katman, öğrenme süreci olmaksızın hesaplama maliyetlerinin düşük tutulmasına ve daha dayanıklı bir model elde etmeye yardımcı olur [40]. Bu katmanda giriş ve çıkış haritalarının sayısı değişmez. Veri boyutu küçültülerek hesaplama karmaşıklığı azaltılır. Havuzlama katmanı, her bir özellik haritasını sabit boyutlu bölgelere ayırarak her bölgede maksimum havuzlama veya ortalama havuzlama gibi işlemler yapar.

Maksimum havuzlama her bölgedeki maksimum değeri çıktı olarak seçerken diğer bilgileri göz ardı etmektedir. Ortalama havuzlama her bölgedeki ortalama değeri çıktı olarak alır [41]. Maksimum ve ortalama havuzlama hesaplamaları sırasıyla aşağıdaki formüller kullanılarak yapılmaktadır.

$$a_{ij} = \frac{1}{|R_{ij}|} \sum_{pq \in R_{ij}} a_{kpq} \quad (3.2)$$

$$a_{ij} = \max_{pq \in R_{ij}} (a_{kpq}) \quad (3.3)$$

- a_{ij} : ortalama veya maksimum değerle elde edilen çıkış değeridir.
- R_{ij} : havuzlama (pooling) işlemi yapılan bölgedir.
- a_{kpq} : R_{ij} bölgesindeki her bir değeri temsil eder.

Havuzlama katmanından tam bağlantılı katmana gelen özellikler kullanılarak her bir sınıf için hesaplama yapılır. Özellik haritaları bu katmanda vektörlere dönüştürülerek sınıflandırma yapılması için kullanılmaktadır. Bu özellik vektörü, Softmax gibi bir

aktivasyon fonksiyonu aracılığıyla her sınıf için bir olasılık değerine dönüştürülmektedir [37].

3.1.1.1. VGG16

Bir konvolüsyonel sinir ağı modeli olan VGGNet, ilk olarak NK. Simonyan ve A. Zisserman tarafından 2015 yılında yayımlanan “Very Deep Convolutional Networks For Large-Scale Image Recognition” makalesinde tanıtılmıştır [42]. VGGNet Oxford Üniversitesi'nin Görsel Geometri Grubu'na (VGG) ait bir modeldir. VGG16 ve VGG19 adından 2 türü bulunmaktadır. Diğer modellerden farkı bu modellerde büyük boyutlu konvolüsyon çekirdekleri yerine birden fazla 3x3 konvolüsyon çekirdeğinin kullanılıyor olmasıdır [43].

VGG16 adından da anlaşılacağı gibi 13 konvolüsyon katmanı ve 3 tam bağlantılı katmandan oluşan toplam 16 katmanlı bir VGGNet türüdür. Konvolüsyon çekirdeği boyutu 3x3'tür ve adım boyutu 1'dir. Beş adet maksimum havuzlama katmanı bulunmakta olup, boyutu 2x2 ve adım boyutu 2'dir. Üç adet tam bağlantı katmanı vardır, ilk iki katmanda 4096 kanal, üçüncü katmanda ise 1000 kanal bulunmaktadır ve bu kanallar 1000 etiket kategorisini temsil eder. Son katman ise Softmax katmanıdır. Softmax, veri setindeki her sınıfa ait olasılığı sonuç olarak kullanan bir aktivasyon fonksiyonudur. Softmax fonksiyonuna ait formül aşağıdaki gibidir.

$$Softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1} e^{z_j}} \quad (3.4)$$

- z_i : i. sınıfa ait skor
- $\sum_{j=1} e^{z_j}$: tüm sınıfların üssü alınmış haldeki skor toplamı

Tüm gizli katmanlar, ReLU doğrusal olmayan aktivasyon fonksiyonu ile takip edilmektedir. ReLU fonksiyonu, giriş değeri eğer pozitifse aynı şekilde iletir. Giriş değeri negatifse sıfıra dönüştürür. ReLU fonksiyonu matematiksel olarak aşağıdaki şekilde ifade edilmektedir.

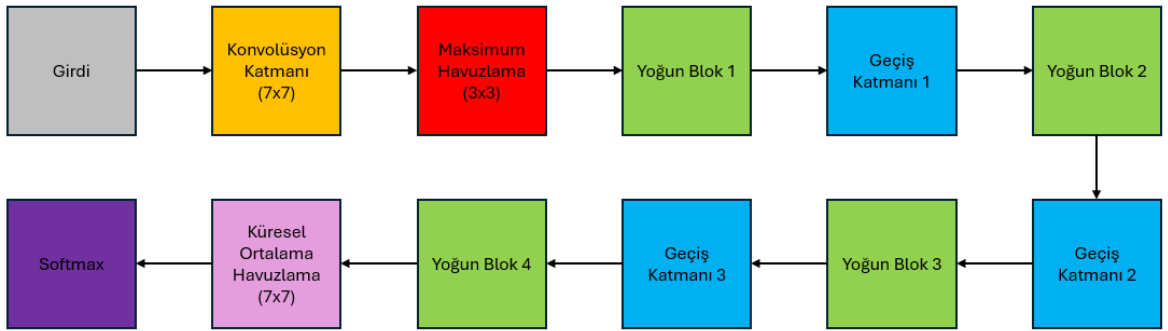
$$f(x) = \max(0, x) \quad (3.5)$$

Burada x giriş değerini, $f(x)$ ise çıkış değerini ifade etmektedir.

uygulanmaktadır. Yoğun bloklar arasında bulunan geçiş katmanlarında ise 1x1 konvolüsyon ve ortalama havuzlama kullanılmaktadır.

DenseNet201 modelinde 224x224 piksel görüntüler kullanılmaktadır. Girdiler ilk başta 7x7 konvolüsyon katmanına, daha sonra maksimum havuzlama işleminde geçerler. Daha sonrasında 4 yoğun blok ve yoğun blokların arasında bulunan 3 geçiş katmanından geçerek küresel ortalama havuzlama katmanına ulaşırlar. Daha sonrasında da Softmax katmanında, veri setindeki her sınıfa ait olasılığı sonuç olarak kullanılmaktadır.

Tam bağlantılı katmanlar, çok fazla parametreye sahip oldukları için bellek ve işlem gücünü oldukça tüketir. Küresel ortalama havuzlama katmanı, tam bağlantılı katmanların yerine kullanılarak parametre sayısını önemli ölçüde azaltır. Bu, belleği daha verimli kullanmayı sağlamaktadır [46].



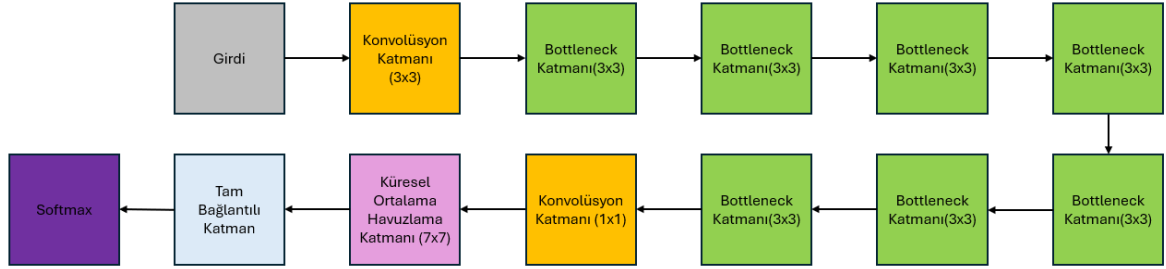
Şekil 3.6 DenseNet201 Mimarisi

Bu tezdeki uygulamalarda kullanılan DenseNet201 modeli, ImageNet veri seti üzerinde SGD + momentum optimizasyon algoritması ile eğitilmiştir. Modelin başlangıç öğrenme oranı 0,1 olarak belirlenmiş ve epoch ilerledikçe azaltılmıştır. Eğitim süreci toplam 90 epoch olup, mini yığın (batch size) boyutu 32 olarak ayarlanmıştır. Kayıp fonksiyonu olarak çapraz-entropi (cross-entropy) kullanılmış ve modelin parametre güncellenmesi için momentum değeri 0,9 olarak belirlenmiştir. Ağırlıkların aşırı büyümesini önlemek için 0,0001 ağırlık bozulması (weight decay) uygulanmıştır. Ayrıca, modelin her katmanında 32 büyüme oranı (growth rate) kullanılmıştır.

3.1.1.3. MobileNetV2

MobileNetV2 mimarisi, başlangıçta 32 filtreli tam evrişimli bir katman ile başlar ve 19 adet residual dar boğaz (bottleneck) bloğu içerir. Aktivasyon fonksiyonu olarak ReLU6 kullanılmıştır. Bu, düşük hassasiyetli hesaplamalarda daha dayanıklı olduğu için tercih

edilmiştir. Ağ, tüm evrişim işlemlerinde 3x3 çekirdek boyutu kullanır ve eğitim sırasında düğüm bırakma (dropout) ve toplu normalizasyon (batch normalization) uygular. İlk katman dışında, ağ boyunca genişleme oranı (expansion rate) sabit kalır [14]. MobileNetV2 modelinde girdi olarak 224x224 boyutunda görüntüler kullanılmaktadır. MobileNetV2 modeline ait mimari aşağıdaki gibidir.



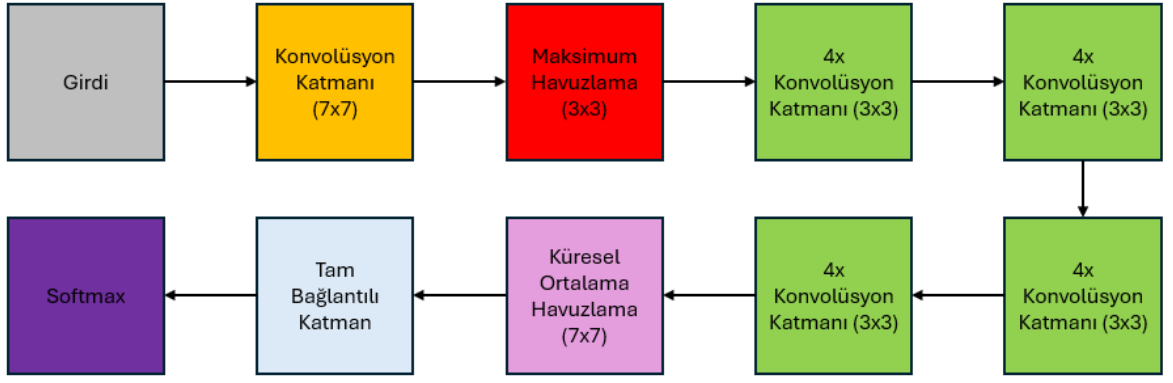
Şekil 3.7 MobileNetV2 Mimarisi

Bu tezdeki uygulamalarda kullanılan MobileNetV2 modeli, ImageNet veri seti üzerinde RMSprop optimizasyon algoritması ile eğitilmiştir. Modelin başlangıç öğrenme oranı 0,045 olarak belirlenmiş ve epoch ilerledikçe 10 kat azaltılmıştır. Eğitim süresi toplam 200 epoch olup, her iterasyonda 96 mini yığın (batch size) kullanılmıştır. Kayıp fonksiyonu olarak çapraz-entropi (cross-entropy) tercih edilmiş ve modelin parametre güncellenmesi için momentum değeri 0,9 olarak ayarlanmıştır. Ağırlıkların aşırı büyümesini engellemek için 0,00004 ağırlık bozulması (weight decay) uygulanmıştır. Ayrıca, modelin verimli çalışmasını sağlamak için 1,0 genişlik çarpanı (width multiplier) kullanılmıştır [14].

3.1.1.4. ResNet18

Türkçe karşılığı “Artık Sinir Ağı” olan ResNet katmanların girişlerine referansla artık fonksiyonları (residual functions) öğrenmesini sağlayan bir derin öğrenme mimarisidir. 2015’te geliştirilen bu model geliştirildiği yılda ImageNet yarışmasını kazanmıştır.

ResNet18 modeli, 16 konvolüsyon katmanı, 2 aşağı örnekleme katmanı ve bazı tam bağlı katmanlardan oluşmaktadır. ResNet18’in giriş görüntü boyutu 224×224 olup, ilk konvolüsyon katmanı dışında tüm konvolüsyon katmanlarının çekirdek boyutu 3×3 ’tür. İlk katmanda 7×7 çekirdek kullanılmaktadır. Son evrişim katmanının çıktısı özelliği ortalama havuzlama işlemine tabi tutulduktan sonra, tam bağlantılı katman ile bir özellik vektörü elde edilir. Daha sonra Softmax normalizasyonu ile sınıflandırma olasılıkları hesaplanır [36]. ResNet18’e ait model aşağıdaki gibidir.



Şekil 3.8 ResNet18 Mimarisi

Bu tezdeki uygulamalarda kullanılan ResNet18 modeli, ImageNet veri seti üzerinde SGD + Momentum optimizasyon algoritması ile eğitilmiştir. Modelin başlangıç öğrenme oranı 0,1 olarak belirlenmiş ve 30, ve 60, epoch'ta 10 kat azaltılarak öğrenme sürecinin optimize edilmesi sağlanmıştır. Eğitim süresi toplam 90 epoch olup, 256 mini yığın (batch size) kullanılmıştır. Kayıp fonksiyonu olarak çapraz-entropi (cross-Entropy) kullanılmış ve modelin parametre güncellenmesi için momentum değeri 0,9 olarak ayarlanmıştır. Ayrıca, ağırlıkların aşırı büyümesini önlemek için 0,0001 ağırlık bozulması (weight decay) uygulanmıştır [13].

3.2. Temel Bileşenler Analizi (PCA)

PCA, gözlemlerin birbiriyle ilişkili birkaç niceliksel bağımlı değişken tarafından tanımlandığı bir veri tablosunu analiz eden çok değişkenli bir tekniktir. Amacı, tablodan önemli bilgileri çıkarmak, onu temel bileşenler adı verilen yeni ortogonal değişkenler kümesi olarak temsil etmek ve gözlemlerin ve değişkenlerin benzerlik modelini haritalarda noktalar olarak görüntülemektir. PCA'nın kullanımındaki amaç veri tablosundan en önemli bilgiyi çıkarmak, yalnızca bu önemli bilgiyi tutarak veri kümesinin boyutunu sıkıştırmak, veri kümesinin açıklamasını basitleştirmek ve gözlemlerin ve değişkenlerin yapısını analiz etmektir [22].

Temel bileşen analizi (PCA), aynı konularda gözlemlenen çeşitli değişkenlerden gelen bilgileri, temel bileşenler adı verilen daha az değişkende birleştiren çok değişkenli bir istatistiksel yöntemdir. Bilgi, orijinal değişkenlerin toplam varyansı ile ölçülür ve temel bileşenler bu varyansın büyük kısmını en iyi şekilde açıklamaktadır. Temel bileşenler,

karmaşık, çok değişkenli bir veri kümesinde bulunan ana özelliklerin sezgisel ve yapılandırılmış bir şekilde yorumlanmasına olanak tanıyan geometrik özelliklere sahiptir [23].

PCA'nın bir veri setine uygulanabilmesi için ilk adımda veri setinde bulunan özneliklerin standartlaştırılması gerekmektedir. Bu işlem özneliklerin ortalamasının 0'a, standart sapmalarının ise 1'e eşit olacak şekilde dönüştürülmesi anlamına gelmektedir. Bu işlemin yapılabilmesi için ilk olarak aşağıdaki formül kullanılarak verilerin ortalaması hesaplanır.

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i \quad (3.6)$$

Daha sonra özneliklerin aşağıdaki formül yardımıyla standart sapmalarının hesaplanması gerekmektedir.

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \mu)^2} \quad (3.7)$$

Verilerin standartlaştırılması işleminde öznelikler ilk olarak ortalama değerden çıkarılır. Daha sonrasında bu değerler standart sapmaya bölünür.

$$X'_i = \frac{X_i - \mu}{\sigma} \quad (3.8)$$

Sonraki adımda kovaryans matrisinin hesaplanması gerekmektedir. Kovaryans birden fazla boyuta sahip veri setlerinde iki öznelik arasındaki ilişkinin hesaplanmasını sağlar. Kovaryansın pozitif olması özneliklerin birbirleriyle doğru orantılı olarak arttığı veya azaldığı anlamına gelmektedir. Kovaryansın negatif olması ise özneliklerin birbirleriyle test orantılı olarak arttığı veya azaldığı anlamına gelmektedir. Kovaryans aşağıdaki formül ile hesaplanmaktadır.

$$cov(X, Y) = \frac{\sum_{i=1}^n (X_i - \mu_X)(Y_i - \mu_Y)}{n-1} \quad (3.9)$$

Yukarıdaki formülde μ_X ve μ_Y sırasıyla X ve Y'nin ortalama değerlerini belirtmek için kullanılan sembollerdir.

Kovaryans formülü yardımıyla ikiden fazla öznitelik için simetrik bir kovaryans matrisi oluşturularak kovaryans matrisinin özvektör ve özdeğerler hesaplanmaktadır. Bir kare matris için, özvektör ve özdeğer aşağıdaki formüller yardımıyla bulunmaktadır:

$$Cv = \lambda v \quad (3.10)$$

$$\det(C - \lambda I) = 0 \quad (3.11)$$

- C: Kovaryans matrisi
- v : Özvektör
- λ : Özdeğer
- I: Birim matrisi

Bulunan özvektörler, özdeğerlerine göre büyükten küçüğe doğru sıralanmaktadır. Bu olay temel bileşenlerin önem sıralamasını göstermektedir. Bu olay sonrasında özdeğeri diğerlerinden daha az olan bileşenlerin kullanımından vazgeçilip özdeğeri daha çok olan bileşenlerin kullanımına karar verilmektedir. Daha sonra seçilen özvektörlerden bir özellik vektörü oluşturulması gerekmektedir. Son olarak oluşturulan özellik vektörünün transpozuyla orijinal veri setinin çarpılmasıyla PCA uygulanmış yeni veri seti elde edilmektedir.

3.3. Yönlü Gradyanlar Histogramı (HoG)

HoG (Histogram of Oriented Gradients, Yönlendirilmiş Gradyanların Histogramı) genellikle bilgisayarlı görü ve görüntü işlemede verileri sınıflandırma işlemlerinde kullanılan bir öznitelik çıkarma yöntemidir [24]. Bu tekniğin mantığı görüntülerin gradyanlarını yoğunluk ve yön histogramlarına dönüştürerek ilgili işlemler için kullanmaktır.

$$I_x(r, c) = I(r, c + 1) - I(r, c - 1) \quad (3.12)$$

$$I_y(r, c) = I(r - 1, c) - I(r + 1, c) \quad (3.13)$$

Yukarıdaki formüller, bir görüntü üzerindeki x-yönündeki (yatay) ve y-yönündeki gradyanlarını hesaplamak için kullanılır. $I(r, c)$, görüntünün (r, c) konumundaki pikselin gri tonlamalı değerini ifade eder. Bu yöntemde ilk olarak görüntüdeki her piksel için yatay ve

dikey yöndeki gradyan değerlerinin hesaplanması gerekmektedir. Her piksel için gradyan büyüklüğü ve yönü sırasıyla aşağıdaki formüller yardımıyla hesaplanmaktadır.

$$\mu = \sqrt{I_x^2 + I_y^2} \quad (3.14)$$

$$\theta = \tan^{-1}(I_y, I_x) \quad \text{mod } \pi \quad (3.15)$$

Bu sayede gradyan yönleri ve büyüklüklerini içeren bir özellik haritası oluşturulmuş olur. Daha sonra görüntü, küçük hürelere bölünmektedir. Her hücre C x C boyutundadır. Örneğin 8x8 piksel boyutunda hücreler kullanılabilir. Her hücrede gradiyent yönelimleri, belirli bir sayıdaki bin'e bölünür. Her bin, belirli bir yönetime karşılık gelir ve genellikle 0 ile 180 derece arasındadır. HoG, her hücrede gradyan yönelimlerinin histogramını oluşturur. Eğer hücre boyutu CxC piksel ve histogramda B bin varsa, her pikselin gradyan yönelimi iki komşu bin arasına dağıtılır. Her pikselle ilgili histogram hesaplaması aşağıdaki formüller yardımıyla yapılmaktadır. Bu aşamada hücrelerin gradyan bilgileri bir araya getirilip, her bir hücreyi birer özellik haritası olarak temsil etmektedir.

$$v_j = \mu \times \frac{c_{j+l} - \theta}{w} \quad (3.16)$$

$$v_{j+l} = \mu \times \frac{\theta - c_j}{w} \quad (3.17)$$

- v_j ve v_{j+l} : histogram binlerine katkı
- c_j ve c_{j+l} : histogram binlerinin merkezleri
- w : bin genişliği

Daha sonra hücreler, örtüşen bloklar oluşturacak şekilde gruplandırılır. Örneğin 2 x 2 hücrelik bloklar oluşturulur. Blok boyutu genellikle 2C x 2C pikseldir. Her blok, 2 hücre yüksekliğinde ve 2 hücre genişliğindedir. Her blok, hücre histogramlarının birleştirilmesiyle oluşturulan bir özellik vektörü b ile temsil edilir ve aşağıdaki formül yardımıyla hesaplanmaktadır.

$$b = \frac{b}{\sqrt{\|b\|^2 + \epsilon}} \quad (3.18)$$

- $\|b\|^2$: blok vektörünün Euclidean normunun karesi
- ϵ : sifra bölünmeyi önlemek için küçük bir pozitif sabittir

Son olarak HoG özellik vektörünün oluşturulması için bloklardaki normalleştirilmiş hücre histogramları birleştirilir. Toplam özellik vektörü, tüm blok histogramlarının birleştirilmiş halidir ve aşağıdaki formül yardımıyla hesaplanmaktadır.

$$h = \frac{h}{\sqrt{\|h\|^2 + \epsilon}} \quad (3.19)$$

- $\|h\|^2$: tüm özellik vektörünün Euclidean normunun karesi
- ϵ : sifıra bölünmeyi önlemek için küçük bir pozitif sabittir.

3.4. Makine Öğrenmesi Modelleri

3.4.1. K-En Yakın Komşu (KNN) Algoritması

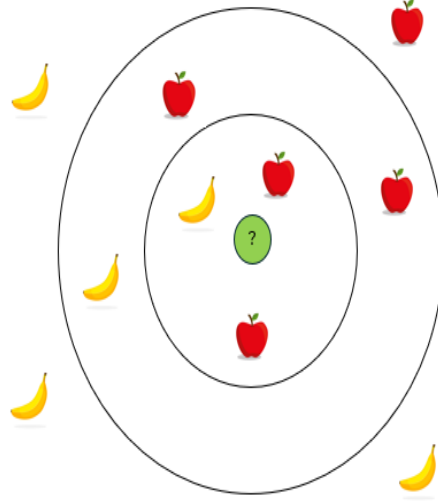
KNN algoritması, yaygın olarak kullanılan sınıflandırma tekniğidir. Yorumlama kolaylığı ve hesaplama süresinin düşük olması nedeniyle yaygın olarak kullanılmaktadır [15].

KNN algoritmasında, bilinmeyen bir örnek, bilinen, eğitilmiş veya etiketlenmiş örneklerle olan benzerliğine göre sınıflandırılmaktadır. Bu benzerlik, bilinmeyen örnek ile tüm etiketlenmiş örnekler arasındaki mesafeler hesaplanarak elde edilmektedir. KNN algoritmasında genel olarak örnekler arası mesafeleri hesaplamada eşitlik 3,1'de verilen Öklid uzaklığı formülü kullanılmaktadır. Sonrasında, sınıflandırma için en yakın komşu örnekler seçilir ve bilinmeyen örnek, en yakın komşu örnekler arasında en çok hangi sınıf varsa o sınıfa atanır [16].

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2} \quad (3.20)$$

- n: noktaların boyut sayısını,
- $d(x_i, x_j)$: iki örnek arasındaki mesafeyi,
- x_{ik} ve x_{jk} : sırasıyla x_i ve x_j noktalarının k bileşenini ifade etmektedir.

KNN algoritmasına ait bir örnek Şekil 3.9'daki gibidir. Örneğin k değeri 3 olarak seçilirse bilinmeyen örneğe en yakın 3 komşusu Öklid uzaklık formülü kullanılarak bulunur. Daha sonrasında bu komşuların çoğunluğuna göre bilinmeyen örneğin muz veya elma olduğuna karar verilir.



Şekil 3.9 KNN Algoritması Örneği

3.4.2. Naive Bayes Algoritması

Naive Bayes algoritması, Thomas Bayes'in ölümünden sonra geliştirilen Bayes Teoremi'ne dayanan bir olasılıksal sınıflandırma mekanizmasıdır. Sınıflandırmanın temel amacı, yeni bir veri parçası ile belirli bir problem alanındaki sınıflandırmalar arasında en iyi eşleştirmeyi bulmaktır [17].

Algoritma tüm değişkenlerin sınıf değişkeninin değerini göz önünde bulundurarak bağımsız olduğunu varsayar. Bu koşullu bağımsızlık varsayımı gerçek dünya uygulamalarında nadiren geçerli olduğundan, algoritma "Naive" olarak nitelendirilir. Naive Bayes algoritması birçok kontrollü sınıflandırma problemi üzerinde hızla öğrenmeye eğilimlidir [18].

Naive Bayes algoritması, bir örneğin hangi sınıfa ait olduğunu tahmin etmek için Bayes teoremini kullanmaktadır. Bayes teoreminde, bir olayın olasılığı, bu olayla ilgili diğer bilgilerin olasılıkları ile ilişkilendirilir ve aşağıdaki formül kullanılarak hesaplanmaktadır. Bu sayede bir verinin belirli bir sınıfa ait olma olasılığı hesaplanmaktadır.

$$P(A|B) = \frac{P(A) * P(B|A)}{P(B)} \quad (3.21)$$

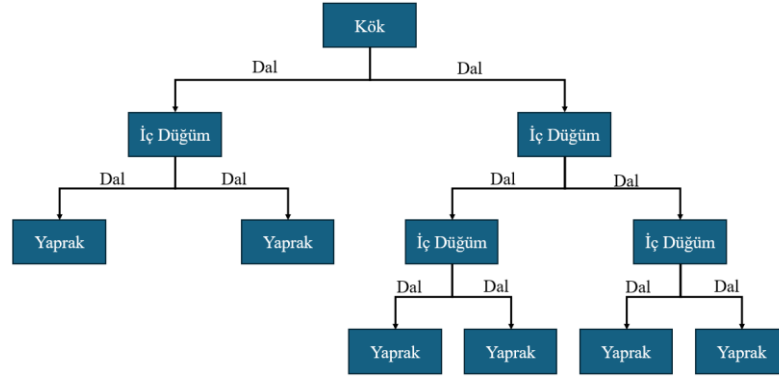
- $P(A|B)$: B olayı meydana geldiğinde A olayının meydana gelme olasılığı,
- $P(A)$: A olayının meydana gelme olasılığı,
- $P(B|A)$: A olayı meydana geldiğinde, B olayının meydana gelme olasılığı,

- $P(B)$: B olayının meydana gelme olasılığı olarak tanımlanmaktadır.

3.4.3. Karar Ağacı Algoritması

Karar ağacı algoritması, 20. yüzyılın sonlarında dijital devrelere geçiş sürecinde elektronik hesaplamalara uygulanan ilk istatistiksel algoritmalar arasında yer alan genel amaçlı tahmin ve sınıflandırma mekanizmasıdır [19]. Karar ağacının yapısı Şekil 3.10'da gösterilmiştir.

Normal bir ağacın kökü, dalları ve yaprakları bulunmaktadır. Karar ağacı algoritmasında da normal bir ağaca benzer bir yapı bulunmaktadır. Karar ağacı, kök düğümü, dalları, iç düğümleri ve yaprak düğümleri içermektedir. Karar ağacı yapısı ilk olarak kök düğümle başlamaktadır Kök düğümden çıkan dallar iç düğümlere besleme yapar ve bu olay yapraklara kadar uzanır. Yapraklar veri kümesindeki tüm olası sonuçları temsil etmektedir [20].



Şekil 3.10 Karar Ağacının Yapısı

Karar ağacı algoritmasında dallanmaların yapılabilmesi için bazı kavramlara ihtiyaç duyulmaktadır. Bunlar, entropi, Gini katsayısı, bilgi kazancıdır.

Entropi, verideki belirsizlik derecesinin ölçüsüdür. Eğer bir veri kümesindeki tüm elemanlar aynı sınıfa aitse entropi sıfıra eşittir. Entropiye ait formül aşağıdaki gibidir.

$$H(S) = -\sum_{i=1}^c p_i \cdot \log_2(p_i) \quad (3.22)$$

- $H(S)$: entropi,
- c : toplam sınıf sayısı,
- p_i : i sınıfının olasılığı olarak tanımlanmaktadır.

Gini indeksi, bir düğümdeki sınıfların olasılıklarını temel alarak verilerin ne kadar düzensiz olduğunu ölçmek için kullanılmaktadır. Eğer indeks düşükse düğüm daha saf, yüksekse daha düzensizdir. Gini indeksi aşağıdaki formül yardımıyla hesaplanmaktadır.

$$G = 1 - \sum_{i=1}^c p_i^2 \quad (3.23)$$

- G: Gini indeksi,
- c: toplam sınıf sayısı,
- p_i : i sınıfının olasılığı olarak tanımlanmaktadır.

Bilgi kazancı, veri kümesinin bölünmesi sonucunda entropideki düşüşü ölçmektedir ve aşağıdaki formül kullanılarak hesaplanmaktadır.

$$IG(S, A) = H(S) - \sum_{j=1}^k \frac{|S_j|}{|S|} \cdot H(S_j) \quad (3.24)$$

- $IG(S, A)$: bilgi kazancı
- S: başlangıç veri kümesi
- A: veri kümesini bölen özellik
- S_j : bölünme sonrası ortaya çıkan alt küme
- k: alt küme sayısı
- $|S_j|$: alt kümenin boyutu
- $|S|$: başlangıç veri kümesinin boyutu olarak tanımlanmaktadır.

3.4.4. SVM Algoritması

SVM, Vapnik-Chervonenkis teorisine dayanan güçlü teorik temellere sahip, güçlü, son teknoloji ürünü bir algoritmadır. Bu algoritmanın mantığı iki sınıfa ait verilerin bir hiper düzlem yardımıyla birbirlerinden ayrılmasına dayanmaktadır [21]. Bu düzlem iki sınıf arasındaki boşluğu (marjini) maksimum seviyede tutmaya çalışır. Burada veri kümesi aşağıdaki gibi temsil edilmektedir.

$$\{(x_i, y_i)\}_{i=1}^N \quad (3.25)$$

- x_i : Giriş verileri
- y_i : Sınıf etiketleri
- N: Veri kümesindeki örnek sayısı olarak tanımlanmaktadır.

SVM algoritmasının amacı verileri iki sınıfa ayıran bir hiper düzlem bulmaktır. Hiper düzlem, aşağıdaki gibi ifade edilmektedir.

$$w \cdot x + b = 0 \quad (3.26)$$

- w : ağırlık vektörü
- b : bias olarak tanımlanmaktadır.

Destek vektörleri ayırım hiper düzleme en yakın örnekler olarak tanımlanmaktadır. SVM algoritmasının amacı, bu hiper düzlemi, her iki sınıfın en yakın üyelerinden mümkün olduğunca uzak olacak şekilde yönlendirmektir. Destek vektörlerinin yer aldığı iki düzlem aşağıdaki gibi tanımlanmaktadır.

$$x_i \cdot w + b = +l \quad (3.27)$$

$$x_i \cdot w + b = -l \quad (3.28)$$

Hiper düzlemin iki destek vektöründen olan uzaklıkları birbirlerine eşittir ve bu uzaklık marjın olarak adlandırılır. Marjının formülü aşağıdaki gibidir.

$$\text{Marjin} = \frac{l}{\|w\|} \quad (3.29)$$

Marjının maksimize edilebilmesi için hiper düzlemin konumlandırılması gerekmektedir. Bu da aşağıda gösterildiği gibi $\|w\|$ değerinin minimize edilmesiyle mümkündür.

$$\min \frac{1}{2} \|w\|^2 \quad (3.30)$$

Her veri noktası için aşağıdaki gibi kısıtlama uygulanır.

$$y_i(x_i \cdot w + b) - l \geq 0 \quad (3.31)$$

Bu kısıtlamanın karşılanabilmesi için Lagrange çarpanları (α) kullanılır.

$$L_p = \min \frac{1}{2} \|w\|^2 - \sum_{i=1}^L \alpha_i [y_i(x_i \cdot w + b) - 1] \quad (3.32)$$

$$L_D = \sum_{i=1}^L \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i \cdot x_j \quad (3.33)$$

Yukarıdaki formüllerde L_p primitif formu, L_D ise ikili formu temsil etmektedir. L_D maksimumuna çıkartılırken aşağıdaki kısıtlamalar dikkate alınmalıdır.

$$\alpha_i \geq 0 \quad (3.34)$$

$$\sum_{i=1}^L \alpha_i y_i = 0 \quad (3.35)$$

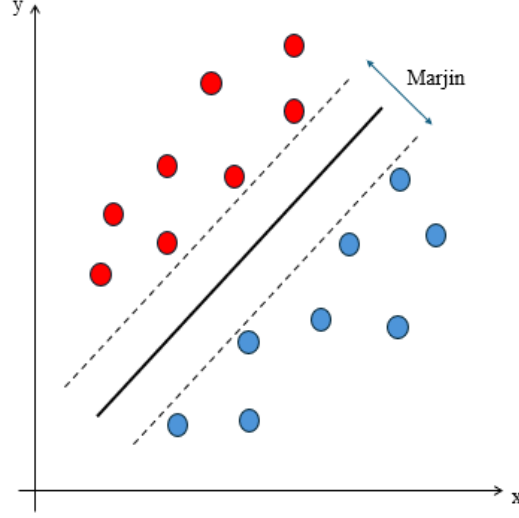
Son olarak w ve b değerleri aşağıdaki formüller yardımıyla hesaplanarak iki sınıf arasındaki ayrımı maksimize eden bir hiperdüzlem bulunması sağlanmaktadır.

$$w = \sum_{i=1}^L \alpha_i y_i x_i \quad (3.36)$$

$$b = y_S - \sum_{M \in S} \alpha_M y_M x_M \cdot x_S \quad (3.37)$$

Yukarıdaki formülde S destek vektörlerinin indeksleri, y_S ise destek vektörlerinin sınıf etiketini temsil etmektedir.

SVM algoritmasının yapısı Şekil 3.11’de gösterilmiştir.



Şekil 3.11 SVM Algoritmasının Yapısı

3.5. Performans Metrikleri

Makine öğrenmesi algoritmalarının performansının değerlendirilmesi için kullanılan birçok farklı metrik bulunmaktadır. Makine öğrenmesinde performans metrikleri, bir modelin ne kadar iyi çalıştığını, veri setindeki örnekleri ne kadar doğru sınıflandırdığını veya tahmin ettiğini ölçmek için kullanılır. Kullanılacak olan performans metrikleri, kullanılacak olan makine öğrenmesi algoritmasına ve veri setinin özelliklerine bağlı olarak seçilmektedir. Bu tezde kullanılacak olan algoritmaların performansını değerlendirmek için doğruluk, hassasiyet, hassasiyet ve F1 skoru metrikleri kullanılmıştır.

3.5.1. Karmaşıklık Matrisi (Confusion Matrix)

Sınıflandırma problemlerinde, ikiye ikilik bir karmaşıklık matrisi, dört olası sonucun her biri tarafından tahmin edilen örnek sayısını gösterir. Matris gerçek pozitiflerin sayısı (TP), gerçek negatiflerin sayısı (TN), yanlış pozitiflerin sayısı (FP) ve yanlış negatiflerin sayısını (FN) içermektedir. Sınıflandırıcı performans ölçümlerinin çoğu bu dört değerden türetilmektedir [25]. TP, modelin doğru bir şekilde pozitif olarak tahmin ettiği örneklerdir. TN, modelin doğru bir şekilde negatif olarak tahmin ettiği örneklerdir. FP, modelin pozitif

olarak yanlış bir şekilde tahmin ettiği örneklerdir. FN, modelin negatif olarak yanlış bir şekilde tahmin ettiği örneklerdir. Karmaşıklık matrisi Şekil 3.6,'da gösterildiği gibidir.

		Gerçek Değer	
		Pozitif	Negatif
Tahmin Edilen Değer	Pozitif	Gerçek Pozitif (TP)	Yanlış Pozitif (FP)
	Negatif	Yanlış Negatif (FN)	Gerçek Negatif (TN)

Şekil 3.12 Karmaşıklık Matrisi

3.5.2. Doğruluk (Accuracy)

Doğruluk, bir sınıflandırma metodunu değerlendirmek için en yaygın ve en basit ölçüdür. Bir modelin doğru tahminlerinin derecesi (ya da tam tersi, yanlış sınıflandırma hatalarının yüzdesi) olarak tanımlanır. Doğruluk aşağıdaki formül yardımıyla hesaplanmaktadır [26].

$$Doğruluk = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.38)$$

3.5.3. Hassasiyet (Precision)

Hassasiyet, pozitif olarak sınıflandırılan örneklerin gerçekten pozitif olan kısmını ölçer. Bu performans metriği, aşağıdaki formülde gösterildiği gibi gerçek pozitif değerlerin, gerçek pozitif ve yanlış pozitif değerlerin toplamına bölünmesiyle elde edilmektedir [27].

$$Hassasiyet = \frac{TP}{TP + FP} \quad (3.39)$$

3.5.4. Duyarlılık (Recall)

Duyarlılık, bir sınıflandırma metodundaki pozitif olan örneklerin ne kadarının tahmin edildiğini gösteren bir performans metriğidir. Duyarlılık, aşağıdaki formülde gösterildiği

gibi gerçek pozitif sayısının gerçek pozitiflerin ve yanlış negatiflerin toplamına bölünmesiyle hesaplanır [28].

$$Duyarlılık = \frac{TP}{TP + FN} \quad (3.40)$$

3.5.5. F1 Skoru

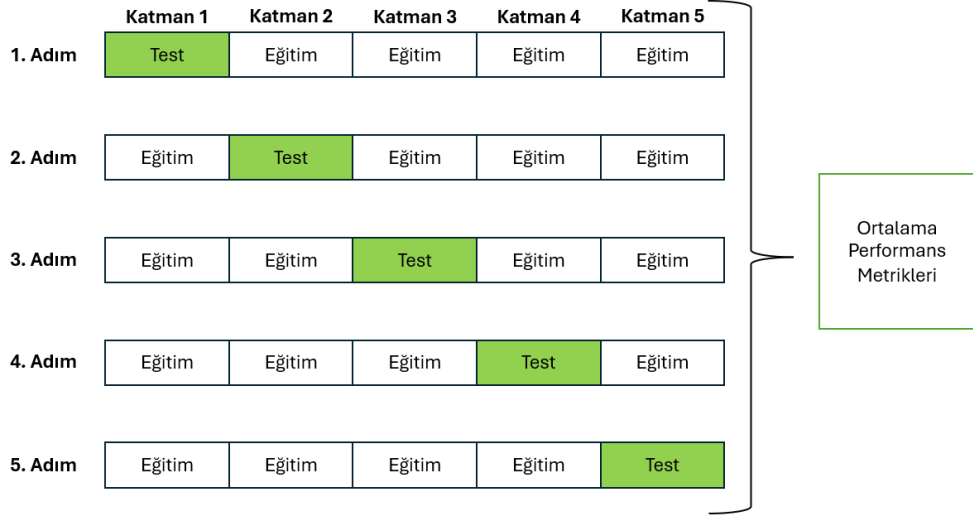
F1 skoru, f-puanı veya f-ölçüsü olarak da bilinmektedir. Bir algoritmanın performansını hesaplamak için hem hassasiyeti hem de duyarlılığı dikkate alır. Matematiksel olarak, aşağıdaki formülde gösterildiği gibi hassasiyet ve duyarlılığın harmonik ortalamasıdır.

$$F1 Skoru = \frac{2 \cdot Hassasiyet \cdot Duyarlilik}{Hassasiyet + Duyarlilik} \quad (3.41)$$

3.6. Çapraz Doğrulama (Cross Validation)

Çapraz doğrulama, model seçimi ve değerlendirmesi için kullanılan veriyi yeniden örnekleme yöntemlerinden biridir. Bu yöntemin amacı aşırı uyumu önlemek ve tahmin modellerinin genelleme hatasını tahmin etmek için kullanılmaktadır [38]. K-katmanlı, rastgele, iç içe gibi birçok çapraz doğrulama yöntemi bulunmaktadır.

K-katmanlı çapraz doğrulamada, ilk olarak veri seti “k” adet alt küme ve parçaya bölünmektedir. Daha sonra her bir parça sırasıyla test seti olacak (kalanlar eğitim seti) şekilde performans metrikleri hesaplanır. Daha sonra bu performans metriklerinin ortalaması alınarak ortalama performans metrikleri hesaplanmaktadır. Bu sayede her parça test verisi olarak, modelin performansının ayrıntılı bir şekilde değerlendirilmesine olanak sağlamaktadır. K-katmanlı çapraz doğrulamada k değerinin 5 olarak seçildiği bir örnek aşağıdaki gibidir.



Şekil 3.13 5 Katmanlı Çapraz Doğrulama Örneği

4. DENEY SONUÇLARI

Bu tez çalışması, MATLAB platformunda yazılan kodlar ile yapılmıştır. İlk olarak VGG16, DenseNet201, MobileNetV2 ve ResNet18 algoritmaları kullanılarak test ve veri seti üzerinde özellik çıkarımı yapılmıştır. Bu özellikler kullanılarak sınıflandırma yapılması amacıyla dört farklı makine öğrenme algoritması uygulanmıştır. Bu algoritmalar, K-Nearest Neighbors (KNN), Naive Bayes, Karar Ağacı ve Destek Vektör Makinesi (SVM) olarak belirlenmiştir. Bu adımda, makine öğrenmesi algoritmalarının VGG16, DenseNet201, MobileNetV2 ve ResNet18 algoritmalarının performansına etkisi gözlemlenmiştir.

Yapılan uygulamalar MATLAB'da yazılan kodlar ile gerçekleştirilmiştir. Yazılan kodlar toplamda 8000 görüntü üzerinde çalışılır; bu görüntüler yeşil Arabica kahve çekirdeğinin dört farklı türüne (defect, longberry, peaberry, premium) göre sınıflandırılmıştır. Her sınıfa ait 2000 adet görüntü bulunmaktadır.

Bir sonraki aşamada, yine konvolüsyonel sinir ağı algoritmalarıyla çıkarılan özellikler doğrultusunda eğitim verisine Temel Bileşen Analizi (PCA) yöntemi uygulanmıştır. PCA, yüksek boyutlu verilerin daha düşük boyutlu bir uzaya dönüştürülmesini sağlayarak, verinin varyansını korur ve temel özellikleri özetler. PCA uygulandıktan sonra, KNN, Naive Bayes, Karar Ağacı ve SVM algoritmaları, bu yeni özellik seti üzerinde tekrar test edilmiştir. Bu adım, PCA'nın algoritmaların performansına etkisini incelemeyi amaçlamaktadır.

Bir sonraki aşamada hem eğitim hem de test verilerinde Yönlü Gradyanlar Histogramı (HoG) özellikleri çıkartılmıştır. HoG, görüntülerdeki nesnelere ve şekillerin tanımlanmasında kullanılan bir yöntemdir ve görüntülerden öznelikler çıkarır. HoG özellikleri çıkarıldıktan sonra, aynı algoritmalar (KNN, Naive Bayes, Karar Ağacı ve SVM) bu yeni özellik seti üzerinde tekrar test edilmiştir. Bu işlem, HoG'nin algoritmaların performansı üzerindeki etkilerini incelemeye yöneliktir.

Kullanılan makine öğrenmesi algoritmaları tüm aşamalarda aynı düzende kullanılmıştır.

KNN algoritması uygulanırken k değerine 1, 3, 5 ve 7 değerleri atanıp performans metriklerinin k'nın değişen değerlerine göre nasıl bir değişim gösterdiği incelenmiştir. Burada tek sayı kullanılmasının sebebi sınıflandırma işlemi sırasında birden fazla sınıf arasında oylama yapıldığında, tek bir sonuca ulaşmak istenmesinden kaynaklanmaktadır.

Örneğin, $k=3$ olarak seçildiğinde, model üç komşunun sınıfına bakar ve hangi sınıf daha fazla komşu tarafından temsil ediliyorsa o sınıf seçilir. Böylece, eşit oy durumu önlenmiş olur. k değerinin çift sayı seçildiği (örneğin $k=4$) durumda iki komşu bir sınıfa, diğer iki komşu başka bir sınıfa ait olabilir. Bu durum, bir beraberlik ortaya çıkarabilir ve modelin hangi sınıfı seçeceğine dair belirsizlik yaratabilir. KNN sınıflandırıcılarını eğitmek için MATLAB’da bulunan “fitcknn” fonksiyonu kullanılmıştır. Bu fonksiyon, verilen eğitim veri seti ve bu veri setine ait etiketlere dayanarak bir KNN modelini oluşturur. Fonksiyonun içerisinde mesafe ölçütü belirtilmediği için, MATLAB varsayılan mesafe ölçütü olarak öklid uzaklığını kullanarak modeli oluşturmaktadır. Modelde komşular eşit ağırlıklıdır ve özellikler normalize edilmeden kullanılmaktadır.

Naive Bayes sınıflandırıcısını eğitmek için MATLAB’da bulunan “fitcnb” fonksiyonu kullanılmıştır. Bu fonksiyon, verilen eğitim veri seti ve bu veri setine ait etiketlere dayanarak bir Naive Bayes modeli oluşturur. Fonksiyonun içerisinde başka bir ölçüt belirtilmediği için, MATLAB varsayılan ayarları kullanarak bir Naive Bayes modeli oluşturmaktadır. Özelliklerin Gaussian (normal) dağılıma sahip olduğu varsayılır. Eğitim verilerindeki sınıf etiketlerini otomatik olarak kullanır.

Karar ağacı sınıflandırıcısını eğitmek için MATLAB’da bulunan “fitctree” fonksiyonu kullanılmıştır. Bu fonksiyon, verilen eğitim veri seti ve bu veri setine ait etiketlere dayanarak bir karar ağacı modeli oluşturur. Fonksiyonun içerisinde başka bir ölçüt belirtilmediği için, fonksiyon, karar ağacını varsayılan ayarlarla oluşturur. Bu varsayılan ayarlar, ağacın yapısını ve modelin performansını doğrudan etkileyen çeşitli matematiksel ve yapısal özellikleri içerir. Varsayılan olarak, karar ağacı oluşturulurken Gini indeksi kriteri kullanılarak bir düğümdeki örneklerin hangi sınıflara ait olduğu ölçülür. Varsayılan ayar olarak ağaç derinliği sınırlandırılmadan büyür. Minimum yaprak boyutu 1 olarak belirlenir ve budama işlemi yapılmaz. Bu olay da her yaprak düğümünde en az bir örnek bulunacağı anlamına gelir. Bir düğümün bölünebilmesi için gereken minimum ebeveyn boyutu 2’dir.

SVM sınıflandırıcısını eğitmek için MATLAB’da bulunan “fitcecoc” fonksiyonu kullanılmıştır. Bu fonksiyon, verilen eğitim veri seti ve bu veri setine ait etiketlere dayanarak bir SVM modeli oluşturur ve iki sınıf arasındaki ayrımı en iyi şekilde sağlayan bir hiperdüzlem bulmak için destek vektör makineleri algoritmasını kullanır. Fonksiyonun içerisinde başka bir ölçüt belirtilmediği için, fonksiyon, SVM algoritmasını varsayılan ayarlarla oluşturur. Bu varsayılan ayarlar, SVM’in yapısını ve modelin performansını

doğrudan etkileyen çeşitli matematiksel ve yapısal özellikleri içerir. Varsayılan ayarlar SVM'in doğrusal kernel fonksiyonunu kullanmasını sağlar. Bu, veri noktalarını doğrusal bir hiperdüzlemle ayırır. Veriler standartlaştırılır. Bu işlem, her özelliğin ortalamasının 0 ve standart sapmasının 1 olacak şekilde ölçeklendirilmesini ifade eder. Matematiksel olarak, SVM, iki sınıf arasındaki hiperdüzlemi bulmayı ve bu hiperdüzlemin marjini maksimize etmeyi hedefler.

Performans metrikleri hem sınıf hem de toplam bazda hesaplanmıştır. Karmaşıklık matrisindeki her sınıfa ait doğru pozitif (TP), yanlış pozitif (FP), yanlış negatif (FN) ve doğru negatif (TN) değerleri birlikte kullanılarak, veri setine ait toplam performans metrikleri elde edilmiştir. Her katmanda bulunan doğruluk değerlerinin ortalaması alınarak da ortalama doğruluk değerleri hesaplanmaktadır. Toplam doğruluk 5 katmanlı çapraz doğrulama sırasında her katman için eğitim doğruluğu ve test doğruluğu değerleri ayrı ayrı kaydedilmiştir. Her katman için elde edilen eğitim doğruluğu, test doğruluğu, hassasiyet, duyarlılık ve F1 skoru değerlerinin ortalaması alınarak genel performans metrikleri hesaplanmıştır. Bu metrikler, modelin genel başarımını yansıtmaktadır.

Son olarak, her bir aşamada bulunan her bir algoritma için elde edilen doğruluk değerleri hesaplanmış ve bu değerler karşılaştırılmıştır. Bu karşılaştırma, PCA ve HoG uygulamalarının algoritmaların performansı üzerindeki etkilerini değerlendirmek amacıyla yapılmıştır. Elde edilen sonuçlar, her algoritmanın PCA ve HoG özellikleri ile nasıl performans gösterdiğini göstermektedir.

4.1. CNN Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

4.1.1. MobileNetV2 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

VGG16 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “mobilenetv2()” komutu ile önceden eğitilmiş olan MobileNetV2 modeli yüklenir. Daha sonra “activations” komutu kullanılarak MobileNetV2 modelinde bulunan “global_average_pooling2d_1” katmanında özellik çıkarımı yapılır. Bu katman MobileNetV2 modelinin tam bağlantılı katmanlarından biridir. Çıkarılan özellikler

oluşturulan boş matrislere eklenerek sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmalarında kullanılmaktadır. MobileNetV2 ile yapılan bu çalışmada her bir görüntüye ait 1280 özellik kullanılmaktadır.

4.1.1.1. MobileNetV2 Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k 'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.1 MobileNetV2+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	68,61	69,80	68,61	68,30
k=3	72,39	72,87	72,39	71,99
k=5	73,94	75,20	73,94	73,51
k=7	74,29	75,92	74,29	73,79

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı %80,34 iken, modelin test doğruluk oranı %74,29 olarak belirlenmiştir. Modelin hassasiyet değeri %75,92 olarak hesaplanmıştır. Bu metrik, modelin pozitif olarak sınıflandırdığı örneklerin ne kadarının gerçek pozitif olduğunu ifade etmektedir. Modelin duyarlılık değeri %74,29 olarak tespit edilmiştir. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir şekilde pozitif olarak sınıflandırıldığını göstermektedir. Son olarak, modelin F1 skoru 73,79 olarak bulunmuştur.

Bu kapsamda $k=7$ için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

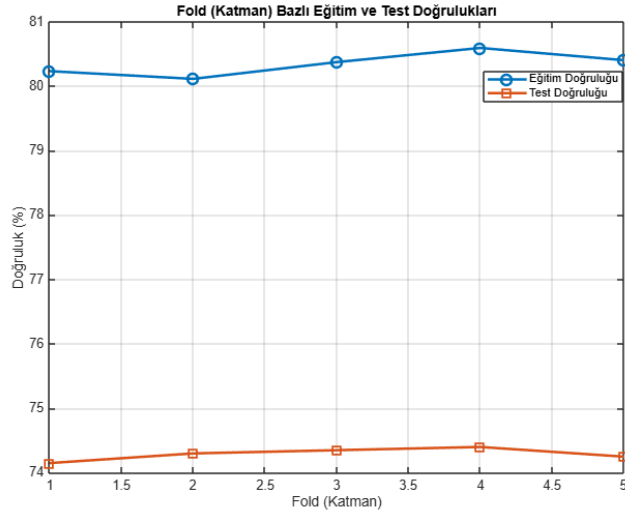
Tablo 4.2 MobileNetV2+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	221	47	52	80
	Longberry	5	371	9	15
	Peaberry	12	37	284	67
	Premium	13	35	40	312

Tablo 4.3 MobileNetV2+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	55,35	88,28	55,35	68,02
Longberry	92,70	75,77	92,70	83,38
Peaberry	71,05	73,82	71,05	72,39
Premium	65,80	78,05	65,80	71,39

Performans metrikleri sınıf bazında incelendiğinde, k değeri farketmeksizin performans metriklerinde genel olarak longberry sınıfının diğerlerinden daha fazla olduğu gözlenmektedir. Defect sınıfı ise genel olarak en düşük performans metriklerine sahip olan sınıf olmuştur.

**Şekil 4.1** MobileNetV2+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.1.1.2. MobileNetV2 Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.4 MobileNetV2+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	382	5	0	13
	Longberry	359	16	0	25
	Peaberry	363	6	0	31
	Premium	342	3	0	55

Tablo 4.5 MobileNetV2+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

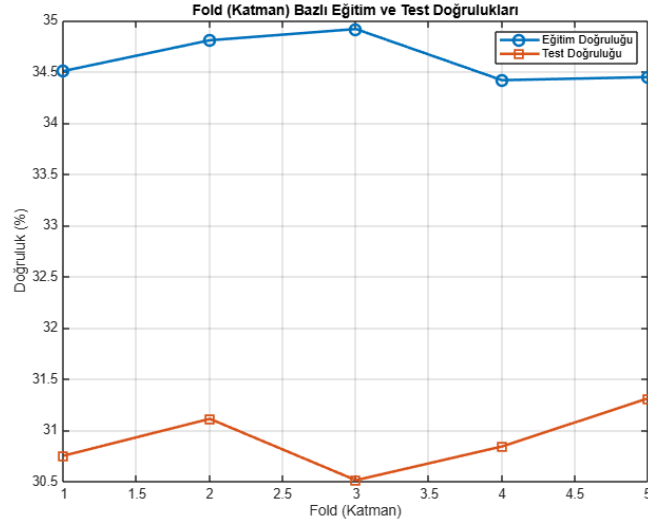
	Accuracy	Precision	Recall	F-1 Score
Defect	95,50	26,42	95,50	41,39
Longberry	4,00	53,33	4,00	41,39
Peaberry	0,00	0,00	0,00	0,00
Premium	13,75	44,35	13,75	20,99

Tablo 4.6 MobileNetV2+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	30,90	0,00	30,90	0,00

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim doğruluk oranı %34,62 olarak hesaplanmıştır. Test veri setindeki örneklerin %30,90'ı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %0,00 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %30,90 olarak bulunmuştur. Son olarak, modelin F1 skoru %0,00 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin defect sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise peaberry sınıfına aittir.



Şekil 4.2 MobileNetV2+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.1.3. MobileNetV2 Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.7 MobileNetV2+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	232	37	73	58
	Longberry	41	273	37	49
	Peaberry	62	32	239	67
	Premium	76	48	67	209

Tablo 4.8 MobileNetV2+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

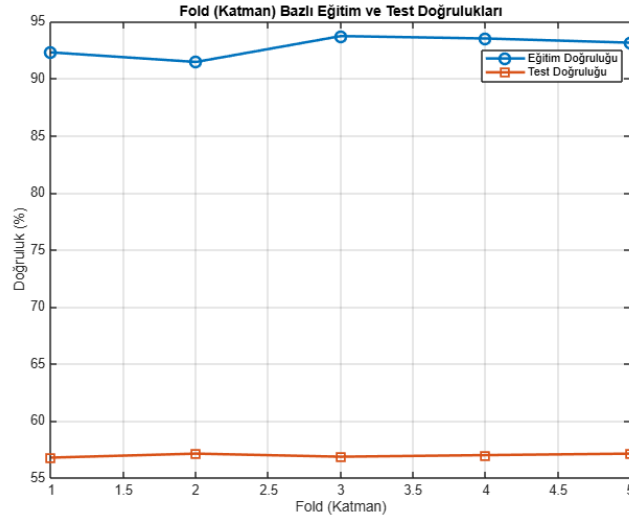
	Accuracy	Precision	Recall	F-1 Score
Defect	58,00	56,45	58,00	57,21
Longberry	68,25	70,00	68,25	69,11
Peaberry	59,75	57,45	59,75	58,58
Premium	52,25	54,57	52,25	53,38

Tablo 4.9 MobileNetV2+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	56,96	57,04	56,96	56,97

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim veri setindeki doğruluk oranı %92,84 olarak bulunmuştur. Test veri setindeki örneklerin %56,96'sı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %57,04 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %56,96 olarak bulunmuştur. Son olarak, modelin F1 skoru %56,97 olarak tespit edilmiştir. Bu değer, modelin yanlış pozitif ve yanlış negatif sonuçları minimize etme yeteneğini yansıtmaktadır.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.3 MobileNetV2+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.1.4. MobileNetV2 Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.10 MobileNetV2+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	303	25	26	46
	Longberry	14	358	7	21
	Peaberry	27	7	336	30
	Premium	34	17	33	316

Tablo 4.11 MobileNetV2+SVM Modelinin Sınıf Bazında Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Defect	75,75	80,16	75,75	77,89
Longberry	89,50	87,96	89,50	88,72
Peaberry	84,00	83,58	84,00	83,79
Premium	79,00	76,51	79,00	77,74

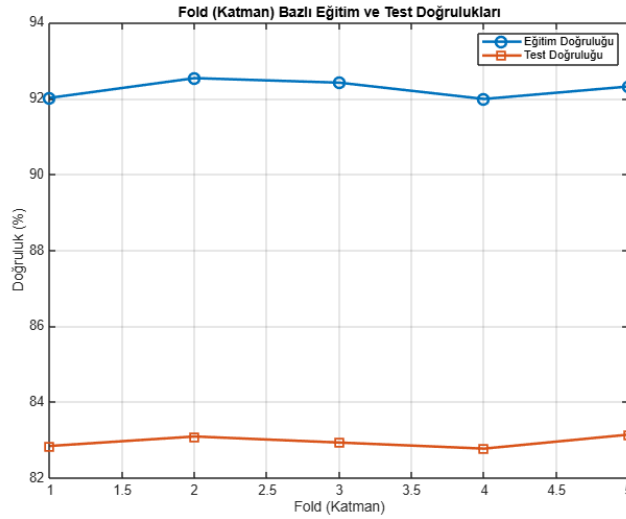
Tablo 4.12 MobileNetV2+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	82,95	83,05	82,95	82,98

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %92,26 iken, test doğruluk oranı %82,95 olarak hesaplanmıştır. Hassasiyet değeri ise %83,05 olarak tespit edilmiştir. Bu metrik, modelin pozitif sınıfa ait olarak sınıflandırdığı örneklerin ne kadarının gerçekte pozitif olduğunu ifade etmektedir. Modelin duyarlılık değeri %82,95 olarak belirlenmiştir. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir biçimde pozitif olarak sınıflandırıldığını göstermektedir. F1 skoru ise %82,98 olarak bulunmuştur. F1 skoru,

hassasiyet ve duyarlılık arasındaki dengeyi oluşturarak modelin genel performansını değerlendiren bir metriktir.

Alt sınıfların performans metriklerine bakıldığında en yüksek doğruluk oranı %89,50 ile longberry sınıfındayken, en düşük doğruluk oranı %75,75 ile defect sınıfındadır.



Şekil 4.4 MobileNetV2+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.2. ResNet18 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

ResNet18 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “resnet18()” komutu ile önceden eğitilmiş olan ResNet18 modeli yüklenir. Daha sonra “activations” komutu kullanılarak ResNet18 modelinde bulunan “pool5” katmanında özellik çıkarımı yapılır. Bu katman ResNet18 modelinin tam bağlantılı katmanlarından biridir. Çıkarılan özellikler oluşturulan boş matrislere eklenerek sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmalarında kullanılmaktadır. VGG16 ile yapılan bu çalışmada her bir görüntüye ait 512 özellik kullanılmaktadır.

4.1.2.1. Resnet18 Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.13 ResNet18+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	70,09	71,37	70,09	69,95
k=3	73,97	74,33	73,97	73,71
k=5	75,05	76,09	75,05	74,73
k=7	75,62	76,95	75,63	75,23

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluğu %81,85 iken, modelin test doğruluk oranı %75,62 olarak belirlenmiştir. Modelin hassasiyet değeri %76,95 olarak hesaplanmıştır. Modelin duyarlılık değeri %75,63 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 75,23 olarak bulunmuştur.

Bu kapsamda k=7 için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

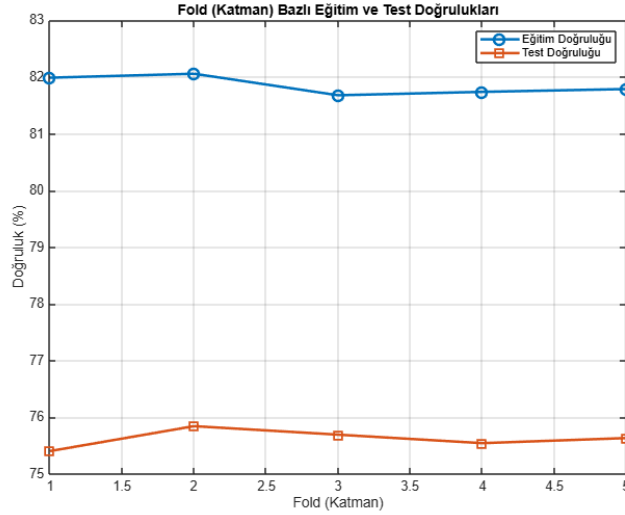
Tablo 4.14 ResNet18+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	234	41	64	61
	Longberry	2	375	7	16
	Peaberry	12	27	305	56
	Premium	13	37	55	295

Tablo 4.15 ResNet18+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	58,60	89,74	58,60	70,88
Longberry	93,70	78,21	93,70	85,23
Peaberry	76,35	70,94	76,35	73,53
Premium	73,85	68,92	73,85	71,27

Performans metrikleri sınıf bazında incelendiğinde, k değeri farketmeksizin performans metriklerinde genel olarak longberry sınıfının diğerlerinden daha fazla olduğu gözlenmektedir. Defect sınıfı ise genel olarak en düşük performans metriklerine sahip olan sınıf olmuştur.



Şekil 4.5 ResNet18+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.1.2.2. ResNet18 Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.16 ResNet18+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	166	40	176	18
	Longberry	44	222	112	22
	Peaberry	50	35	297	18
	Premium	39	53	269	39

Tablo 4.17 ResNet18+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

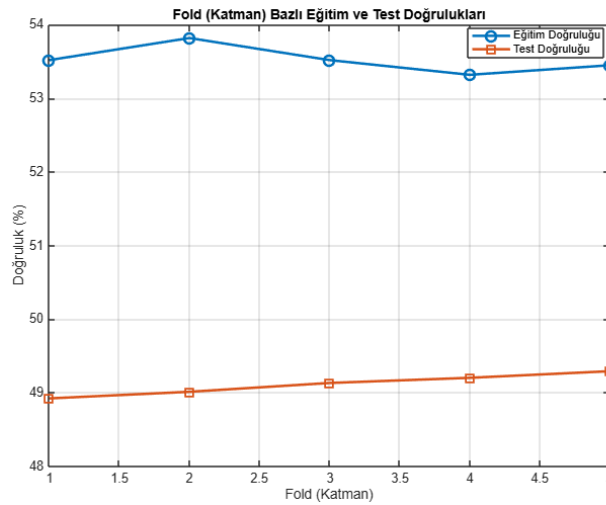
	Accuracy	Precision	Recall	F-1 Score
Defect	41,50	55,52	41,50	47,50
Longberry	55,50	63,43	55,50	59,20
Peaberry	74,25	34,78	74,25	47,37
Premium	9,75	40,21	9,75	15,69

Tablo 4.18 ResNet18+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	49,11	52,90	49,11	46,27

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim doğruluk oranı %53,53 olarak hesaplanmıştır. Test veri setindeki örneklerin %49,11'i doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %52,90 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %49,11 olarak bulunmuştur. Son olarak, modelin F1 skoru %46,27 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin peaberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.

**Şekil 4.6** ResNet18+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.2.3. ResNet18 Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.19 ResNet18+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	247	40	68	45
	Longberry	42	293	32	33
	Peaberry	69	38	206	87
	Premium	60	48	90	202

Tablo 4.20 ResNet18+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

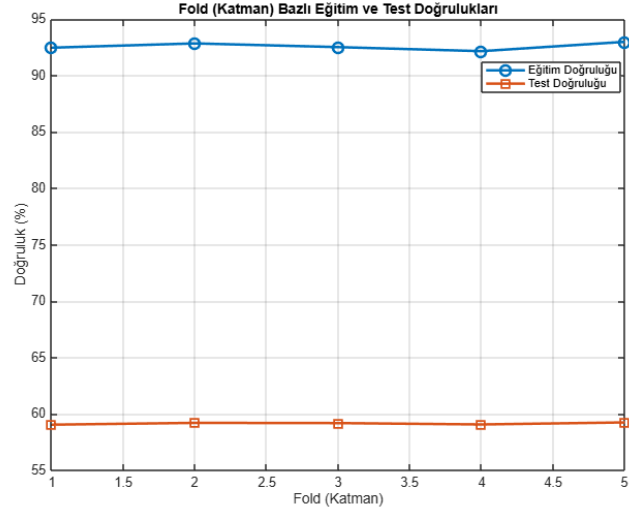
	Accuracy	Precision	Recall	F-1 Score
Defect	61,75	59,09	61,75	60,39
Longberry	73,25	69,93	73,25	71,55
Peaberry	51,50	52,02	51,50	51,76
Premium	50,50	55,04	50,50	52,67

Tablo 4.21 ResNet18+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	59,12	59,10	59,12	59,07

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim veri setindeki doğruluk oranı %92,62 iken, test veri setindeki örneklerin %59,12'si doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %59,10 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %59,12 olarak bulunmuştur. Son olarak, modelin F1 skoru %59,07 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.7 ResNet18+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.2.4. ResNet18 Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.22 ResNet18+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	329	14	31	26
	Longberry	22	357	8	13
	Peaberry	24	10	337	29
	Premium	33	13	22	332

Tablo 4.23 ResNet18+SVM Modelinin Sınıf Bazında Performans Metrikleri

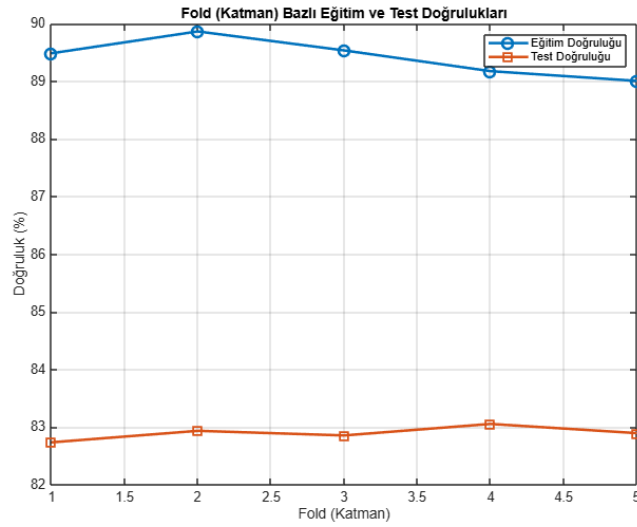
	Accuracy	Precision	Recall	F-1 Score
Defect	82,25	80,64	82,25	81,44
Longberry	89,25	90,61	89,25	89,92
Peaberry	84,25	84,67	84,25	84,46
Premium	83,00	83,00	83,00	83,00

Tablo 4.24 ResNet18+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	82,89	82,95	82,89	82,90

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluğu %89,42 olarak bulunmuştur. Modelin doğruluk oranı %82,89 olarak hesaplanmıştır. Hassasiyet değeri ise %82,95 olarak tespit edilmiştir. Modelin duyarlılık değeri %82,89 olarak belirlenmiştir. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir biçimde pozitif olarak sınıflandırıldığını göstermektedir. F1 skoru ise %82,90 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında doğruluk oranlarının %80'in üstünde olduğu gözlemlenmektedir. En yüksek doğruluk oranı %89,25 ile longberry sınıfındayken, en düşük doğruluk oranı %82,25 ile defect sınıfındadır.



Şekil 4.8 ResNet18+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.3. VGG16 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

VGG16 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “vgg16()” komutu ile önceden eğitilmiş olan VGG16 modeli yüklenir. Daha sonra “activations” komutu kullanılarak VGG16 modelinde bulunan “fc7” katmanında özellik çıkarımı yapılır. Bu katman VGG16 modelinin tam bağlantılı katmanlarından biridir. Çıkarılan özellikler oluşturulan boş matrislere eklenerek sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmalarında kullanılmaktadır. VGG16 ile yapılan bu çalışmada her bir görüntüye ait 4096 özellik kullanılmaktadır.

4.1.3.1. VGG16 Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.25 VGG16+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	70,01	71,49	70,01	69,81
k=3	72,95	71,49	72,95	72,64
k=5	74,10	75,24	74,10	73,75
k=7	74,60	76,10	74,60	74,27

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, eğitim doğruluk oranı %82,52, modelin test doğruluk oranı %74,60 olarak belirlenmiştir. Bu oran, test veri setindeki örneklerin %74,60'ının doğru bir şekilde sınıflandırıldığını göstermektedir. Modelin hassasiyet değeri %76,10 olarak hesaplanmıştır. Bu metrik, modelin pozitif olarak sınıflandırdığı örneklerin ne kadarının gerçek pozitif olduğunu ifade etmektedir. Modelin duyarlılık değeri %74,60 olarak tespit edilmiştir. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir şekilde pozitif olarak sınıflandırıldığını göstermektedir. Bu durumda, modelin pozitif örneklerin %74,60'ının doğru bir şekilde tanıdığı belirtilmektedir. Son

olarak, modelin F1 skoru %74,27 olarak bulunmuştur. F1 skoru, hassasiyet ve duyarlılık arasında bir denge oluşturarak modelin genel performansını değerlendiren bir metriktir.

Bu kapsamda $k=7$ için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

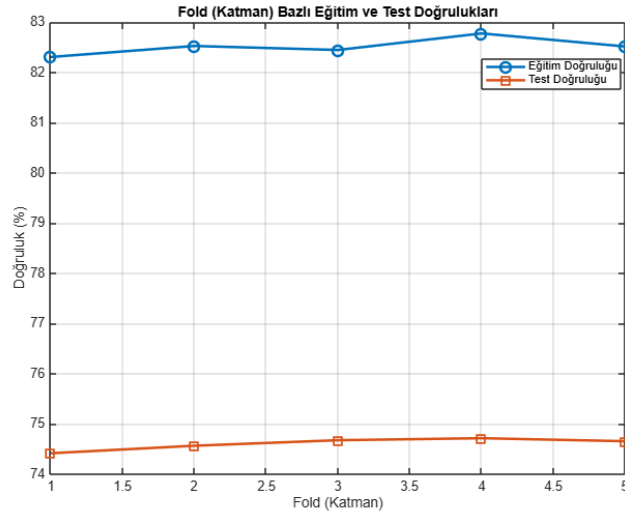
Tablo 4.26 VGG16+KNN ile Elde Edilen Karmaşıklık Matrisi ($k=7$)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	234	45	54	67
	Longberry	3	369	8	20
	Peaberry	15	31	289	65
	Premium	10	32	57	301

Tablo 4.27 VGG16+KNN Modelinin Sınıf Bazında Performans Metrikleri ($k=7$)

	Accuracy	Precision	Recall	F-1 Score
Defect	58,60	89,61	58,60	70,83
Longberry	92,25	77,37	92,25	84,12
Peaberry	72,30	70,93	72,30	71,60
Premium	75,25	66,50	75,25	70,55

Performans metrikleri sınıf bazında incelendiğinde, k değeri farketmeksizin performans metriklerinde genel olarak longberry sınıfının diğerlerinden daha fazla olduğu gözlenmektedir. Defect sınıfı ise genel olarak en düşük performans metriklerine sahip olan sınıf olmuştur.



Şekil 4.9 VGG16+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.1.3.2. VGG16 Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.28 VGG16+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	285	43	33	39
	Longberry	13	330	19	38
	Peaberry	63	52	252	33
	Premium	55	0	44	231

Tablo 4.29 VGG16+Naive Bayes Modeli Sınıf Bazında Performans Metrikleri

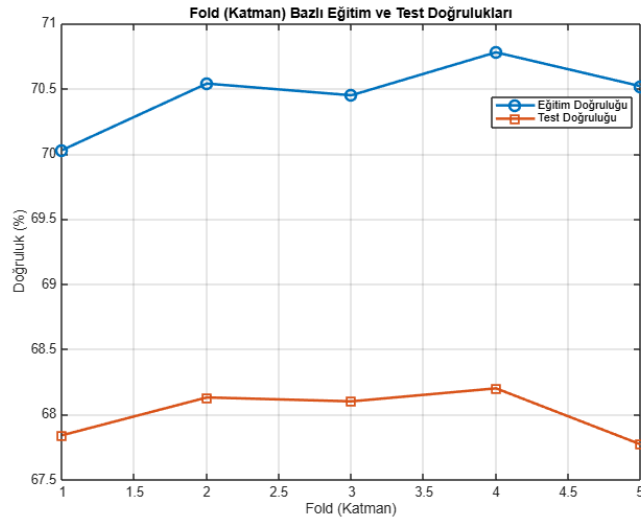
	Accuracy	Precision	Recall	F-1 Score
Defect	71,25	68,51	71,25	69,85
Longberry	82,50	73,74	82,50	73,74
Peaberry	63,00	72,41	63,00	67,38
Premium	57,75	67,74	57,75	62,35

Tablo 4.30 VGG16+Naive Bayes Modeli Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	68,01	67,96	68,01	67,77

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim doğruluk oranı %70,46 olarak bulunmuştur. Test veri setindeki örneklerin %68,01'i doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %67,96 olarak hesaplanmıştır. Modelin pozitif tahminlerinin %63,96'sının doğru olduğu anlaşılmaktadır. Modelin duyarlılık değeri ise %68,01 olarak bulunmuştur. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir biçimde pozitif olarak sınıflandırıldığını göstermektedir. Son olarak, modelin F1 skoru %67,77 olarak tespit edilmiştir. Bu değer, modelin yanlış pozitif ve yanlış negatif sonuçları minimize etme yeteneğini yansıtmaktadır.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.10 VGG16+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.3.3. VGG16 Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.31 VGG16+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	248	39	55	58
	Longberry	38	294	33	35
	Peaberry	65	40	235	60
	Premium	64	42	80	214

Tablo 4.32 VGG16+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

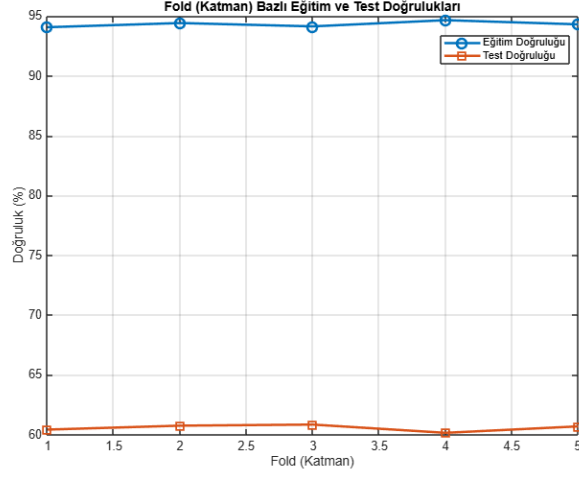
	Accuracy	Precision	Recall	F-1 Score
Defect	62,00	59,76	62,00	60,86
Longberry	73,50	70,84	73,50	72,15
Peaberry	58,75	58,31	58,75	58,53
Premium	53,50	58,31	53,50	55,80

Tablo 4.33 VGG16+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	60,56	60,44	60,56	60,46

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim doğruluk oranı %94,33 olarak hesaplanmıştır. Test veri setindeki örneklerin %60,56'sı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %60,46 olarak hesaplanmıştır. Modelin pozitif tahminlerinin %60,46'sının doğru olduğu anlaşılmaktadır. Modelin duyarlılık değeri ise %60,56 olarak bulunmuştur. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir biçimde pozitif olarak sınıflandırıldığını göstermektedir. Son olarak, modelin F1 skoru %60,46 olarak tespit edilmiştir. Bu değer, modelin yanlış pozitif ve yanlış negatif sonuçları minimize etme yeteneğini yansıtmaktadır.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.11 VGG16+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.3.4. VGG16 Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.34 VGG16+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	332	13	20	35
	Longberry	10	360	9	21
	Peaberry	20	9	336	35
	Premium	29	14	28	329

Tablo 4.35 VGG16+SVM Modelinin Sınıf Bazında Performans Metrikleri

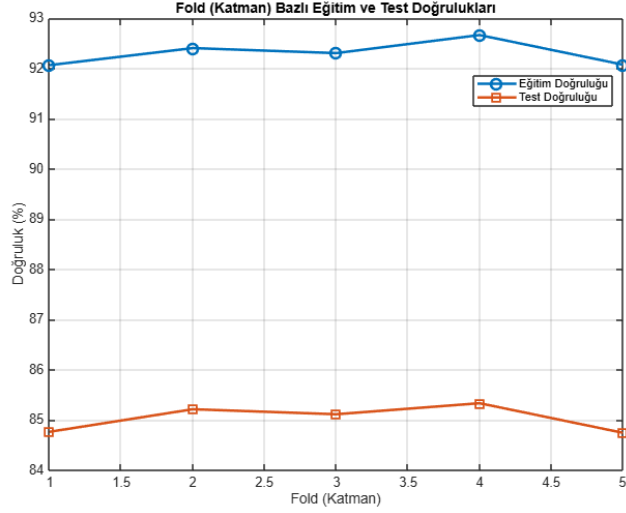
	Accuracy	Precision	Recall	F-1 Score
Defect	83,00	84,91	83,00	83,94
Longberry	90,00	90,91	90,00	90,45
Peaberry	84,00	85,50	84,00	84,74
Premium	82,25	78,33	82,25	80,24

Tablo 4.36 VGG16+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	85,03	85,09	85,03	85,04

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Eğitim doğruluk oranı %92,31 olarak hesaplanmıştır. Modelin doğruluk oranı %85,03 olarak hesaplanmıştır. Hassasiyet değeri ise %85,09 olarak tespit edilmiştir. Bu metrik, modelin pozitif sınıfa ait olarak sınıflandırdığı örneklerin ne kadarının gerçekte pozitif olduğunu ifade etmektedir. Modelin duyarlılık değeri %85,03 olarak belirlenmiştir. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir biçimde pozitif olarak sınıflandırıldığını göstermektedir. F1 skoru ise %85,04 olarak bulunmuştur. F1 skoru, hassasiyet ve duyarlılık arasındaki dengeyi oluşturarak modelin genel performansını değerlendiren bir metriktir.

Alt sınıfların performans metriklerine bakıldığında doğruluk oranlarının %80'inin üstünde olduğu gözlemlenmektedir. En yüksek doğruluk oranı %90 ile longberry sınıfındayken, en düşük doğruluk oranı %82,25 ile premium sınıfındadır.



Şekil 4.12 VGG16+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.4. DenseNet201 Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

DenseNet201 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “densenet201()” komutu ile önceden eğitilmiş olan DenseNet201 modeli yüklenir. Daha sonra “activations” komutu kullanılarak DenseNet201 modelinde bulunan “avg_pool” katmanında özellik çıkarımı yapılır. Bu katman DenseNet201 modelinin havuzlama katmanlarından biridir. Çıkarılan özellikler oluşturulan boş matrislere eklenerek sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmalarında kullanılmaktadır. DenseNet201 ile yapılan bu çalışmada her bir görüntüye ait 1920 özellik kullanılmaktadır.

4.1.4.1. DenseNet201 Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.37 DenseNet201+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	74,16	75,60	74,16	74,03
k=3	77,56	78,34	77,56	77,33
k=5	78,50	79,90	78,50	78,22
k=7	78,80	80,36	78,80	78,48

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, eğitim doğruluk oranı %85,28 iken, modelin test doğruluk oranı %78,80 olarak belirlenmiştir. Modelin hassasiyet değeri %80,36 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %78,80 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 78,48 olarak bulunmuştur.

Bu kapsamda k=7 için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

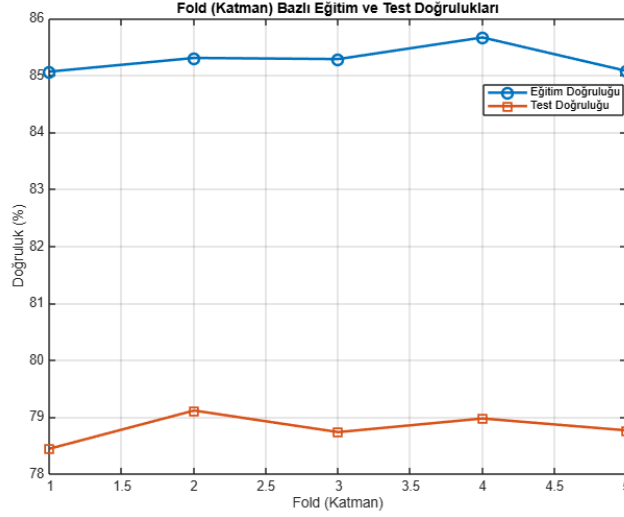
Tablo 4.38 DenseNet201+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	250	49	48	53
	Longberry	2	381	5	12
	Peaberry	7	32	320	41
	Premium	7	39	45	309

Tablo 4.39 DenseNet201+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	62,55	94,30	62,55	75,19
Longberry	95,30	76,03	95,30	84,58
Peaberry	80,10	76,60	80,10	78,30
Premium	77,25	74,50	77,25	75,84

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.13 DenseNet201+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.1.4.2. DenseNet201 Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.40 DenseNet201+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	294	31	34	41
	Longberry	7	361	13	19
	Peaberry	33	38	295	34
	Premium	35	51	29	285

Tablo 4.41 DenseNet201+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

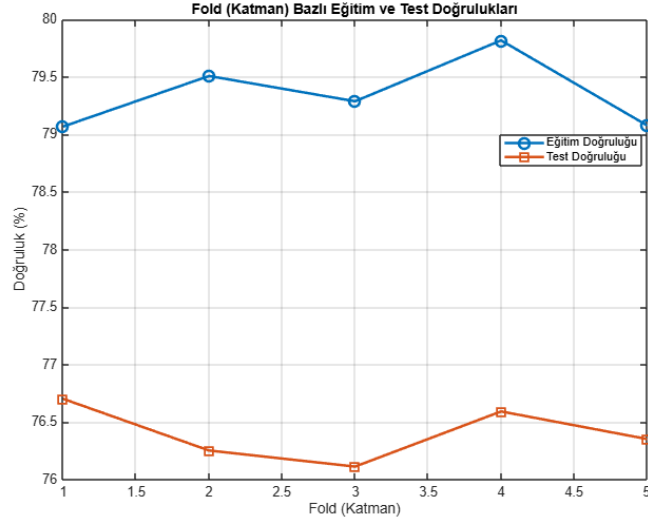
	Accuracy	Precision	Recall	F-1 Score
Defect	73,50	79,67	73,50	76,46
Longberry	90,25	75,05	90,25	81,95
Peaberry	73,35	79,51	73,75	76,52
Premium	71,25	75,20	71,25	73,17

Tablo 4.42 DenseNet201+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	76,40	76,45	76,40	76,26

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim doğruluk oranı %79,35 olarak hesaplanmıştır. Test veri setindeki örneklerin %76,40'ı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %76,45 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %76,40 olarak bulunmuştur. Duyarlılık, gerçek pozitif örneklerin ne kadarının doğru bir biçimde pozitif olarak sınıflandırıldığını göstermektedir. Son olarak, modelin F1 skoru %76,26 olarak tespit edilmiştir. Bu değer, modelin yanlış pozitif ve yanlış negatif sonuçları minimize etme yeteneğini yansıtmaktadır.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına, en düşük performans metrikleri ise premium sınıfına ait olduğu gözlenmektedir.



Şekil 4.14 DenseNet201+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.4.3. DenseNet201 Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.43 DenseNet201+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	223	50	73	54
	Longberry	36	308	21	35
	Peaberry	72	31	228	69
	Premium	69	44	62	225

Tablo 4.44 DenseNet201+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

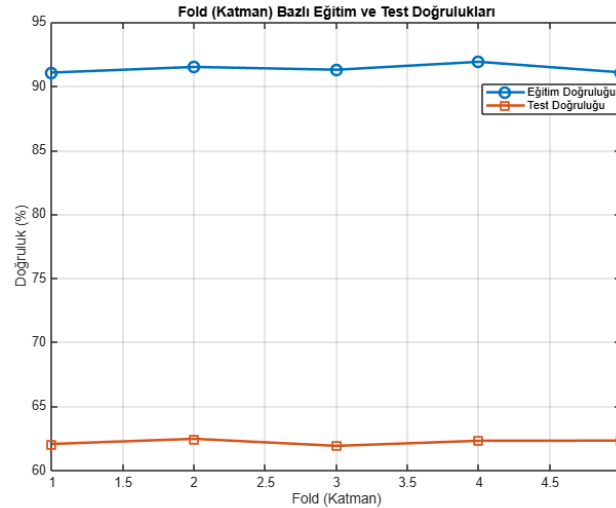
	Accuracy	Precision	Recall	F-1 Score
Defect	55,75	55,75	55,75	55,75
Longberry	77,00	71,13	77,00	73,95
Peaberry	57,00	59,38	57,00	58,16
Premium	56,25	58,75	56,25	57,47

Tablo 4.45 DenseNet201+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	62,20	62,25	62,20	62,19

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Eğitim doğruluk oranı %91,37 olarak hesaplanmıştır. Test veri setindeki örneklerin %62,20'si doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %62,25 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %62,20 olarak bulunmuştur. Son olarak, modelin F1 skoru %62,19 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise defect sınıfına aittir.



Şekil 4.15 DenseNet201+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.1.4.4. DenseNet201 Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.46 DenseNet201+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	350	9	18	23
	Longberry	6	382	2	10
	Peaberry	13	1	360	26
	Premium	22	13	15	350

Tablo 4.47 DenseNet201+SVM Modelinin Sınıf Bazında Performans Metrikleri

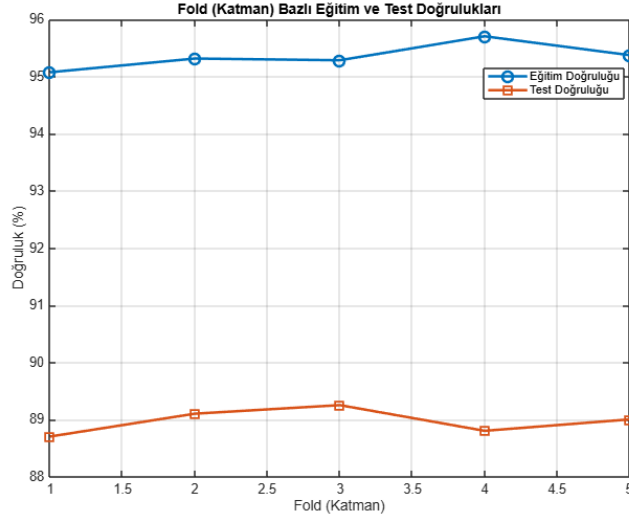
	Accuracy	Precision	Recall	F-1 Score
Defect	87,50	89,51	87,50	88,50
Longberry	95,50	94,32	95,50	94,91
Peaberry	90,00	91,14	90,00	90,57
Premium	87,50	85,57	87,50	86,53

Tablo 4.48 DenseNet201+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	88,95	88,98	88,95	88,96

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %95,36 iken, test doğruluk oranı %88,95 olarak hesaplanmıştır. Hassasiyet değeri ise %88,98 olarak tespit edilmiştir. Modelin duyarlılık değeri %89,95 olarak belirlenmiştir. F1 skoru ise %88,96 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında doğruluk oranlarının %85'in üstünde olduğu gözlemlenmektedir. En yüksek doğruluk oranı %95,50 ile longberry sınıfındayken, en düşük doğruluk oranı %82,25 ile premium ve defect sınıflarına aittir.



Şekil 4.16 DenseNet201+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2. CNN ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

4.2.1. MobileNetV2 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

MobileNetV2 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “mobilenetv2()” komutu ile önceden eğitilmiş olan MobileNetV2 modeli yüklenir. Daha sonra “activations” komutu kullanılarak MobileNetV2 modelinde bulunan “global_average_pooling2d_1” katmanında özellik çıkarımı yapılır. Bu katman MobileNetV2 modelinin tam bağlantılı katmanlarından biridir.

Daha sonrasında MobileNetV2 modeli ile çıkarılan özellikler standartlaştırılır. Verilere PCA uygulanarak varyansı %95’e eşit ve büyük bileşenler ana bileşen olarak seçilmektedir. Temel bileşen sayısı %95 varyansta ortalama 48 olarak bulunmuştur.

Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır.

4.2.1.1. MobileNetV2 ve PCA Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k 'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.49 MobileNetV2+PCA+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	66,80	68,46	66,80	66,19
k=3	70,41	71,47	70,41	69,82
k=5	71,20	73,25	71,20	70,40
k=7	71,86	74,68	71,86	70,89

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı %79,55, test doğruluk oranı %71,86 olarak belirlenmiştir. Modelin hassasiyet değeri %74,68 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %71,86 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 70,89 olarak bulunmuştur.

Bu kapsamda $k=7$ için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

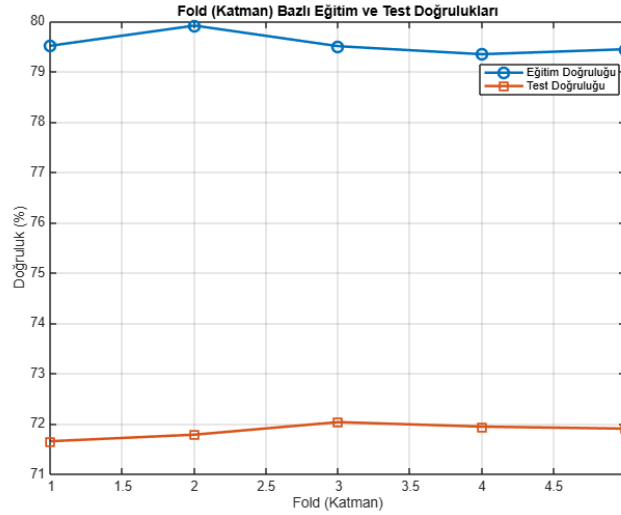
Tablo 4.50 MobileNetV2+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi ($k=7$)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	186	68	60	86
	Longberry	1	375	6	17
	Peaberry	10	42	270	78
	Premium	9	39	34	318

Tablo 4.51 MobileNetV2+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	46,60	90,39	46,60	61,49
Longberry	93,85	71,59	93,85	81,22
Peaberry	67,40	73,05	67,40	70,08
Premium	79,60	63,67	79,60	70,74

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.17 MobileNetV2+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.2.1.2. MobileNetV2 ve PCA Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.52 MobileNetV2+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	345	21	2	32
	Longberry	208	128	8	56
	Peaberry	274	51	11	64
	Premium	220	57	6	117

Tablo 4.53 MobileNetV2+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

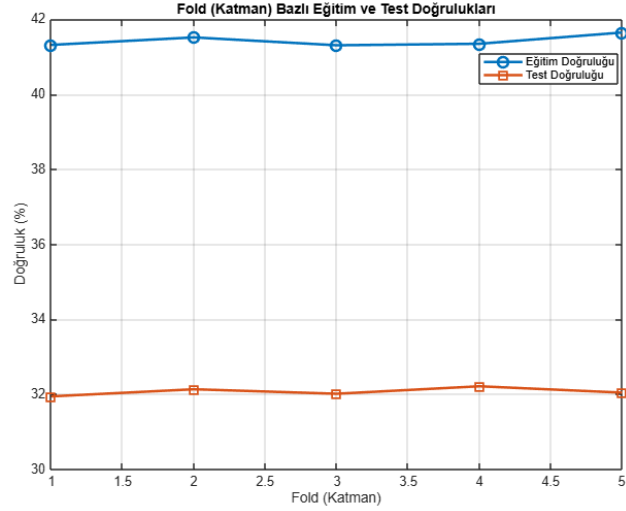
	Accuracy	Precision	Recall	F-1 Score
Defect	86,25	32,95	86,25	47,68
Longberry	32,00	49,81	32,00	38,96
Peaberry	2,75	40,74	2,75	5,15
Premium	29,25	43,49	29,25	34,98

Tablo 4.54 MobileNetV2+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	32,07	0,00	32,07	0,00

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %41,43 olarak hesaplanmıştır. Test veri setindeki örneklerin %32,07'sini doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %0,00 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %32,07 olarak bulunmuştur Son olarak, modelin F1 skoru %0,00 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin defect sınıfına, en düşük performans metrikleri ise peaberry sınıfına ait olduğu gözlenmektedir.



Şekil 4.18 MobileNetV2+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.1.3. MobileNetV2 ve PCA Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.55 MobileNetV2+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	208	53	66	73
	Longberry	49	273	39	39
	Peaberry	72	40	215	73
	Premium	78	37	65	220

Tablo 4.56 MobileNetV2+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

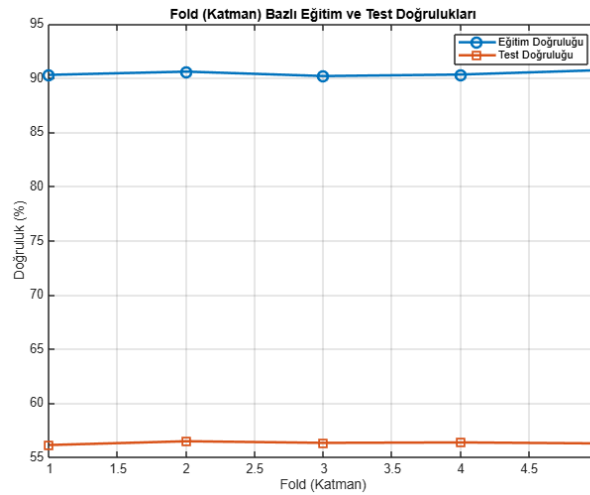
	Accuracy	Precision	Recall	F-1 Score
Defect	52,00	51,11	52,00	51,55
Longberry	68,25	67,74	68,25	68,00
Peaberry	53,75	55,84	53,75	54,78
Premium	55,00	54,32	55,00	54,66

Tablo 4.57 MobileNetV2+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	56,29	56,36	56,29	56,23

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %90,45 olarak hesaplanmıştır. Test veri setindeki örneklerin %56,29'u doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %56,36 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %56,29 olarak bulunmuştur. Son olarak, modelin F1 skoru %56,23 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise defect sınıfına aittir.

**Şekil 4.19** MobileNetV2+PCA+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.1.4. MobileNetV2 ve PCA Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.58 MobileNetV2+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	316	18	22	44
	Longberry	19	356	10	15
	Peaberry	29	10	329	32
	Premium	48	16	31	305

Tablo 4.59 MobileNetV2+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri

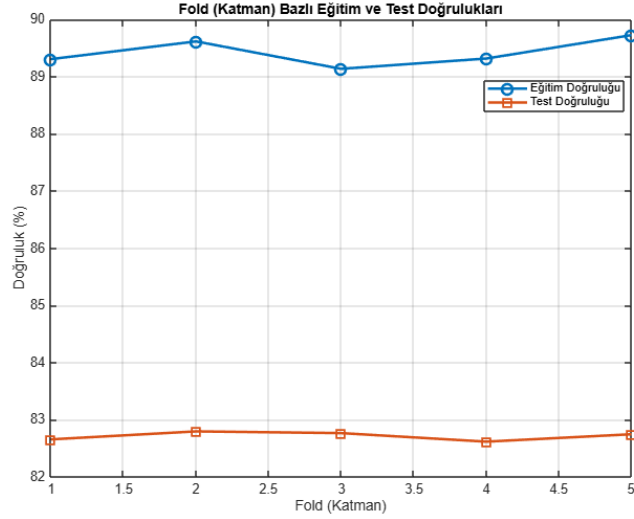
	Accuracy	Precision	Recall	F-1 Score
Defect	79,00	76,70	79,00	77,83
Longberry	89,00	89,00	89,00	88,00
Peaberry	82,25	83,93	82,25	83,08
Premium	76,25	77,02	76,25	76,63

Tablo 4.60 MobileNetV2+PCA+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	82,71	82,77	82,71	82,73

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %89,42 olarak bulunmuştur. Modelin test doğruluk oranı %82,71 olarak hesaplanmıştır. Hassasiyet değeri ise %82,77 olarak tespit edilmiştir. Modelin duyarlılık değeri %82,71 olarak belirlenmiştir. F1 skoru ise %82,73 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında en yüksek doğruluk oranı %89,00 ile longberry sınıfındayken, en düşük doğruluk oranı %76,25 ile premium sınıfına ait olduğu görülmektedir.



Şekil 4.20 MobileNetV2+PCA+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.2. ResNet18 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

ResNet18 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “resnet18()” komutu ile önceden eğitilmiş olan ResNet18 modeli yüklenir. Daha sonra “activations” komutu kullanılarak ResNet18 modelinde bulunan “pool5” katmanında özellik çıkarımı yapılır. Bu katman ResNet18 modelinin tam bağlantılı katmanlarından biridir.

Daha sonrasında ResNet18 modeli ile çıkarılan özellikler standartlaştırılır. Verilere PCA uygulanarak varyansı %95’e eşit ve büyük bileşenler ana bileşen olarak seçilmektedir. Temel bileşen sayısı %95 varyansta ortalama 241 olarak bulunmuştur.

Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır.

4.2.2.1. ResNet18 ve PCA Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.61 ResNet18+PCA+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	69,76	71,41	69,76	69,55
k=3	73,39	73,92	73,39	73,02
k=5	74,75	76,08	74,75	74,31
k=7	75,03	76,58	75,02	74,51

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı, 80,42, test doğruluk oranı %75,03 olarak belirlenmiştir. Modelin hassasiyet değeri %76,58 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %75,02 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 74,51 olarak bulunmuştur.

Bu kapsamda k=7 için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

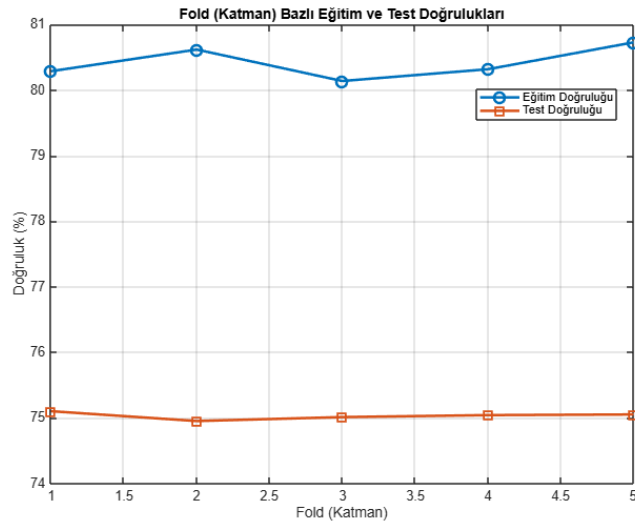
Tablo 4.62 ResNet18+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	224	46	68	62
	Longberry	1	379	5	15
	Peaberry	11	28	303	58
	Premium	12	38	55	295

Tablo 4.63 ResNet18+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	55,90	90,24	55,90	69,03
Longberry	94,80	77,03	94,80	84,99
Peaberry	75,65	70,37	75,65	72,90
Premium	73,75	68,68	73,75	71,12

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.21 ResNet18+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.2.2.2. ResNet18 ve PCA Tabanlı Özellik Çıkarımı ve Naive Bayes ile

Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.64 ResNet18+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	241	28	61	70
	Longberry	55	302	5	38
	Peaberry	61	8	254	77
	Premium	78	12	27	283

Tablo 4.65 ResNet18+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

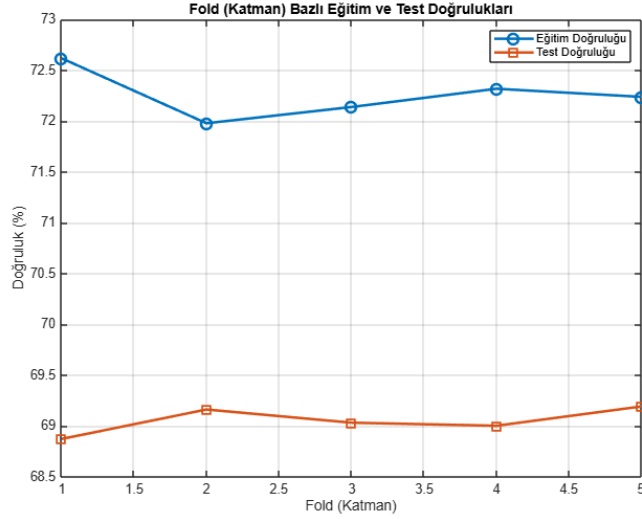
	Accuracy	Precision	Recall	F-1 Score
Defect	60,25	55,40	60,25	57,72
Longberry	75,50	86,29	75,50	80,53
Peaberry	63,50	73,20	63,50	68,01
Premium	70,75	60,47	70,75	65,21

Tablo 4.66 ResNet18+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	69,05	69,87	69,05	69,27

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %72,26 olarak hesaplanmıştır. Test veri setindeki örneklerin %69,05'ini doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %69,87 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %69,05 olarak bulunmuştur Son olarak, modelin F1 skoru %69,27 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına, en düşük performans metrikleri ise defect sınıfına ait olduğu gözlenmektedir.



Şekil 4.22 ResNet18+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.2.3. ResNet18 ve PCA Tabanlı Özellik Çıkarımı ve Karar Ağacı ile

Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.67 ResNet18+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	235	32	60	73
	Longberry	32	294	29	45
	Peaberry	55	28	206	111
	Premium	68	46	101	185

Tablo 4.68 ResNet18+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

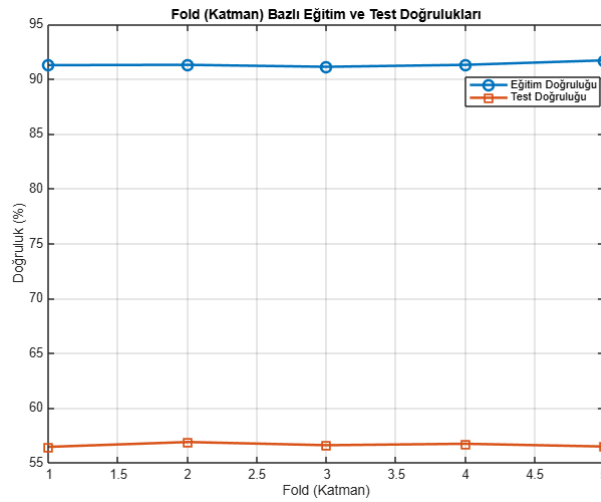
	Accuracy	Precision	Recall	F-1 Score
Defect	58,75	60,26	58,75	59,49
Longberry	73,50	73,50	73,50	73,50
Peaberry	51,50	52,02	51,50	51,76
Premium	46,25	44,69	46,25	45,45

Tablo 4.69 ResNet18+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	56,60	56,67	56,60	56,60

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %93,36 olarak hesaplanmıştır. Test veri setindeki örneklerin %56,60'sı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %56,67 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %56,60 olarak bulunmuştur. Son olarak, modelin F1 skoru %56,60 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.23 ResNet18+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.2.4. ResNet18 ve PCA Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.70 ResNet18+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	319	22	26	33
	Longberry	14	353	10	23
	Peaberry	29	2	334	35
	Premium	32	17	29	322

Tablo 4.71 ResNet18+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri

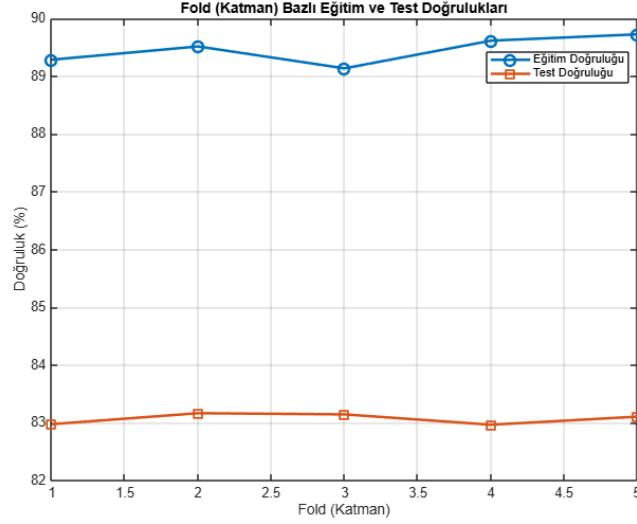
	Accuracy	Precision	Recall	F-1 Score
Defect	79,75	80,96	79,75	80,35
Longberry	88,25	89,59	88,25	88,92
Peaberry	83,50	83,71	83,50	83,60
Premium	80,50	77,97	80,50	79,21

Tablo 4.72 ResNet18+PCA+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	83,07	83,10	83,07	83,08

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %89,46, test doğruluk oranı %83,07 olarak hesaplanmıştır. Hassasiyet değeri ise %83,10 olarak tespit edilmiştir. Modelin duyarlılık değeri %83,07 olarak belirlenmiştir. F1 skoru ise %83,08 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında en yüksek doğruluk oranı %88,25 ile longberry sınıfındayken, en düşük doğruluk oranı %79,75 ile defect sınıfına ait olduğu görülmektedir.



Şekil 4.24 ResNet18+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.3. VGG16 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

VGG16 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “vgg16()” komutu ile önceden eğitilmiş olan VGG16 modeli yüklenir. Daha sonra “activations” komutu kullanılarak VGG16 modelinde bulunan “fc7” katmanında özellik çıkarımı yapılır. Bu katman VGG16 modelinin tam bağlantılı katmanlarından biridir.

Daha sonrasında VGG16 modeli ile çıkarılan özellikler standartlaştırılır. Verilere PCA uygulanarak varyansı %95'e eşit ve büyük bileşenler ana bileşen olarak seçilmektedir. Temel bileşen sayısı %95 varyansta ortalama 89 olarak bulunmuştur.

Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır.

4.2.3.1. VGG16 ve PCA Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.73 VGG16+PCA+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	69,50	70,91	69,50	69,36
k=3	72,86	73,28	72,86	72,58
k=5	74,20	75,23	74,20	73,90
k=7	75,02	76,36	75,02	74,74

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı %83,44, test doğruluk oranı %75,02 olarak belirlenmiştir. Modelin hassasiyet değeri %76,36 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %75,02 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 74,74 olarak bulunmuştur.

Bu kapsamda k=7 için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

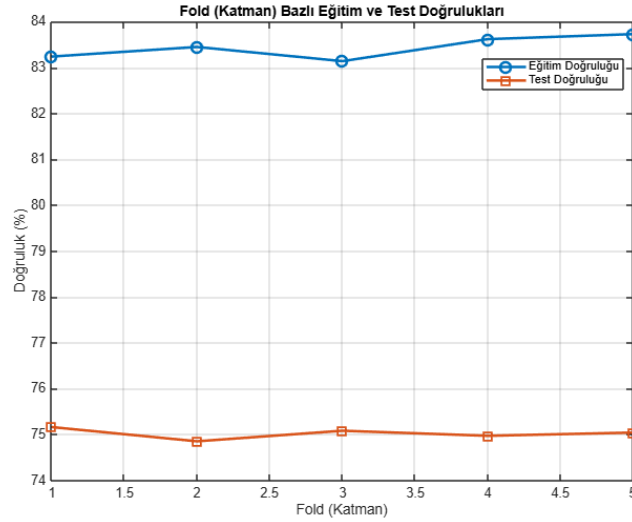
Tablo 4.74 VGG16+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	240	43	55	62
	Longberry	2	370	7	21
	Peaberry	16	32	287	65
	Premium	10	31	55	304

Tablo 4.75 VGG16+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	60,05	89,45	60,05	71,86
Longberry	92,40	77,65	92,40	84,38
Peaberry	71,70	71,03	71,70	71,35
Premium	75,95	67,30	75,95	71,36

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.25 VGG16+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.2.3.2. VGG16 ve PCA Tabanlı Özellik Çıkarımı ve Naive Bayes ile

Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.76 VGG16+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	275	19	48	58
	Longberry	44	292	14	50
	Peaberry	67	15	252	66
	Premium	70	13	31	286

Tablo 4.77 VGG16+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

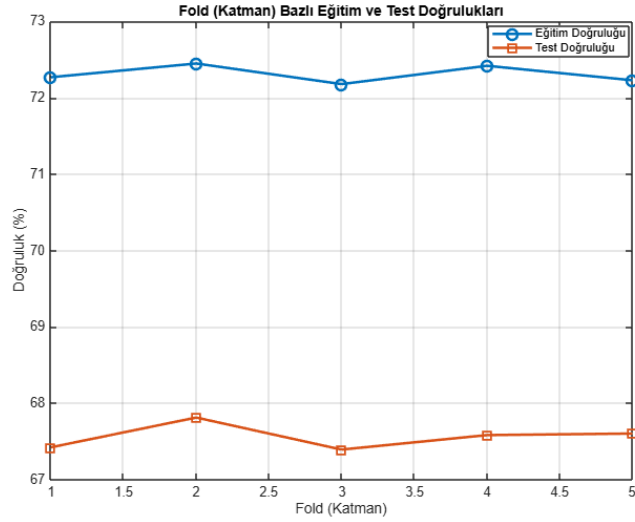
	Accuracy	Precision	Recall	F-1 Score
Defect	68,75	60,31	68,75	64,25
Longberry	73,00	86,14	73,00	79,03
Peaberry	63,00	73,04	63,00	67,65
Premium	71,50	62,17	71,50	66,51

Tablo 4.78 VGG16+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	67,56	69,11	67,56	67,93

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %72,31 olarak hesaplanmıştır. Test veri setindeki örneklerin %67,56'sını doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %69,11 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %67,56 olarak bulunmuştur Son olarak, modelin F1 skoru %67,93 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına, en düşük performans metrikleri ise peaberry sınıfına ait olduğu gözlenmektedir.



Şekil 4.26 VGG16+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.3.3. VGG16 ve PCA Tabanlı Özellik Çıkarımı ve Karar Ağacı ile

Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.79 VGG16+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	210	48	63	79
	Longberry	43	276	35	46
	Peaberry	74	33	207	86
	Premium	64	48	83	205

Tablo 4.80 VGG16+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

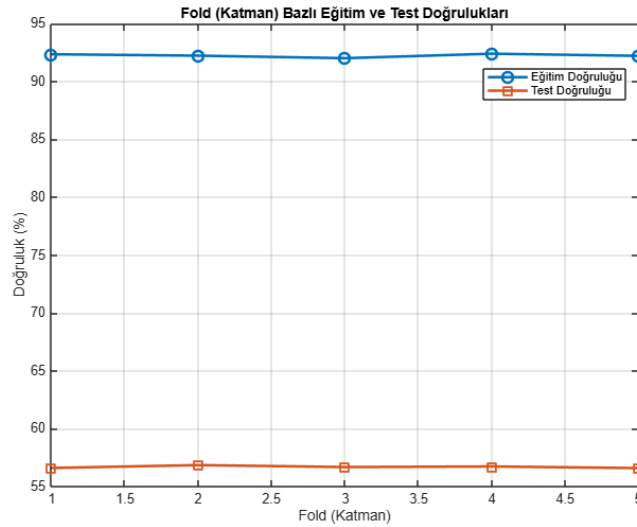
	Accuracy	Precision	Recall	F-1 Score
Defect	52,50	53,71	52,50	53,10
Longberry	69,00	68,15	69,00	68,57
Peaberry	51,75	53,35	51,75	52,54
Premium	51,25	49,28	51,25	50,25

Tablo 4.81 VGG16+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	56,66	56,77	56,66	56,69

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %92,26 olarak hesaplanmıştır. Test veri setindeki örneklerin %56,66'sı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %56,77 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %56,66 olarak bulunmuştur. Son olarak, modelin F1 skoru %56,69 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.27 VGG16+PCA+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.3.4. VGG16 ve PCA Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.82 VGG16+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	317	18	36	29
	Longberry	11	358	11	20
	Peaberry	31	12	325	32
	Premium	32	23	34	311

Tablo 4.83 VGG16+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri

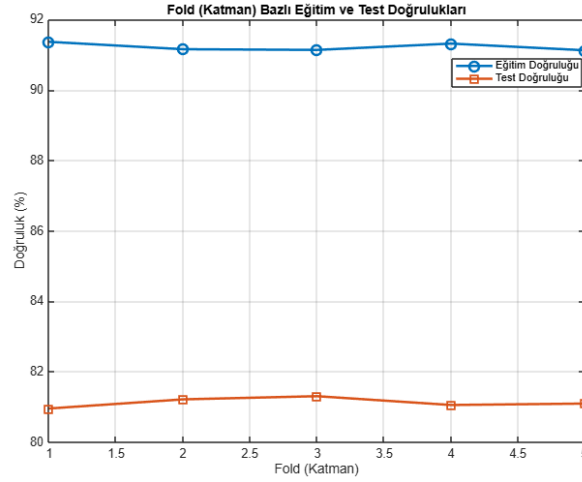
	Accuracy	Precision	Recall	F-1 Score
Defect	79,25	81,07	79,25	80,15
Longberry	89,50	87,10	89,50	88,29
Peaberry	81,25	80,05	81,25	80,65
Premium	77,75	79,34	77,75	78,54

Tablo 4.84 VGG16+PCA+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	81,12	81,14	81,12	81,09

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %91,22 iken, test doğruluk oranı %81,12 olarak hesaplanmıştır. Hassasiyet değeri ise %81,14 olarak tespit edilmiştir. Modelin duyarlılık değeri %81,12 olarak belirlenmiştir. F1 skoru ise %81,09 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında en yüksek doğruluk oranı %89,50 ile longberry sınıfındayken, en düşük doğruluk oranı %82,25 ile premium sınıfına ait olduğu görülmektedir.



Şekil 4.28 VGG16+PCA+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.4. DenseNet201 ve PCA Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

DenseNet201 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “densenet201()” komutu ile önceden eğitilmiş olan DenseNet201 modeli yüklenir. Daha sonra “activations” komutu kullanılarak DenseNet201 modelinde bulunan “avg_pool” katmanında özellik çıkarımı yapılır. Bu katman DenseNet201 modelinin havuzlama katmanlarından biridir. Çıkarılan özellikler oluşturulan boş matrislere eklenerek sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmalarında kullanılmaktadır

Daha sonrasında DenseNet201 modeli ile çıkarılan özellikler standartlaştırılır. Verilere PCA uygulanarak varyansı %95’e eşit ve büyük bileşenler ana bileşen olarak seçilmektedir. Ana bileşen sayısı %95 varyansta ortalama 525 olarak bulunmuştur.

Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır.

4.2.4.1. DenseNet201 ve PCA Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k’nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.85 DenseNet201+PCA+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	80,65	81,66	80,65	80,41
k=3	83,39	83,97	83,39	83,16
k=5	83,60	84,65	83,60	83,24
k=7	83,38	84,74	83,38	82,92

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 5 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı %90,32, test doğruluk oranı %83,60 olarak belirlenmiştir. Modelin hassasiyet değeri %84,65 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %83,60 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 83,24 olarak bulunmuştur.

Bu kapsamda k=5 için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

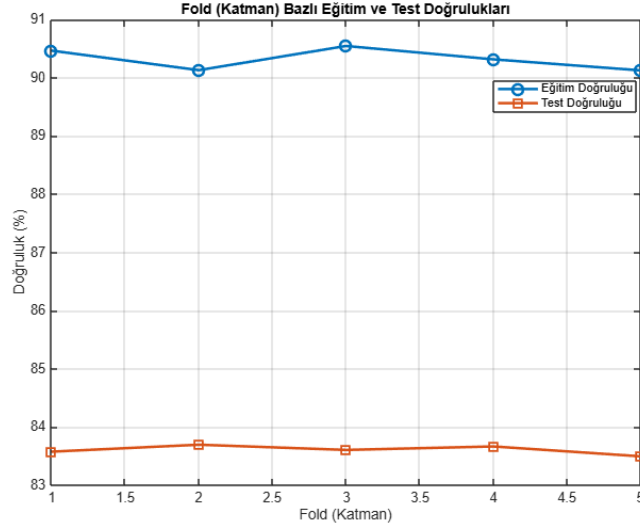
Tablo 4.86 DenseNet201+PCA+KNN ile Elde Edilen Karmaşıklık Matrisi (k=5)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	262	38	52	48
	Longberry	1	389	2	8
	Peaberry	4	16	350	30
	Premium	9	22	31	338

Tablo 4.87 DenseNet201+PCA+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=5)

	Accuracy	Precision	Recall	F-1 Score
Defect	65,40	95,12	65,40	77,49
Longberry	97,15	83,47	97,15	89,79
Peaberry	87,40	80,23	87,40	83,65
Premium	84,45	79,77	84,45	82,03

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.29 DenseNet201+PCA+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=5)

4.2.4.2. DenseNet201 ve PCA Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.88 DenseNet201+PCA+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	278	38	46	38
	Longberry	44	342	5	9
	Peaberry	63	45	242	50
	Premium	65	32	16	287

Tablo 4.89 DenseNet201+PCA+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

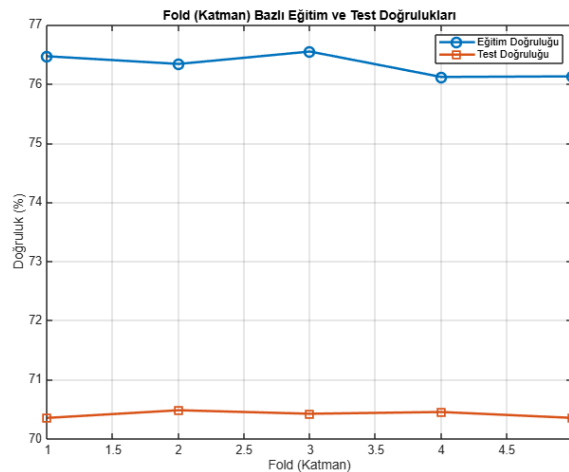
	Accuracy	Precision	Recall	F-1 Score
Defect	69,50	61,78	69,50	65,41
Longberry	85,50	74,84	85,50	79,81
Peaberry	60,50	78,32	60,50	68,27
Premium	71,75	74,74	71,75	73,21

Tablo 4.90 DenseNet201+PCA+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	70,41	70,70	70,41	70,27

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı 76,32 olarak hesaplanmıştır. Test veri setindeki örneklerin %70,41'i doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %70,70 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %70,41 olarak bulunmuştur. Son olarak, modelin F1 skoru %70,27 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek doğruluk oranının longberry sınıfına, en düşük doğruluk oranının ise peaberry sınıfına ait olduğu gözlenmektedir.



Şekil 4.30 DenseNet201+PCA+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk

Grafiği

4.2.4.3. DenseNet201 ve PCA Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.91 DenseNet201+PCA+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	235	51	51	63
	Longberry	31	296	39	34
	Peaberry	75	31	209	85
	Premium	84	45	56	215

Tablo 4.92 DenseNet201+PCA+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

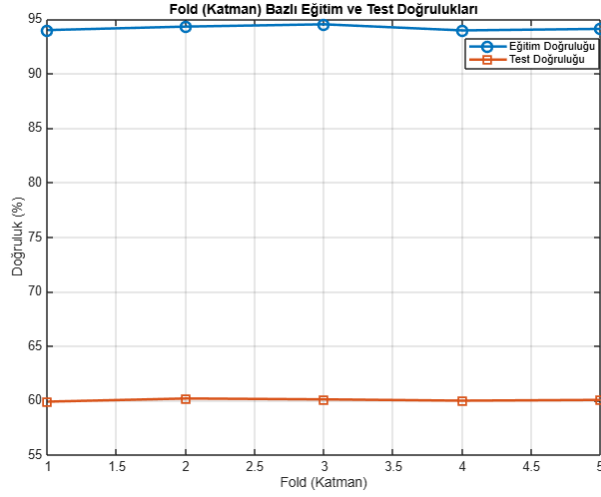
	Accuracy	Precision	Recall	F-1 Score
Defect	58,75	55,29	58,75	56,97
Longberry	74,00	69,98	74,00	71,93
Peaberry	52,25	58,87	52,25	55,36
Premium	53,75	54,16	53,75	53,95

Tablo 4.93 DenseNet201+PCA+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	60,02	60,04	60,02	60,00

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %94,21 olarak bulunmuştur. Test veri setindeki örneklerin %60,02'si doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %60,04 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %60,02 olarak bulunmuştur. Son olarak, modelin F1 skoru %60,00 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek doğruluk oranının longberry sınıfına ait olduğu gözlemlenmektedir. En düşük doğruluk oranı ise peaberry sınıfına aittir.



Şekil 4.31 DenseNet201+PCA+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.2.4.4. DenseNet201 ve PCA Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.94 DenseNet201+PCA+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	362	7	12	19
	Longberry	8	374	1	17
	Peaberry	14	2	370	14
	Premium	15	10	18	357

Tablo 4.95 DenseNet201+PCA+SVM Modelinin Sınıf Bazında Performans Metrikleri

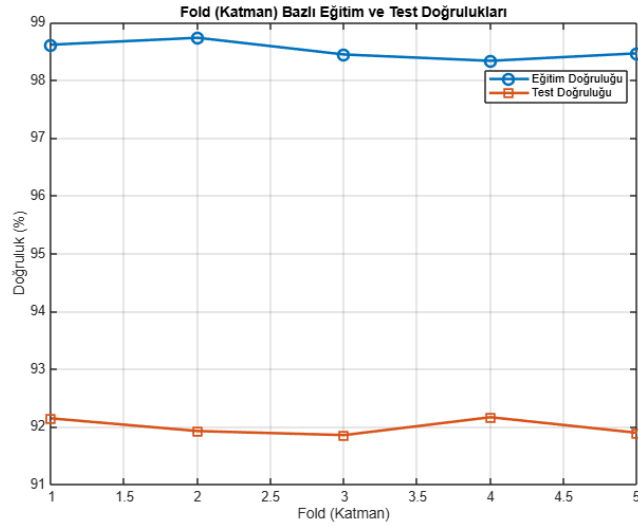
	Accuracy	Precision	Recall	F-1 Score
Defect	90,50	90,73	90,50	90,61
Longberry	93,50	95,17	93,50	94,33
Peaberry	92,50	92,27	92,50	92,38
Premium	89,25	87,71	89,25	88,48

Tablo 4.96 DenseNet201+PCA+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	91,99	92,01	91,99	91,99

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %98,52 iken, test doğruluk oranı %91,99 olarak hesaplanmıştır. Hassasiyet değeri ise %92,01 olarak tespit edilmiştir. Modelin duyarlılık değeri %91,99 olarak belirlenmiştir. F1 skoru ise %91,99 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında doğruluk oranlarının %89'un üstünde olduğu gözlemlenmektedir. En yüksek doğruluk oranı %93,50 ile longberry sınıfındayken, en düşük doğruluk oranı %89,25 ile premium sınıfına aittir.



Şekil 4.32 DenseNet201+PCA+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3. CNN ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

4.3.1. MobileNetV2 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

MobileNetV2 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “mobilenetv2()” komutu ile önceden eğitilmiş olan MobileNetV2 modeli yüklenir. Daha sonra “activations” komutu kullanılarak MobileNetV2 modelinde bulunan “global_average_pooling2d_1” katmanında özellik çıkarımı yapılır. Bu katman MobileNetV2 modelinin tam bağlantılı katmanlarından biridir.

Daha sonrasında görüntülerin HoG özellikleri çıkarılarak MobileNetV2 ile çıkarılan özelliklerle birleştirilir. Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır. MobileNetV2 ile yapılan bu çalışmada her bir görüntüye ait 27524 özellik kullanılmaktadır.

4.3.1.1. MobileNetV2 ve HoG Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.97 MobileNetV2+HoG+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	67,19	68,49	67,19	66,64
k=3	70,30	70,98	70,30	69,73
k=5	71,65	73,06	71,65	70,93
k=7	71,84	73,85	71,84	71,06

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluğu, 80,55, test doğruluk oranı %71,84 olarak belirlenmiştir. Modelin hassasiyet

değeri %73,85 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %71,84 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 71,06 olarak bulunmuştur.

Bu kapsamda $k=7$ için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

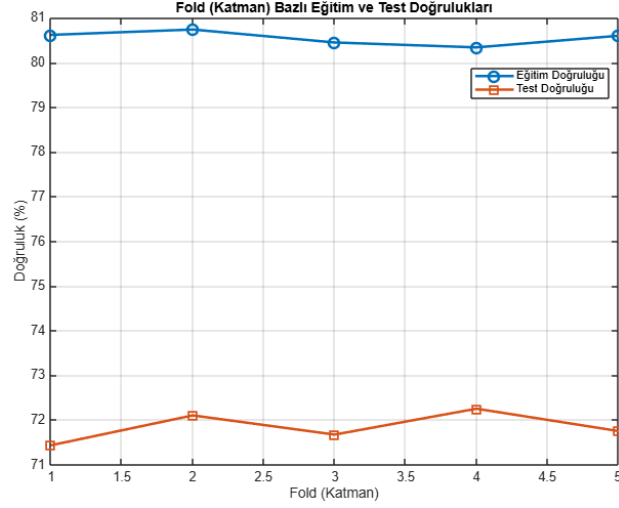
Tablo 4.98 MobileNetV2+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi ($k=7$)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	200	64	55	81
	Longberry	4	372	6	18
	Peaberry	13	51	271	65
	Premium	13	39	41	307

Tablo 4.99 MobileNetV2+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri ($k=7$)

	Accuracy	Precision	Recall	F-1 Score
Defect	49,90	86,78	49,90	63,32
Longberry	93,05	70,64	93,05	80,29
Peaberry	67,65	72,75	67,65	70,09
Premium	76,75	65,24	76,75	70,53

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.33 MobileNetV2+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.3.1.2. MobileNetV2 ve HoG Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.100 MobileNetV2+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

	Predicted Class				
	Defect	Longberry	Peaberry	Premium	
Actual Class	Defect	335	15	10	40
Longberry	194	112	37	57	
Peaberry	222	30	52	96	
Premium	189	29	22	160	

Tablo 4.101 MobileNetV2+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

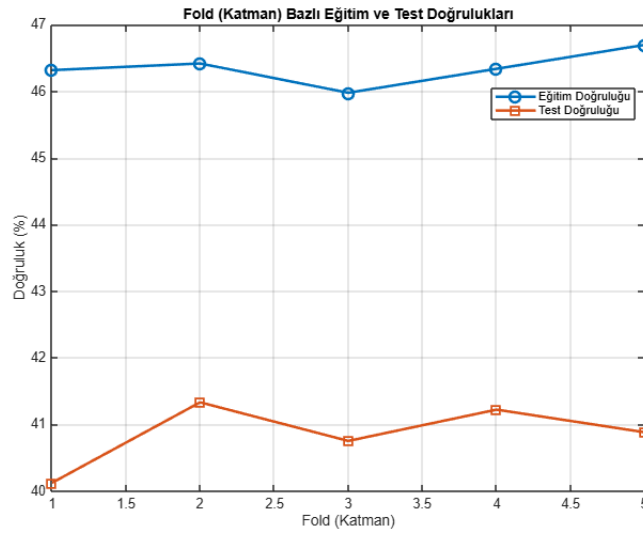
	Accuracy	Precision	Recall	F-1 Score
Defect	83,75	35,64	83,75	50,00
Longberry	28,00	60,22	28,00	38,23
Peaberry	13,00	42,98	13,00	19,96
Premium	40,00	45,33	40,00	42,50

Tablo 4.102 MobileNetV2+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	40,86	47,31	40,86	38,26

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluğu %46,35 olarak hesaplanmıştır. Test veri setindeki örneklerin %40,86'sını doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %47,31 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %40,86 olarak bulunmuştur Son olarak, modelin F1 skoru %38,26 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin defect sınıfına, en düşük performans metrikleri ise peaberry sınıfına ait olduğu gözlenmektedir.



Şekil 4.34 MobileNetV2+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.1.3. MobileNetV2 ve HoG Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.103 MobileNetV2+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	191	54	82	73
	Longberry	48	261	42	49
	Peaberry	65	52	203	80
	Premium	81	54	71	194

Tablo 4.104 MobileNetV2+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

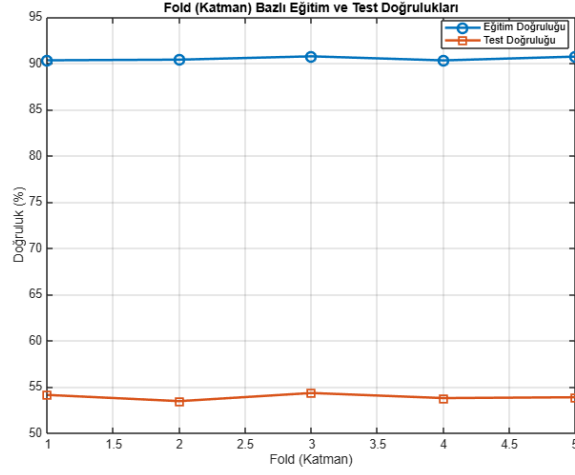
	Accuracy	Precision	Recall	F-1 Score
Defect	47,75	49,61	47,75	48,66
Longberry	65,25	62,00	65,25	63,58
Peaberry	50,75	51,01	50,75	50,88
Premium	48,50	48,99	48,50	48,74

Tablo 4.105 MobileNetV2+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	53,90	53,89	53,90	53,84

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelde eğitim doğruluk oranı %90,53 olarak bulunmuştur. Test veri setindeki örneklerin %53,90'ı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %53,86 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %53,90 olarak bulunmuştur. Son olarak, modelin F1 skoru %53,84 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise defect sınıfına aittir.



Şekil 4.35 MobileNetV2+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.1.4. MobileNetV2 ve HoG Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.106 MobileNetV2+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	316	24	20	40
	Longberry	5	371	5	19
	Peaberry	19	6	343	32
	Premium	22	14	35	329

Tablo 4.107 MobileNetV2+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri

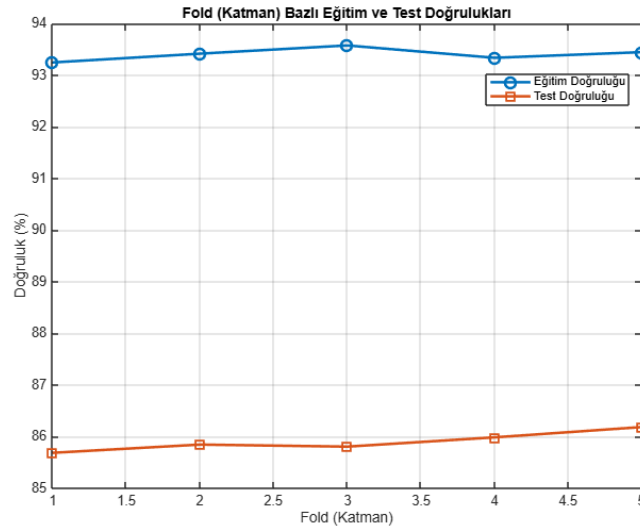
	Accuracy	Precision	Recall	F-1 Score
Defect	79,00	87,29	79,00	82,94
Longberry	92,75	89,40	92,75	91,04
Peaberry	85,75	85,11	85,75	85,43
Premium	82,25	78,33	82,24	80,24

Tablo 4.108 MobileNetV2+HoG+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	85,86	85,92	85,86	85,85

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %93,41, test doğruluk oranı %85,86 olarak hesaplanmıştır. Hassasiyet değeri ise %85,92 olarak tespit edilmiştir. Modelin duyarlılık değeri %85,86 olarak belirlenmiştir. F1 skoru ise %85,85 olarak bulunmuştur.

En yüksek doğruluk oranı %92,75 ile longberry sınıfındayken, en düşük doğruluk oranı %79,00 ile defect sınıfına aittir.



Şekil 4.36 MobileNetV2+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.2. ResNet18 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

ResNet18 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “resnet18()” komutu ile önceden eğitilmiş olan ResNet18 modeli yüklenir. Daha sonra “activations” komutu kullanılarak ResNet18 modelinde bulunan

“pool5” katmanında özellik çıkarımı yapılır. Bu katman ResNet18 modelinin tam bağlantılı katmanlarından biridir.

Daha sonrasında görüntülerin HoG özellikleri çıkarılır ResNet18 ile çıkarılan özelliklerle birleştirilir. Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır. VGG16 ile yapılan bu çalışmada her bir görüntüye ait 26756 özellik kullanılmaktadır.

4.3.2.1. ResNet18 ve HoG Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.109 ResNet18+HoG+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	66,25	67,77	66,25	65,71
k=3	69,91	70,40	69,91	69,39
k=5	71,74	73,34	71,74	71,13
k=7	72,20	74,34	72,20	71,50

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı %79,26, test doğruluk oranı %72,20 olarak belirlenmiştir. Modelin hassasiyet değeri %74,34 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %72,20 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 71,50 olarak bulunmuştur.

Bu kapsamda k=7 için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

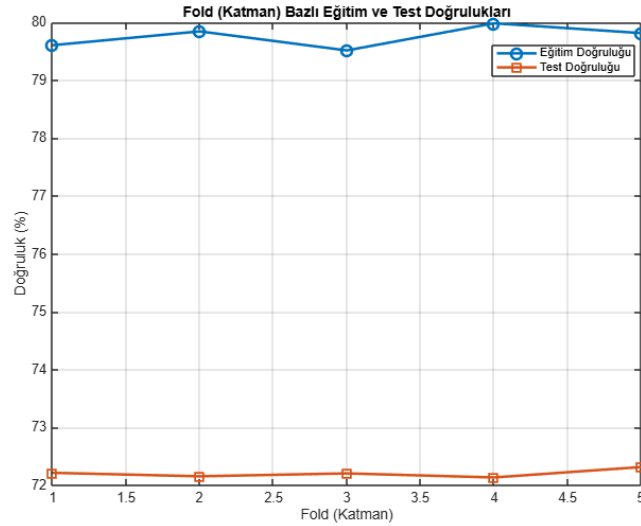
Tablo 4.110 ResNet18+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	204	59	57	80
	Longberry	3	374	4	19
	Peaberry	12	43	274	71
	Premium	10	40	47	303

Tablo 4.111 ResNet18+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	51,10	88,95	51,10	64,85
Longberry	93,15	72,03	93,15	81,22
Peaberry	67,00	70,77	67,00	68,80
Premium	73,50	64,54	73,50	68,72

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.

**Şekil 4.37** ResNet18+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.3.2.2. ResNet18 ve HoG Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.112 ResNet18+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	240	34	85	41
	Longberry	71	258	45	26
	Peaberry	115	26	204	55
	Premium	111	68	95	126

Tablo 4.113 ResNet18+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

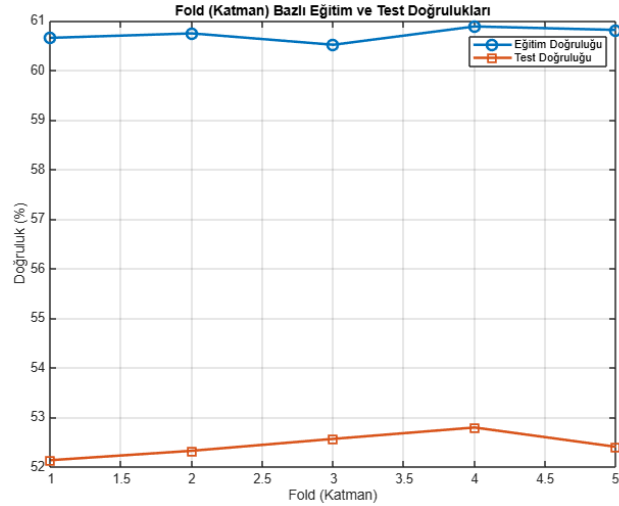
	Accuracy	Precision	Recall	F-1 Score
Defect	60,00	44,69	60,00	51,23
Longberry	64,50	66,84	64,50	65,65
Peaberry	51,00	47,55	51,00	49,22
Premium	31,50	50,81	31,50	38,89

Tablo 4.114 ResNet18+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	52,45	54,19	52,45	52,26

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %60,73 olarak bulunmuştur. Test veri setindeki örneklerin %52,45'ini doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %54,19 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %52,45 olarak bulunmuştur Son olarak, modelin F1 skoru %52,26 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına, en düşük performans metrikleri ise premium sınıfına ait olduğu gözlenmektedir.



Şekil 4.38 ResNet18+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.2.3. ResNet18 ve HoG Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.115 ResNet18+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	207	53	68	72
	Longberry	32	287	41	40
	Peaberry	85	45	189	81
	Premium	78	54	93	175

Tablo 4.116 ResNet18+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

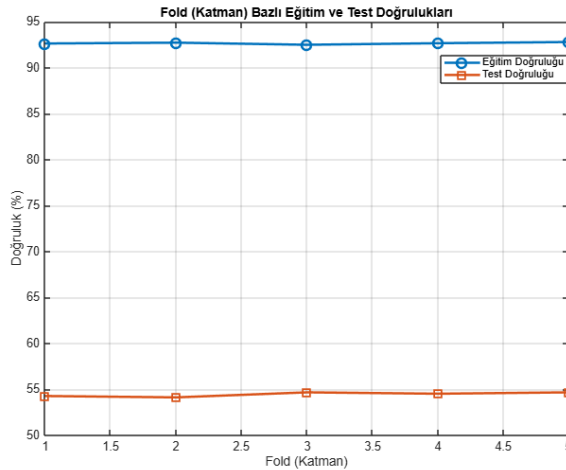
	Accuracy	Precision	Recall	F-1 Score
Defect	51,75	51,49	51,75	51,62
Longberry	71,75	65,38	71,75	68,41
Peaberry	47,25	48,34	47,25	47,79
Premium	43,75	47,55	43,75	45,57

Tablo 4.117 ResNet18+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	54,43	54,37	54,43	54,37

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %92,69 olarak hesaplanmıştır. Test veri setindeki örneklerin %54,43'ü doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %54,37 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %54,43 olarak bulunmuştur. Son olarak, modelin F1 skoru %54,37 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.39 ResNet18+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.2.4. ResNet18 ve HoG Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.118 ResNet18+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	327	13	29	31
	Longberry	5	373	4	18
	Peaberry	15	5	344	36
	Premium	31	11	19	339

Tablo 4.119 ResNet18+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri

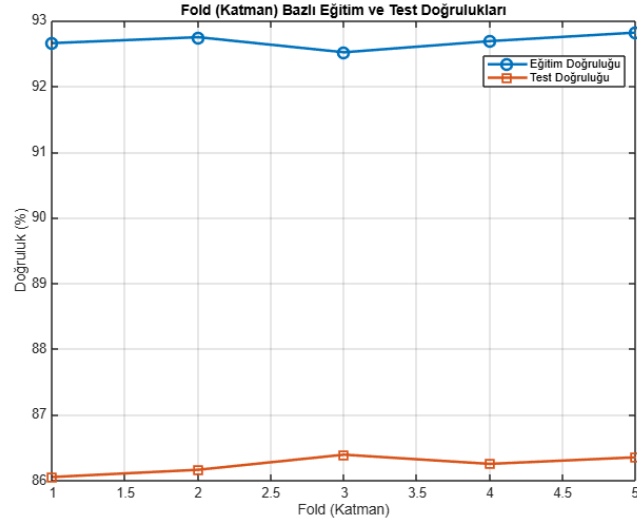
	Accuracy	Precision	Recall	F-1 Score
Defect	81,75	86,51	81,75	84,06
Longberry	93,25	92,79	93,25	93,02
Peaberry	86,00	86,67	86,00	86,43
Premium	84,75	79,95	84,75	82,28

Tablo 4.120 ResNet18+HoG+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	86,24	86,26	86,24	86,23

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %92,69, test doğruluk oranı %86,24 olarak hesaplanmıştır. Hassasiyet değeri ise %86,26 olarak tespit edilmiştir. Modelin duyarlılık değeri %86,24 olarak belirlenmiştir. F1 skoru ise %86,23 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında doğruluk oranlarının %80'in üstünde olduğu gözlemlenmektedir. En yüksek doğruluk oranı %93,25 ile longberry sınıfındayken, en düşük doğruluk oranı %81,75 ile defect sınıfına aittir.



Şekil 4.40 ResNet18+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.3. VGG16 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

VGG16 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “vgg16()” komutu ile önceden eğitilmiş olan VGG16 modeli yüklenir. Daha sonra “activations” komutu kullanılarak VGG16 modelinde bulunan “fc7” katmanında özellik çıkarımı yapılır. Bu katman VGG16 modelinin tam bağlantılı katmanlarından biridir.

Daha sonrasında görüntülerin HoG özellikleri çıkarılır VGG16 ile çıkarılan özelliklerle birleştirilir. Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır. VGG16 ile yapılan bu çalışmada her bir görüntüye ait 30337 özellik kullanılmaktadır.

4.3.3.1. VGG16 ve HoG Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.121 VGG16+HoG+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	70,59	72,24	70,59	70,35
k=3	72,76	73,29	72,76	72,41
k=5	74,24	75,37	74,24	73,90
k=7	74,76	76,19	74,76	74,42

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 7 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı %83,69, test doğruluk oranı %74,76 olarak belirlenmiştir. Modelin hassasiyet değeri %76,19 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %74,76 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 74,42 olarak bulunmuştur.

Bu kapsamda k=7 için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

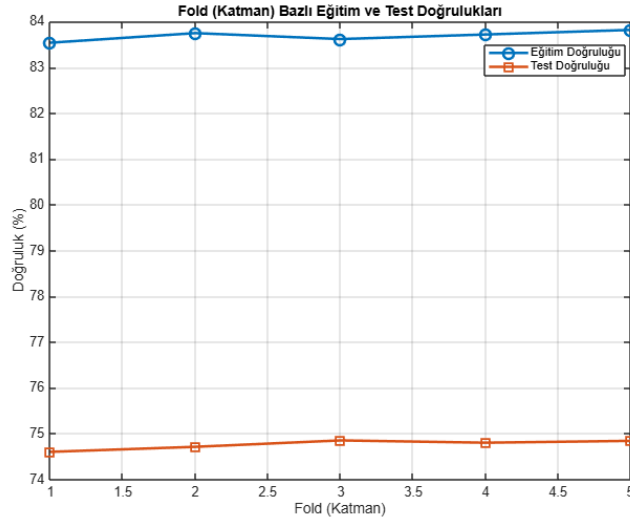
Tablo 4.122 VGG16+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi (k=7)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	234	44	55	67
	Longberry	4	370	6	20
	Peaberry	16	35	287	63
	Premium	11	31	54	304

Tablo 4.123 VGG16+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri (k=7)

	Accuracy	Precision	Recall	F-1 Score
Defect	58,60	89,24	58,60	70,73
Longberry	92,60	77,04	92,60	84,10
Peaberry	71,85	71,45	71,85	71,62
Premium	76,00	67,04	76,00	71,22

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.41 VGG16+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği (k=7)

4.3.3.2. VGG16 ve HoG Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.124 VGG16+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	232	33	59	76
	Longberry	14	310	40	36
	Peaberry	60	29	240	71
	Premium	45	43	93	219

Tablo 4.125 VGG16+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

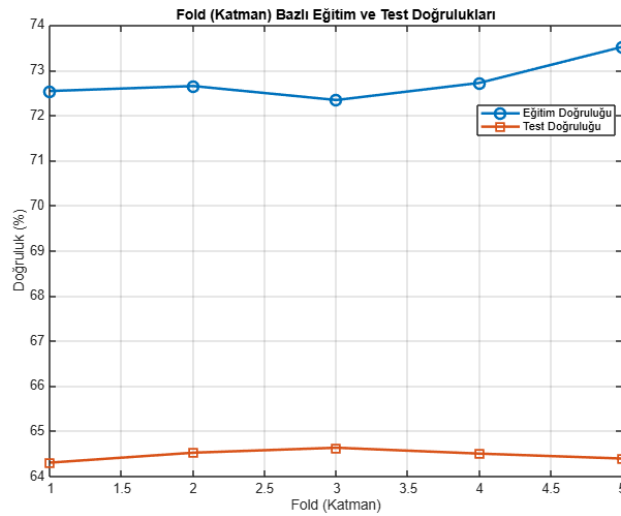
	Accuracy	Precision	Recall	F-1 Score
Defect	58,00	66,10	58,00	61,78
Longberry	77,50	74,70	77,50	76,07
Peaberry	60,00	55,56	60,00	57,69
Premium	54,75	54,48	54,75	54,61

Tablo 4.126 VGG16+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	64,47	63,72	63,48	63,30

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı 72,75 olarak hesaplanmıştır. Test veri setindeki örneklerin %64,47'sini doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %63,72 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %63,48 olarak bulunmuştur. Son olarak, modelin F1 skoru %63,30 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına, en düşük performans metrikleri ise premium sınıfına ait olduğu gözlenmektedir.

**Şekil 4.42** VGG16+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.3.3. VGG16 ve HoG Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.127 VGG16+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	239	25	68	68
	Longberry	41	285	34	40
	Peaberry	75	28	213	84
	Premium	68	46	75	211

Tablo 4.128 VGG16+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

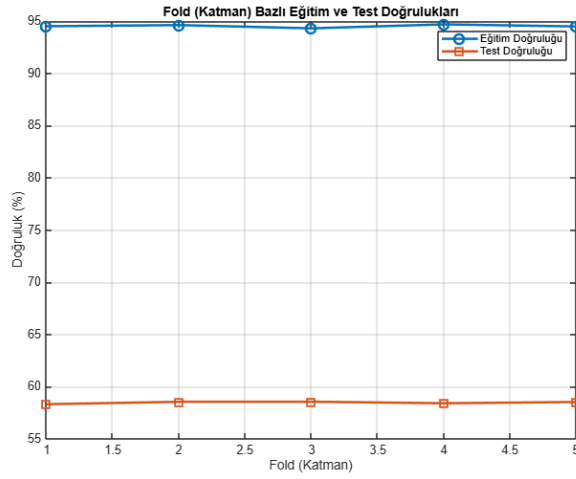
	Accuracy	Precision	Recall	F-1 Score
Defect	59,75	56,00	59,75	58,08
Longberry	71,25	74,22	71,25	72,70
Peaberry	53,25	54,62	53,25	53,92
Premium	52,75	52,36	52,75	52,55

Tablo 4.129 VGG16+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	58,44	58,51	58,44	58,43

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %94,55 olarak hesaplanmıştır. Test veri setindeki örneklerin %58,4'ü doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %58,41 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %58,44 olarak bulunmuştur. Son olarak, modelin F1 skoru %58,43 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek performans metriklerin longberry sınıfına ait olduğu gözlemlenmektedir. En düşük performans metrikleri ise premium sınıfına aittir.



Şekil 4.43 VGG16+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.3.4. VGG16 ve HoG Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.130 VGG16+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	330	8	24	38
	Longberry	10	364	5	21
	Peaberry	19	7	356	18
	Premium	18	14	26	342

Tablo 4.131 VGG16+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri

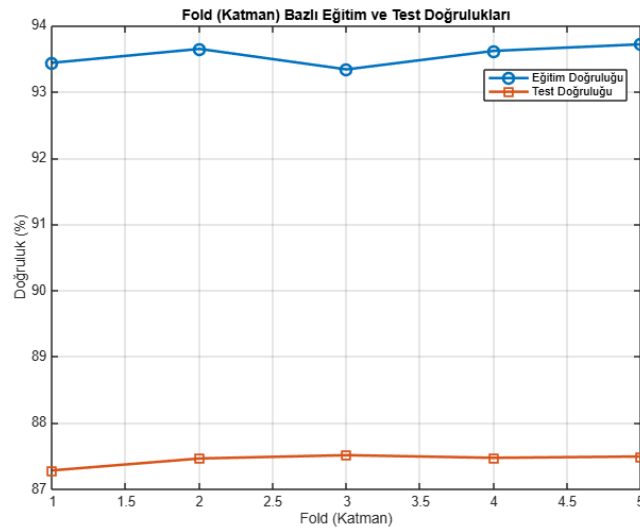
	Accuracy	Precision	Recall	F-1 Score
Defect	82,50	87,53	82,50	84,94
Longberry	91,00	92,62	91,00	91,80
Peaberry	89,00	86,62	89,00	87,79
Premium	85,50	87,47	87,44	87,43

Tablo 4.132 VGG16+HoG+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	87,44	87,47	87,44	87,43

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin eğitim doğruluk oranı %93,55, test doğruluk oranı %87,44 olarak hesaplanmıştır. Hassasiyet değeri ise %87,47 olarak tespit edilmiştir. Modelin duyarlılık değeri %87,44 olarak belirlenmiştir. F1 skoru ise %87,43 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında doğruluk oranlarının %80'in üstünde olduğu gözlemlenmektedir. En yüksek doğruluk oranı %91,00 ile longberry sınıfındayken, en düşük doğruluk oranı %82,50 ile defect sınıfına aittir.



Şekil 4.44 VGG16+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.4. DenseNet201 ve HoG Tabanlı Özellik Çıkarımı ve Makine Öğrenmesi Algoritmaları ile Veri Seti Sınıflandırması

DenseNet201 modeli ile özellik çıkarımı yapmak için bir fonksiyon oluşturulur. Bu fonksiyonda ilk olarak “densenet201()” komutu ile önceden eğitilmiş olan DenseNet201 modeli yüklenir. Daha sonra “activations” komutu kullanılarak DenseNet201 modelinde

bulunan “avg_pool” katmanında özellik çıkarımı yapılır. Bu katman DenseNet201 modelinin havuzlama katmanlarından biridir. Çıkarılan özellikler oluşturulan boş matrislere eklenerek sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmalarında kullanılmaktadır

Daha sonra “extractHOGFeatures” komutu kullanılarak görüntülerin HoG özellikleri çıkartılır. Burada her görüntü 8x8 boyutundaki hücreler ayrılmaktadır. HoG, her hücrede gradyan yönelimlerinin histogramını oluşturur. Oluşturulan histogramlar birşere görüntüyü temsil eden bir özellik vektörüne dönüştürülür.

Son olarak çıkarılan özellikler doğrultusunda veri setine sırasıyla KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmaktadır. DenseNet201 ile yapılan bu çalışmada her bir görüntüye ait 28164 özellik kullanılmaktadır.

4.3.4.1. DenseNet201 ve HoG Tabanlı Özellik Çıkarımı ve KNN ile Sınıflandırma

Bu uygulamada k'nın sırasıyla 1, 3, 5 ve 7 olan değerleri için ortalama performans metrikleri aşağıdaki tabloda belirtilmiştir.

Tablo 4.133 DenseNet201+HoG+KNN Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
k=1	72,80	74,31	72,80	72,47
k=3	75,79	76,82	75,79	75,45
k=5	77,01	78,67	77,01	76,66
k=7	76,80	78,67	76,80	76,37

K-en yakın komşu (KNN) algoritması ile gerçekleştirilen sınıflandırma sürecinde, en yüksek performans metrikleri k değeri 5 için elde edilmiştir. Bu bağlamda, modelin eğitim doğruluk oranı %84,54, test doğruluk oranı %77,01 olarak belirlenmiştir. Modelin hassasiyet değeri %78,67 olarak hesaplanmıştır. Bir başka performans metriği olan duyarlılık değeri %77,01 olarak tespit edilmiştir. Son olarak, modelin F1 skoru 76,66 olarak bulunmuştur.

Bu kapsamda $k=5$ için bulunan karmaşıklık matrisi ve sınıf bazında performans metrikleri aşağıdaki tablolardaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

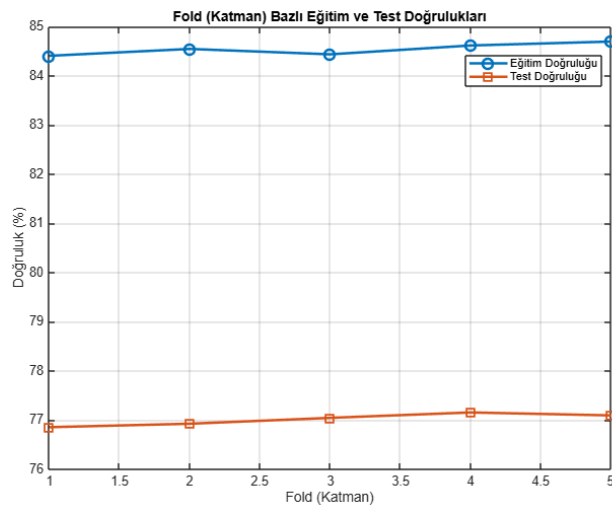
Tablo 4.134 DenseNet201+HoG+KNN ile Elde Edilen Karmaşıklık Matrisi ($k=5$)

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	245	57	45	53
	Longberry	2	382	4	12
	Peaberry	9	46	307	38
	Premium	12	46	44	298

Tablo 4.135 DenseNet201+HoG+KNN Modelinin Sınıf Bazında Performans Metrikleri ($k=5$)

	Accuracy	Precision	Recall	F-1 Score
Defect	61,25	91,74	61,25	73,43
Longberry	95,60	72,09	95,60	82,19
Peaberry	76,70	76,66	76,70	76,67
Premium	74,50	74,20	74,50	74,33

Performans metrikleri sınıf bazında incelendiğinde longberry sınıfının doğruluk oranının diğer sınıflara göre daha yüksek, defect sınıfının ise daha düşük olduğu gözlemlenmektedir.



Şekil 4.45 DenseNet201+HoG+KNN Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

($k=5$)

4.3.4.2. DenseNet201 ve HoG Tabanlı Özellik Çıkarımı ve Naive Bayes ile Sınıflandırma

Naive Bayes algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.136 DenseNet201+HoG+Naive Bayes ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	200	45	93	62
	Longberry	19	322	50	9
	Peaberry	82	34	217	67
	Premium	50	35	132	183

Tablo 4.137 DenseNet201+HoG+Naive Bayes Modelinin Sınıf Bazında Performans Metrikleri

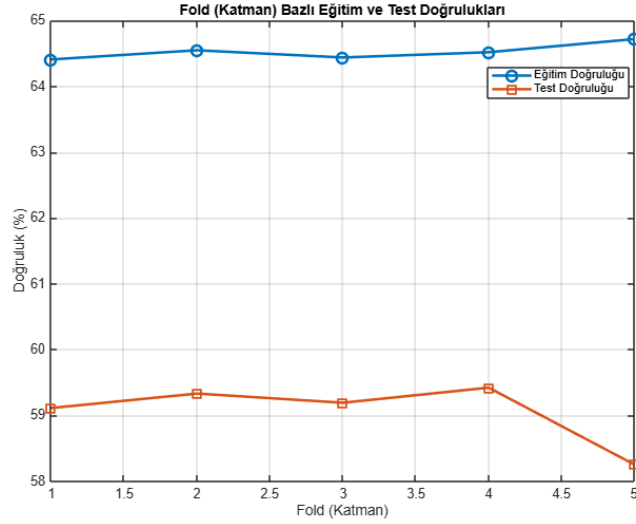
	Accuracy	Precision	Recall	F-1 Score
Defect	50,00	56,98	50,00	53,26
Longberry	80,50	74,85	80,50	77,03
Peaberry	54,25	44,11	54,25	48,65
Premium	45,75	57,01	45,75	50,76

Tablo 4.138 DenseNet201+HoG+Naive Bayes Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Naive Bayes	59,26	59,82	59,85	59,08

Naive Bayes algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı %64,53 olarak bulunmuştur. Test veri setindeki örneklerin %59,26'sı doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %59,82 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %59,85 olarak bulunmuştur. Son olarak, modelin F1 skoru %59,08 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek doğruluk oranının longberry sınıfına, en düşük doğruluk oranının ise premium sınıfına ait olduğu gözlenmektedir.



Şekil 4.46 DenseNet201+HoG+Naive Bayes Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.4.3. DenseNet201 ve HoG Tabanlı Özellik Çıkarımı ve Karar Ağacı ile Sınıflandırma

Karar ağacı algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.139 DenseNet201+HoG+Karar Ağacı ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	203	43	92	62
	Longberry	22	316	48	14
	Peaberry	67	25	238	70
	Premium	64	39	112	185

Tablo 4.140 DenseNet201+HoG+Karar Ağacı Modelinin Sınıf Bazında Performans Metrikleri

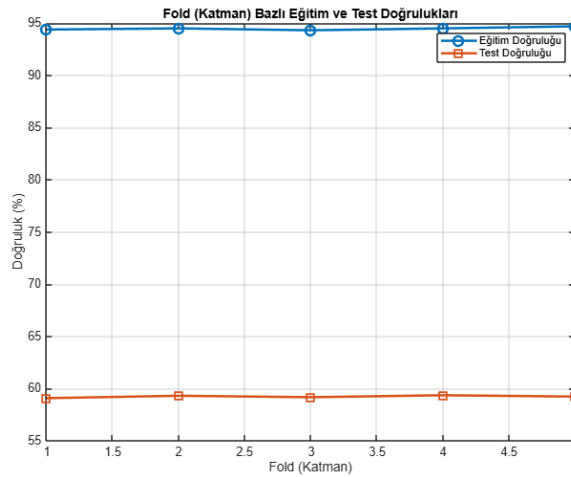
	Accuracy	Precision	Recall	F-1 Score
Defect	50,75	57,02	50,75	53,70
Longberry	79,00	74,70	79,00	76,79
Peaberry	59,50	48,57	59,50	53,48
Premium	46,25	55,89	46,25	50,62

Tablo 4.141 DenseNet201+HoG+Karar Ağacı Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
Decision Tree	59,20	59,77	59,20	59,02

Karar ağacı algoritması veri setine uygulandığında elde edilen performans metrikleri, doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Modelin eğitim doğruluk oranı 94,50 olarak bulunmuştur. Test veri setindeki örneklerin %59,20'si doğru bir şekilde sınıflandırılmıştır. Hassasiyet değeri %59,77 olarak hesaplanmıştır. Modelin duyarlılık değeri ise %59,20 olarak bulunmuştur. Son olarak, modelin F1 skoru %59,02 olarak tespit edilmiştir.

Sınıf bazında performans metriklerine göz atıldığı zaman en yüksek doğruluk oranının longberry sınıfına ait olduğu gözlemlenmektedir. En düşük doğruluk oranı ise premium sınıfına aittir.



Şekil 4.47 DenseNet201+HoG+Karar Ağacı Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

4.3.4.4. DenseNet201 ve HoG Tabanlı Özellik Çıkarımı ve SVM ile Sınıflandırma

SVM algoritmasına ait karmaşıklık matrisi ve performans metriklerine ait tablolar aşağıdaki gibidir. Ek olarak uygulamaya ait her bir katman için eğitim ve test doğruluklarını gösteren grafik aşağıdaki şekilde yer almaktadır.

Tablo 4.142 DenseNet201+HoG+SVM ile Elde Edilen Karmaşıklık Matrisi

		Predicted Class			
		Defect	Longberry	Peaberry	Premium
Actual Class	Defect	358	4	17	21
	Longberry	3	385	2	10
	Peaberry	8	1	365	26
	Premium	23	14	11	352

Tablo 4.143 DenseNet201+HoG+SVM Modelinin Sınıf Bazında Performans Metrikleri

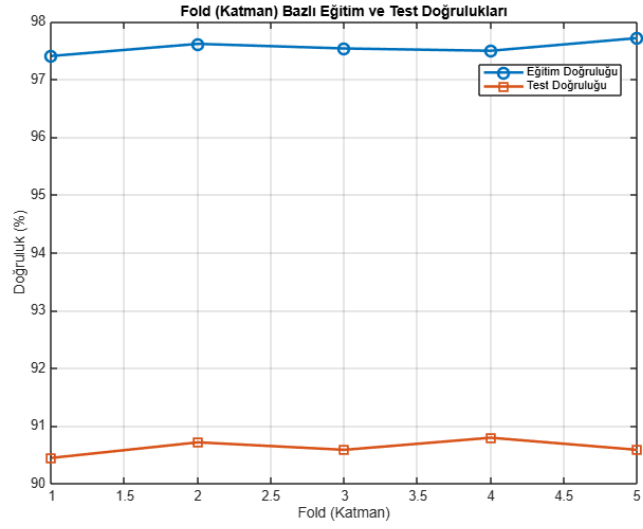
	Accuracy	Precision	Recall	F-1 Score
Defect	89,50	91,33	89,50	90,40
Longberry	96,25	95,30	96,25	95,77
Peaberry	91,25	92,41	91,25	91,82
Premium	88,00	86,06	88,00	87,02

Tablo 4.144 DenseNet201+HoG+SVM Modelinin Ortalama Performans Metrikleri

	Accuracy	Precision	Recall	F-1 Score
SVM	90,62	90,65	90,62	90,62

SVM (Destek Vektör Makineleri) algoritması veri setine uygulandığında elde edilen performans metrikleri doğruluk, hassasiyet, duyarlılık ve F1 skoru olarak belirlenmiştir. Bu modele ait performans metrikleri, KNN, Naive Bayes ve karar ağacı algoritmalarına kıyasla daha yüksektir. Modelin doğruluk oranı %90,62 olarak hesaplanmıştır. Hassasiyet değeri ise %90,65 olarak tespit edilmiştir. Modelin duyarlılık değeri %90,62 olarak belirlenmiştir. F1 skoru ise %90,62 olarak bulunmuştur.

Alt sınıfların performans metriklerine bakıldığında doğruluk oranlarının %88'inn üstünde olduğu gözlemlenmektedir. En yüksek doğruluk oranı %96,25 ile longberry sınıfındayken, en düşük doğruluk oranı %88,00 ile premium sınıfına aittir.



Şekil 4.48 DenseNet201+HoG+SVM Modelinin Katman Bazlı Eğitim ve Test Doğruluk Grafiği

5. TARTIŞMA VE SONUÇ

Bu çalışmada yeşil kahve çekirdeklerinin dört farklı türüne (defect, longberry, peaberry ve premium) göre sınıflandırılması hedeflenmiştir. Veri setinde ilk olarak VGG16, DenseNet201, MobileNetV2 ve ResNet18 modelleri özellik çıkarımı için kullanılarak KNN, Naive Bayes, karar ağacı ve SVM algoritmaları uygulanmışlardır. Daha sonra VGG16, DenseNet201, MobileNetV2 ve ResNet18 özellik çıkarımlarına ek olarak sırasıyla PCA ve HoG eklenerek kullanılarak özellik çıkarımı yapılarak algoritmalar test edilmiştir. Veri setine uygulanan farklı yöntemlere ait doğruluk oranları aşağıdaki tablolarda verilmiştir.

Tablo 5.1 VGG16 ile Uygulanan Algoritmalara Ait Doğruluk Oranları

VGG16		Öznitelik Çıkarma		
		VGG16	VGG16+PCA	VGG16+HoG
Sınıflandırma Yöntemleri	KNN	74,60 (k=7)	75,02 (k=7)	74,76 (k=7)
	Naive Bayes	68,01	67,56	64,47
	Decision Tree	60,56	56,66	58,44
	SVM	85,03	81,12	87,44

Tablo 5.2 Densenet201 ile Uygulanan Algoritmalara Ait Doğruluk Oranları

DenseNet201		Öznitelik Çıkarma		
		DenseNet201	DenseNet201+PCA	DenseNet201+HoG
Sınıflandırma Yöntemleri	KNN	78,80 (k=7)	83,60 (k=5)	71,01 (k=5)
	Naive Bayes	76,40	70,41	59,26
	Decision Tree	62,50	60,02	59,20
	SVM	88,95	91,99	90,62

Tablo 5.3 MobileNetV2 ile Uygulanan Algoritmalara Ait Doğruluk Oranları

MobileNetV2		Öznitelik Çıkarma		
		MobileNetV2	MobileNetV2+PCA	MobileNetV2+HoG
Sınıflandırma Yöntemleri	KNN	74,29 (k=7)	71,86 (k=7)	71,84 (k=7)
	Naive Bayes	30,90	32,07	40,86
	Decision Tree	56,96	56,29	53,90
	SVM	82,95	82,71	85,86

Tablo 5. 4 ResNet18 ile Uygulanan Algoritmalara Ait Doğruluk Oranları

ResNet18		Öznitelik Çıkarma		
		ResNet18	ResNet18+PCA	ResNet18+HoG
	KNN	75,62 (k=7)	75,03 (k=7)	72,20 (k=7)
Sınıflandırma	Naive Bayes	49,11	69,05	52,45
Yöntemleri	Decision Tree	59,12	56,60	54,43
	SVM	82,89	83,07	86,24

VGG16 modeline ait sonuçlar incelendiği zaman, en yüksek doğruluk oranının %87,44 ile HoG özellik çıkarımı ile kullanılan SVM algoritmasına ait olduğu görülmektedir. Normal verilere uygulanan SVM algoritmasında sonuç %85,03'ken, PCA ile sonuç %81,12'ye düşmüştür. Bu durum HoG ile çıkarılan özelliklerin SVM algoritmasının sınıflandırma performansını pozitif yönde etkilediğini göstermektedir. KNN algoritması uygulanan özellik çıkarım yöntemi farketmeksizin $k=7$ 'yken daha yüksek sınıflandırma performans göstermiştir. Naive Bayes algoritması genel olarak diğer algoritmalara kıyasla daha düşük sonuç vermiştir.

Karar ağacı algoritması tüm uygulamalarda yüksek eğitim doğruluk değerine sahipken, test doğruluk değerleri eğitim doğruluğunun çok altında bir değerde kalmıştır. Bunun sebebi modelin eğitim verisi üzerinde iyi çalışırken, test verisinde genelleme yapma yeteneğinin düşük olmasından kaynaklanmaktadır. Karar ağacı algoritması ne kadar çok derinleşirse, eğitim verisinin özelliklerine çok fazla uyum sağlayan bir yapıdadır. Eğer belirli sınıflarda çok fazla dal oluşumu olurken, belirli sınıflarda daha az dal oluşursa modelin genelleme özelliği negatif yönde etkilenir. Kullanılan veri seti ve kullanılan CNN metodları göz önünde bulundurularak hiperparametrelerin optimize edilmesi, aşırı öğrenmeden kaçınmak ve daha yüksek test doğruluk oranlarına ulaşabilmek için gerekmektedir.

DenseNet201 modeline ait sonuçlar incelendiğinde, en yüksek doğruluk oranının %91,99 ile PCA özellik çıkarımı ile kullanılan SVM algoritmasına ait olduğu görülmektedir. Bu durum PCA ile çıkarılan özelliklerin SVM algoritmasının sınıflandırma performansını pozitif yönde etkilediğini göstermektedir. Normal veri setine ve HoG ile çıkarılan özelliklere uygulanan SVM algoritmasında sonuçların biraz daha düşük olduğu görülmektedir. Karar ağacı algoritmasına ait doğruluk oranları diğer algoritmalara kıyasla daha düşüktür.

MobileNetV2 modeline ait sonuçları bakıldığında, en yüksek doğruluk oranının %85,86 ile HoG özellik çıkarımı ile kullanılan SVM algoritmasına ait olduğu görülmüştür. ResNet18 modeline ait sonuçları bakıldığında ise en yüksek doğruluk oranının %86,24 ile HoG özellik çıkarımı ile kullanılan SVM algoritmasına ait olduğu görülmüştür. Bu iki CNN algoritmasına ait uygulamalar incelendiğinde sonuçların genel olarak VGG16 ve DenseNet201 uygulamalarına göre daha düşük olduğu görülmüştür.

Tüm modeller incelendiği zaman tüm modellerde yüksek doğruluk oranlarına SVM algoritması ile ulaşıldığı görülmüştür. En yüksek doğruluk oranının %91,99 ile DenseNet201 ve PCA ile çıkarılan özelliklere uygulanan SVM algoritmasından olduğu gözükmemektedir. Tablolar incelendiğinde DenseNet201 uygulanan algoritmaların, genel olarak diğer modellere kıyasla daha yüksek sonuçlar verdiği görülmüştür. Naive Bayes ve karar ağacı modelleri dört konvolüsyonel sinir ağı modelinde de en düşük sonuçları vermiştir.

USK-Coffee veri setini oluşturan araştırmacılar ResNet18 ile %81,13 doğruluk oranına ulaşırken, MobileNetV2 ile %81,31 doğruluk oranına ulaşmışlardır. Tablo 5.3 ve Tablo 5.4 incelendiğinde SVM sınıflandırma yönteminin uygulandığı 3 uygulamada da veri setinin oluşturan araştırmacılardan daha yüksek sonuçlara ulaşıldığı görülmektedir. Bu çalışma, USK-Coffee veri seti üzerinde derin öğrenme algoritmalarının direkt olarak veri setine uygulanmasındansa, öznitelik çıkarma amacıyla kullanılıp ardından SVM algoritması ile sınıflandırma yapılmasının daha yüksek doğruluk sağladığı göstermektedir.

KAYNAKLAR

- [1] M. S. Butt and M. T. Sultan, "Coffee and its Consumption: Benefits and Risks," *Critical Reviews In Food Science and Nutrition*, vol. 51 no. 4, pp. 363-373, Mar 2011, doi: 10.1080/10408390903586412
- [2] M. F. Ramos, D. E. Ribeiro, M. Â. Cirillo and F. Meira, "Discrimination of the sensory quality of the *Coffea arabica* L. (cv. Yellow Bourbon) produced in different altitudes using decision trees obtained by the CHAID method," *Journal of the Science of Food and Agriculture*, vol 96, no. 10, pp. 3543-3551, Nov 2015, doi: 10.1002/jsfa.7539
- [3] F. Wei and M. Tanokura, "Chemical Changes in the Components of Coffee Beans during Roasting," *Coffee in Health and Disease Prevention*, pp. 83-91, 2015, doi: 10.1016/B978-0-12-409517-5.00010-3
- [4] J. N. C. Sarino, M. M. Bayas, E. R. Arboleda, E. C. Guevarra and R. M. Dellosa, "Classification Of Coffee Bean Degree Of Roast Using Image Processing And Neural Network," *International Journal of Scientific & Technology Research*, vol 8, no. 10, pp. 3231-3233, Oct 2019.
- [5] J. R. Santos, O. Viegas, R. N. Páscoa, I. M. Ferreira, A. O. Rangel and J. A. Lopes, "In-line monitoring of the coffee roasting process with near infrared spectroscopy: Measurement of sucrose and colour," *Food Chemistry*, vol 208, pp. 103-110, Oct 2016, doi: 10.1016/j.foodchem.2016.03.114
- [6] B. H. Belay and T. Tegegne, "Ethiopian Roasted Coffee Classification Using Imaging Techniques," *International Conference on the Advancement of Science and Technology*, May 2015
- [7] E. R. Arboleda, A. C. Fajardo and R. P. Medina, "An image processing technique for coffee black beans identification," 2018 IEEE International Conference on Innovative Research and Development (ICIRD), pp. 1-5, May 2018 doi: 10.1109/ICIRD.2018.8376325.
- [8] E. R. Arboleda, A. C. Fajardo and R. P. Medina, "Classification of coffee bean species using image processing, artificial neural network and K nearest neighbors,"

- 2018 IEEE International Conference on Innovative Research and Development (ICIRD), pp. 1-5, May 2018, doi: 10.1109/ICIRD.2018.8376326.
- [9] T. H. Nasution and U. Andayani, "Recognition of Roasted Coffee Bean Levels using Image Processing and Neural Network," IOP Conference Series: Materials Science and Engineering, vol. 180, no. 1, 2017, doi: 10.1088/1757-899X/180/1/012059
- [10] E. M. de Oliveira, D. S. Leme, B. H. G. Barbosa, M. P. Rodarte and R. G. F. A. Pereira, "A computer vision system for coffee beans classification based on computational intelligence techniques," Journal of Food Engineering, vol 171, pp. 22-27, Feb 2016, doi: 10.1016/j.foodeng.2015.10.009
- [11] Y. Unal, Y. S. Taspinar, I. Cinar, R. Kursun and M. Koklu, "Application of Pre-Trained Deep Convolutional Neural Networks for Coffee Beans Species Detection," Food Analytical Methods, vol 15, pp. 3232-3243, Aug. 2022.
- [12] A. Febriana, K. Muchtar, R. Dawood and C. -Y. Lin, "USK-COFFEE Dataset: A Multi-Class Green Arabica Coffee Bean Dataset for Deep Learning," 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), pp. 469-473, Jun 2022, doi: 10.1109/CyberneticsCom55287.2022.9865489
- [13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.
- [14] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks" Proceedings of the IEEE conference on computer vision and pattern recognition, Jan 2018, pp. 4510,4520, doi: 10.48550/arXiv.1801.04381
- [15] A. Moldagulova and R. B. Sulaiman, "Using KNN algorithm for classification of textual documents," 2017 8th International Conference on Information Technology (ICIT), May 2017, pp. 665-671, doi: 10.1109/ICITECH.2017.8079924.
- [16] A. Tharwat, H. Mahdi, M. Elhoseny and A. E. Hassanien, "Recognizing human activity in mobile crowdsensing environment using optimized k-NN algorithm,"

- Expert Systems with Applications, vol 107, pp. 32-44, Oct 2018, doi: 10.1016/j.eswa.2018.04.017
- [17] F.-J. Yang, "An Implementation of Naive Bayes Classifier," 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Dec 2018, pp. 301-306, doi: 10.1109/CSCI46756.2018.00065.
- [18] M. M. Saritas and A. Yasar, «Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification,» International Journal of Intelligent Systems and Applications in Engineering, vol 7, no. 2, pp. 88-91, Jun 2019, doi: 10.18201/ijisae.2019252786
- [19] B. De Ville, "Decision trees," WIREs Computational Statistics, vol 5, no. 6, pp. 448-455, Oct 2013, doi: 10.1002/wics.1278
- [20] H. Patel and P. Prajapati, "Study and Analysis of Decision Tree Based Classification Algorithms," International Journal of Computer Sciences and Engineering, vol 6, no. 10, pp. 74-78, Oct 2018.
- [21] J. Cervantes, F. G. Lamont, L. R. Mazahua and A. L. Chau, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," Neurocomputing, vol 408, pp. 189-215, Sep 2020, doi: 10.1016/j.neucom.2019.10.118
- [22] H. Abdi and L. J. Williams, "Principal Component Analysis," Wiley Interdisciplinary Reviews: Computational Statistics, vol 2, no. 4, pp. 433-459, Jul 2010, doi: 10.1002/wics.101
- [23] M. Greenacre, P. J. Groenen, T. Hastie, A. I. D'Enza, A. Markos and E. Tuzhilina, "Principal component analysis," Nature Reviews Methods Primers, vol 2, no. 1, Dec 2022, doi: 10.1038/s43586-022-00184-w
- [24] P. Carcagnì, M. D. Coco, M. Leo and C. Distanto, "Facial expression recognition and histograms of oriented gradients: a comprehensive study," vol. 4, no. 1, Jan 2015
- [25] N. Seliya, T. M. Khoshgoftaar and J. Van Hulse, "A Study on the Relationships of Classifier Performance Metrics," 2009 21st IEEE International Conference on Tools with Artificial Intelligence, Nov 2009, pp. 59-66, doi: 10.1109/ICTAI.2009.25.

- [26] C. Ferri, J. H. Orallo ve E. R. Modroiou, "An experimental comparison of performance measures for classification," *Pattern Recongition Letters*, vol. 30, no. 1, pp. 27-38, Jan. 2009, 10.1016/j.patrec.2008.08.010
- [27] J. Davis ve M. Goadrich, "The relationship between Precision-Recall and ROC curves," *Proceedings of the 23rd international conference on Machine learning*, pp. 233-240, Jun 2006, doi: 10.1145/1143844.1143874
- [28] M. Vakili, M. Ghamsari ve M. Rezaei, "Performance Analysis and Comparison of Machine and Deep Learning Algorithms for IoT Data Classification," 2020, doi: 10.48550/arXiv.2001.09636
- [29] C. Pinto, J. Furukawa, H. Fukai and S. Tamura, "Classification of Green coffee bean images basec on defect types using convolutional neural network (CNN)," 2017 International Conference on Advanced Informatics, Concepts, Theory, and Applications (ICAICTA), Aug. 2017, pp. 1-5, doi: 10.1109/ICAICTA.2017.8090980
- [30] B.T. Tolawak, G. Abebe and G. Goro, "Classification of Ethiopian Coffee Beans Using Imaging Techniques," *East African Jounrla of Sciences*, vol. 7, no. 1 pp.1-10, Jan 2013
- [31] N. F. Huang, D.L Chou, C. A. Lee, F.P. Wu, A. C. Chuang, Y. H. Chen and Y. C. Tsai, "Smart agriculture: real-time classification of green coffee beans by using a convolutional neural network" *IET Smart Cities*, vol. 2, no. 4, pp. 167-172, Oct 2020, doi: 10.1049/iet-smc.2020.0068
- [32] H. L. Gope and H.Fukai, "Normal and peaberry coffee beans classification from green coffee bean images using convolutional neural networks and support vector machine," *International Journal of Computer and Information Engineering*, vol. 14, no. 6, pp. 189-196, 2020
- [33] J.P.L. Pizzaia, I.R. Salcides, G. Almeida, R. Contarato and R. de Almeida, "Arabica coffee samples classification using a Multilayer Perceptron neural network," 13th IEEE International Conference on Industry Applications (INDUSCON) pp. 80-84, Nov 2018 doi: 10.1109/INDUSCON.2018.8627271.

- [34]R. Janandi and T. W. Cenggoro “An Implementation of Convolutional Neural Network for Coffee Beans Quality Classification in a Mobile Information System” Aug. 2020, doi: 10.13140/RG.2.2.23163.54561
- [35]N. -F. Huang, D. -L. Chou and C. -A. Lee, "Real-Time Classification of Green Coffee Beans by Using a Convolutional Neural Network," 2019 3rd International Conference on Imaging, Signal Processing and Communication (ICISPC), pp. 107-111, Jul. 2019, doi: 10.1109/ICISPC.2019.8935644.
- [36]X. Ou, P. Yan, Y. Zhang, B. Tu, G. Zhang, J. Wu and W. Li, “Moving Object Detection Method via ResNet-18 With Encoder–Decoder Structure in Complex Scenes” IEEE Access vol.7, Aug 2019
- [37]M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, M. Hasan, B. C. V. Esesn, A. A. S. Aswal and V.K. Asari, “The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches,” Mar. 2018
- [38]M. W. Browne, “Cross-Validation Methods,” Journal of Mathematical Psychology, vol. 44, no. 1, pp. 108-132, Mar 2000, doi: 10.1006/jmps.1999.1279
- [39]J. Schmidhuber, “Annotated History of Modern AI and Deep Learning,” Dec. 2018
- [40]T. Williams and R. Li, “An Ensemble of Convolutional Neural Networks Using Wavelets for Image Classification,” Journal of Software Engineering and Applications, vol:11, no: 2 pp.69-88, Jan 2018, doi: 10.4236/jsea.2018.112004
- [41]L. Zhao and Z. Zhang, “A improved pooling method for convolutional neural networks,” Scientific Reports, vol. 14, no. 1, Jan 2024, doi: 10.1038/s41598-024-51258-6
- [42]K. Simonyan and A. Zisserman “Very Deep Convolutional Networks for Large-Scale Image Recognition,” 2014
- [43]H. Yang, J. Ni, J. Gao and T. Luan, “A novel method for peanut variety identification and classification by Improved VGG16,” vol. 11, no. 1, Aug 2021, doi: 10.1038/s41598-021-95240-y

- [44] G. Huang, Z. Liu, L. van der Maaten and K.Q. Weinberger, "Densely connected convolutional networks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4700-4708, 2017
- [45] S. P. Godlin Jasil and V. Ulagamuthalvi, "Skin Lesion Classification Using Pre-Trained DenseNet201 Deep Neural Network," 2021 3rd International Conference on Signal Processing and Communication (ICPSC), May 2021, pp. 393-396, doi: 10.1109/ICSPC51351.2021.9451818.
- [46] T. Y. Hsiao, Y. C. Chang, H. H. Chou and C. T. Chiu, "Filter-based deep-compression with global average pooling for convolutional networks," Journal of Systems Architecture, vol. 95, pp. 9-18, May 2019, doi: 10.1016/j.sysarc.2019.02.008