

T.C.
BANDIRMA ONYEDİ EYLÜL ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ZEYTİN MEYVESİ HASTALIKLARININ
DERİN ÖĞRENME İLE TESPİTİ

Çağla TOPRAK ERDURAK

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

Haziran 2025

T.C.
BANDIRMA ONYEDİ EYLÜL ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ZEYTİN MEYVESİ HASTALIKLARININ
DERİN ÖĞRENME İLE TESPİTİ

Çağla TOPRAK ERDURAK

DANIŞMAN

Doç. Dr. Serhat KILIÇARSLAN

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

Haziran 2025

ONAY

Çağla TOPRAK ERDURAK tarafından hazırlanan “Zeytin Meyvesi Hastalıklarının Derin Öğrenme ile Tespiti” adlı tez çalışması 24/06/2025 tarihinde yapılan sınavla aşağıdaki jüri tarafından Bandırma Onyedİ Eylül Üniversitesi Lisansüstü Eğitim Enstitüsü Mekatronik Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Doç. Dr. Serhat
KILIÇARSLAN

(Danışman)

Doç. Dr. Emrah DÖNMEZ

(Üye)

Doç. Dr. Kemal ADEM

(Üye)

Bu tezin kabulü, Lisansüstü Eğitim Enstitüsü Yönetim Kurulu’nun / / gün ve / sayılı kararı ile onaylanmıştır.

Prof. Dr. Serhat DUMAN

Enstitü Müdürü

BANDIRMA ONYEDİ EYLÜL ÜNİVERSİTESİ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
YÜKSEK LİSANS TEZİ
ETİK BEYANI

Bandırma Onyedi Eylül Üniversitesi Lisansüstü Eğitim Enstitüsü tez yazım kılavuzuna göre hazırlamış olduğum “**Zeytin Meyvesi Hastalıklarının Derin Öğrenme ile Tespiti**” adlı tezimin özgün bir çalışma olduğunu, tez hazırlanırken tüm aşamalarda bilimsel etik ilkelerine uygun davrandığımı, tez kapsamında sunulan tüm verileri bilimsel etik ilkelerine uygun elde ettiğimi, tezde faydalandığım tüm eserlere atıf yaptığımı ve kaynaklar kısmında bu eserleri gösterdiğimi beyan ederim. 24/06/2025

Çağla TOPRAK ERDURAK
İmza

ÖNSÖZ

Tarım sektöründe sürdürülebilirlik, verimlilik ve kalite odaklı yaklaşımlar, son yıllarda teknolojik gelişmelerin ışığında hızla önem kazanmıştır. Özellikle zeytin tarımı hem ülkemizin hem de Akdeniz Havzası'nın ekonomik ve kültürel açıdan vazgeçilmez bir unsuru olarak öne çıkmaktadır. Zeytin tarımında hastalıkların erken teşhisi, ürün kalitesi ve verimliliğin korunması ise kritik bir öneme sahiptir.

Bu tez çalışmasında zeytin meyvesi görüntülerinden hastalık tespiti amacıyla transfer öğrenme ve Transformer tabanlı güncel derin öğrenme modellerinin sınıflandırma performansları karşılaştırmalı olarak analiz edilmiştir. Bunun yanı sıra, özgün MetaFormer tabanlı mimari kombinasyonları ve yeni token mixer yaklaşımları geliştirilerek literatüre katkı sağlanması hedeflenmiştir.

Bu tezin hazırlanması sürecinde, bilgi ve tecrübeleriyle her zaman yol gösteren, destek ve ilgisini hiçbir zaman esirgemeyen değerli danışmanım Doç. Dr. Serhat KILIÇARSLAN'a en içten teşekkürlerimi sunarım. Kendisinin akademik rehberliği ve yönlendirmeleri, çalışmanın her aşamasında yolumu aydınlatmıştır.

Zeytin verilerinin toplanmasında sağladığı katkılar için Marmarabirlik Yönetim Kurulu Başkanı Sayın Mehmet ÖZÇAKIR'a ve zeytin hastalıkları konusunda bilgi ve desteğini esirgemeyen Ziraat Mühendisi Umut ŞİNO'ya teşekkür ederim. Bu katkılar, çalışmanın sahaya dayalı verilerle daha sağlıklı biçimde yürütülmesini mümkün kılmıştır.

Hayatım boyunca yanımda olan, sevgileri ve destekleriyle bana güç veren canım annem ve babam Mehtap ve Necati TOPRAK'a; sevgili kardeşlerime ve her zaman desteğini hissettiren eşimin kıymetli ailesi Nihal ve Bayram ERDURAK'a en derin şükranlarımı sunuyorum.

Son olarak sevgisi ve sonsuz desteğiyle her daim yanımda olan sevgili eşim Yasin ERDURAK'a en içten teşekkürlerimi sunarım. Bu süreci onun inancı ve desteği olmadan tamamlamam mümkün olmazdı.

ÖZET

YÜKSEK LİSANS TEZİ

ZEYTİN MEYVESİ HASTALIKLARININ DERİN ÖĞRENME İLE TESPİTİ

Çağla TOPRAK ERDURAK

Bandırma Onyedİ Eylöl Üniversitesi
Lisansüstü Eğitim Enstitüsü
Mekatronik Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Serhat KILIÇARSLAN

Haziran 2025, 103 sayfa

Bu tez çalışmasında, zeytin meyvesinde yaygın olarak görülen iki önemli hastalığın zararlılara bağı fiziksel bozulmalar ve fungal enfeksiyonlara dayalı deformasyonların görsel veriler üzerinden derin öğrenme yöntemleriyle sınıflandırılmasına yönelik kapsamlı bir analiz gerçekleştirilmiştir. Balıkesir ili Bandırma ilçesindeki zeytinliklerden toplanan 1644 meyve örneğı “sağlıklı”, “zeytin sineğı ısııklı” ve “mantar hastalıklı” olmak üzere üç sınıfa ayrılmıştır. Tüm meyveler, sabit ışık koşullarında özel olarak tasarlanmış bir görüntüleme kutusunda fotoğraflanmış; ardından segmentasyon, eşikleme, morfolojik işlemler ve nesne yerelleştirme gibi görüntü işleme adımlarıyla sınıflandırmaya hazır hâle getirilmiştir.

Modelleme sürecinde, dokuz farklı CNN tabanlı önceden eğitilmiş mimari (VGG16, ResNet50, InceptionV3, DenseNet121, vb.) ile birlikte altı adet dikkat tabanlı Transformer modeli (ViT, BEiT, DeiT III, MobileViT, LeViT, MaxViT) kullanılmış; her biri on farklı optimizasyon algoritması ile ayrı ayrı eğitilerek sınıflandırma performansları karşılaştırmalı olarak analiz edilmiştir. Bunun yanı sıra, MetaFormer iskeletine dayalı CaFormer, ConvFormer ve

IdentityFormer gibi mevcut modeller test edilmiş ve bu yapılarla aynı veri seti üzerinde sistematik karşılaştırmalar gerçekleştirilmiştir.

Çalışmanın özgün katkılarından biri olarak, MetaFormer mimarisinden esinlenilerek altı yeni hibrit yapı önerilmiştir. Bu yapılar, konvolüsyonel, rastgele, kimliksel ve Transformer bloklarının farklı kombinasyonlarıyla oluşturulmuş ve sınıflandırma doğruluğu, model karmaşıklığı ve eğitim süresi gibi çok boyutlu ölçütlerle değerlendirilmiştir. Ayrıca, DeformationScore adını taşıyan ve lokal doku yoğunluğu ile kenar duyarlılığına dayalı skor üretimi gerçekleştiren yeni bir token mixer modülü tasarlanmıştır; MetaFormer çerçevesine entegre edilerek DeformationScoreFormer adlı tamamen yeni ve özgün bir model ortaya konmuştur.

Model performansları; doğruluk, F1 skoru, karışıklık matrisi ile eğitim ve kayıp eğrileri gibi metrikler üzerinden ayrıntılı olarak değerlendirilmiştir. Elde edilen sonuçlar hem klasik CNN tabanlı modellerin hem de Transformer ve MetaFormer mimarilerinin sınırlı veri koşullarında dahi zeytin meyvesi hastalıklarını yüksek doğrulukla sınıflandırabildiğini ortaya koymuştur. Özellikle önerilen hibrit MetaFormer yapılarının, bazı durumlarda daha düşük hesaplama maliyetiyle daha yüksek sınıflandırma başarımı sunduğu gözlemlenmiştir. Bu çalışma, zeytin üretiminde hastalık tespiti ve kalite kontrol süreçlerine yönelik derin öğrenme tabanlı yaklaşımların etkinliğini göstermekte; aynı zamanda tarımsal görüntü analizinde model seçimi ve mimari tasarım kararlarına rehberlik eden bir kaynak sunmaktadır.

Anahtar Kelimeler: Derin Öğrenme, Zeytin Meyvesi Hastalıkları, Transformer, Metaformer

ABSTRACT

M.Sc. THESIS

DETECTION OF OLIVE FRUIT DISEASES WITH DEEP LEARNING

Çağla TOPRAK ERDURAK

**Bandırma Onyedi Eylül University
Graduate Education Institute
Department of Mechatronics Engineering**

Advisor: Assoc. Prof. Dr. Serhat KILIÇARSLAN

June 2025, 103 pages

This thesis presents a comprehensive analysis aimed at classifying two major diseases commonly observed in olive fruits physical damage caused by pests and deformations due to fungal infections through deep learning techniques applied to visual data. A total of 1,644 olive fruit samples were collected from orchards in the Bandırma district of Balıkesir province and categorized into three classes: “healthy,” “pest-damaged,” and “fungal-infected,” based on expert evaluation. All samples were photographed under controlled lighting conditions using a specially designed imaging box. The resulting images were then processed using segmentation, thresholding, morphological operations, and object localization steps to prepare them for classification.

In the modeling phase, nine different pre-trained CNN-based architectures (e.g., VGG16, ResNet50, InceptionV3, DenseNet121) and six attention-based Transformer models (ViT, BEiT, DeiT III, MobileViT, LeViT, MaxViT) were employed. Each model was trained with ten different optimization algorithms and comparatively evaluated in terms of classification performance.

Additionally, existing MetaFormer-based models such as CaFormer, ConvFormer, and IdentityFormer were tested on the same dataset for systematic comparison.

One of the key contributions of this study is the proposal of six novel hybrid architectures inspired by the MetaFormer framework. These architectures combine convolutional, random, identity, and Transformer blocks in various configurations, and were assessed using multi-dimensional criteria including classification accuracy, architectural complexity, and training time. Furthermore, a novel token mixer module named DeformationScore—which generates scores based on local texture density and edge sensitivity—was designed and integrated into the MetaFormer structure to form an entirely new model called DeformationScoreFormer.

Model performances were thoroughly evaluated using metrics such as accuracy, F1-score, confusion matrix, and training-validation loss curves. The results demonstrated that both conventional CNN-based models and Transformer/MetaFormer-based architectures can accurately classify olive fruit diseases even under limited data conditions. Notably, the proposed hybrid MetaFormer models outperformed some existing models by achieving higher accuracy with lower computational cost in certain scenarios. This study highlights the effectiveness of deep learning approaches for disease detection and quality control in olive production and serves as a valuable resource for model selection and architectural design in agricultural image analysis.

Keywords: Deep Learning, Olive Fruit Diseases, Transformer, MetaFormer

İÇİNDEKİLER

	Sayfa
ÖZET	I
ABSTRACT	III
İÇİNDEKİLER.....	V
ŞEKİL LİSTESİ	VIII
TABLO LİSTESİ.....	XI
SİMGE VE KISALTMA LİSTESİ.....	XII
1. BÖLÜM: GİRİŞ VE LİTERATÜR İNCELEMESİ.....	1
1.1 Giriş	1
1.2 Tez İçeriği.....	4
1.3 Tezin Amacı ve Katkısı	5
1.4 Literatür Taraması.....	5
2. BÖLÜM: MATERYAL VE YÖNTEM	9
2.1 Veri Seti Tanımı ve Görüntü Toplama Süreci.....	9
2.2 Görüntü İşleme ve Ön İşleme.....	10
2.2.1 Evrişimsel Sinir Ağları.....	11
2.3 CNN Tabanlı Transfer Öğrenme Modelleri	12
2.3.1 VGG16.....	14
2.3.2 ResNet50.....	15
2.3.3 InceptionV3	16
2.3.4 DenseNet121	17
2.3.5 NASNet Mobile	18
2.3.6 MobileNet	19
2.3.7 Xception Mimarisi	20
2.3.8 InceptionResNetV2.....	22

2.3.9	EfficientNetB0	23
2.4	Transformer Tabanlı Modeller	24
2.4.1	Vision Transformer (ViT).....	26
2.4.2	BERT Pre-Training of Image Transformers (BEiT)	27
2.4.3	Data-efficient Image Transformer (DeiT III)	28
2.4.4	Mobile Vision Transformer (MobileViT v2).....	29
2.4.5	Multi-Axis Vision Transformer (MaxViT).....	30
2.4.6	Lightweight & Efficient Vision Transformer (LeViT).....	32
2.4.7	Transformer Tabanlı Modeller İçin Deneysel Ayarlar/Optimizasyon Parametreleri	33
2.5	MetaFormer Tabanlı Yapılar.....	35
2.5.1	CAFormer	37
2.5.2	ConvFormer	39
2.5.3	PoolFormer.....	40
2.5.4	IdentityFormer	41
2.5.5	RandFormer.....	42
2.6	MetaFormer Tabanlı Özgün Blok Kombinasyonları.....	43
2.6.1	Hiperparametre Ayarları.....	48
2.7	Önerilen Model Mimarisi: DeformationScoreFormer.....	49
2.7.1	Mimari Genel Bakış	50
2.7.2	DeformationFormer Bloğu	51
2.7.3	Aşama Tabanlı Mimari ve Çıkış Boyutları.....	53
2.7.4	Eğitim Yapılandırmaları ve Hiperparametreler	53
2.7.5	Modelin Matematiksel Açıklaması	54
2.8	Değerlendirme Metrikleri.....	56
3.	BULGULAR.....	57
3.1	Transfer Öğrenme Tabanlı Mimarilerinin Karşılaştırmalı Analizi	57

3.1.1	Transfer Öğrenme Modellerinin Performans Karşılaştırması	57
3.1.2	Modellerin Karşılaştırmalı Değerlendirilmesi.....	60
3.2	Transformer Tabanlı Görüntü Sınıflandırma Modelleri ve Optimizasyon Yöntemlerinin Karşılaştırmalı Analizi	62
3.2.1	BEiT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi	62
3.2.2	DeiT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi.....	65
3.2.3	LeViT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi	67
3.2.4	MaxViT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi.....	70
3.2.5	MobileViT: Optimizasyon Algoritmalarına Göre Performans Analizi.....	73
3.2.6	ViT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi	75
3.2.7	Genel Değerlendirme ve Sonuç	78
3.3	MetaFormer Tabanlı Modellerin Karşılaştırmalı Performans Analizi.....	79
3.4	MetaFormer Tabanlı Özgün Hibrit Blok Kombinasyonlarının Sınıflandırma Performansları	82
3.5	Önerilen Özgün DeformationScoreFormer Mimarisi ile Zeytin Görüntülerinin Sınıflandırılması: Performans Analizi ve Değerlendirme.....	84
3.5.1	Sınıflandırma Performansı: Doğruluk, Kayıp ve F1-Skoru	84
3.5.2	DeformationScore Haritaları ile Dikkat Bölgelerinin Yorumlanması	85
3.5.3	Grad-CAM Tabanlı Görsel Açıklanabilirlik	87
3.6	Tartışma ve Sonuç	88
3.7	Öneriler	90
4.	KAYNAKLAR	93
	ÖZGEÇMİŞ	104

ŞEKİL LİSTESİ

<u>Sekil</u>	<u>Sayfa</u>
Şekil 2- 1: Veri seti için zeytin meyvesi toplama süreci.....	9
Şekil 2- 2: Zeytin meyvelerinin sınıflandırılması amacıyla kullanılan kapalı görüntüleme kutusu	10
Şekil 2- 3: Segmentasyon ve ön işleme süreci sonrası elde edilen, sınıflandırma için kullanılmaya hazır örnek zeytin meyvesi görselleri (soldan sağa: sinek ısırıklı, mantar hastalıklı, sağlıklı).....	11
Şekil 2- 4: Transfer öğrenme mimarisinin genel yapısı	13
Şekil 2- 5: VGG16 modelinin genel mimari yapısı	14
Şekil 2- 6: ResNet50 mimarisinin blok temelli yapısı.....	15
Şekil 2- 7: InceptionV3 mimarisinin katman temelli genel yapısı.	17
Şekil 2- 8: DenseNet121 modelinin genel mimari akışı	18
Şekil 2- 9: NASNet-Mobile mimarisinin şematik akışı.	19
Şekil 2- 10: MobileNetV1 mimarisinin blok temelli yapısı	20
Şekil 2- 11: Xception mimarisinin blok diyagramı.....	21
Şekil 2- 12: InceptionResNetV2 modelinin mimari akış diyagramı	22
Şekil 2- 13: EfficientNetB0 mimarisinin blok temelli yapısı.....	23
Şekil 2- 14: Transformer mimarisi	25
Şekil 2- 15: Vision Transformer mimarisinin yapısal akışı.....	26
Şekil 2- 16: BEiT modelinin yapısal akışı.....	28
Şekil 2- 17: DeiT mimarisinin yapısal şeması	29
Şekil 2- 18: MobileViT v2 modelinin yapısal akışı	30
Şekil 2- 19: MaxViT mimarisinin şematik yapısı.	31
Şekil 2- 20: LeViT mimarisinin katman yapısını ve evreleme sistemini gösteren şematik yapı.....	32
Şekil 2- 21: MetaFormer yapısının genel mimari iskeleti.....	36
Şekil 2- 22: CFormer mimarisinin dört aşamalı genel şeması.....	38
Şekil 2- 23: ConvFormer mimarisinin dört aşamalı genel yapısı.....	40
Şekil 2- 24: PoolFormer mimarisinin dört aşamalı genel yapısı	41
Şekil 2- 25: IdentityFormer mimarisinin dört aşamalı genel yapısı	42
Şekil 2- 26: RandFormer mimarisinin dört aşamalı genel yapısı	43

Şekil 2- 27: Geliştirilen özgün hibrit modellerin aşamalı blok dizilimleri	47
Şekil 2- 28: DeformationScoreFormer mimarisinin aşama bazlı genel yapısı.....	51
Şekil 3- 1: Modellerin Hold-Out yöntemiyle elde edilen karmaşıklık matrisleri	58
Şekil 3- 2: Modellerin 10-Fold çapraz doğrulama elde edilen karmaşıklık matrisleri	60
Şekil 3- 3: BEiT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri	63
Şekil 3- 4: BEiT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	64
Şekil 3- 5: BEiT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	64
Şekil 3- 6: DEiT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri	66
Şekil 3- 7: DEiT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	66
Şekil 3- 8: DEiT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	67
Şekil 3- 9 : LeViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri	68
Şekil 3- 10: LeViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	69
Şekil 3- 11: LeViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	70
Şekil 3- 12: MaxViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri.....	71
Şekil 3- 13: MaxViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	72
Şekil 3- 14: MaxViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	72
Şekil 3- 15: MobileViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri.....	74
Şekil 3- 16: MobileViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	74

Şekil 3- 17: MobileViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	75
Şekil 3- 18: ViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri	76
Şekil 3- 19: ViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	77
Şekil 3- 20: ViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi	78
Şekil 3- 21: CAFormer-S18 Modeli Karmaşıklık Matrisi	80
Şekil 3- 22: ConvFormer-S18 Modeli Karmaşıklık Matrisi	81
Şekil 3- 23: PoolFormer-V2-S12 Modeli Karmaşıklık Matrisi	82
Şekil 3- 24: DeformationScoreFormer modeli için test veri kümesine ait karışıklık matrisi.	85
Şekil 3- 25: Sırasıyla sinek ısırığı, mantar enfeksiyonu ve sağlıklı örnekler için DeformationScore haritaları ve bindirmeli görseller	86
Şekil 3- 26: Sinek ısırıklı, sağlıklı ve mantar hastalıklı sınıflarına ait Grad-CAM çıktıları.	88

TABLO LİSTESİ

<u>Tablo</u>	<u>Sayfa</u>
Tablo 1: Zeytin Tarımında Derin Öğrenme ve Görüntü İşleme Yöntemlerine Dayalı Literatür Özeti	7
Tablo 2- 1: Blok Türlerinin Bilişsel Roller ve Literatür Dayanakları.....	44
Tablo 2- 2: Geliştirilen özgün hibrit modellerin katman temelli yapısal parametreleri.....	48
Tablo 2- 3: Geliştirilen özgün hibrit modellerin ortak hiperparametre ayarları.....	49
Tablo 2- 4: Geliştirilen özgün hibrit modellerin parametre ve FLOPs değerleri.....	49
Tablo 2- 5: DeformationScoreFormer mimarisinin katman düzeyinde yapısal özellikleri..	53
Tablo 2- 6: DeformationScoreFormer mimarisinin eğitim yapılandırmaları	54
Tablo 3- 1: Modellerin Hold-Out yöntemiyle elde edilen doğrulama kümesindeki genel sınıflandırma performansları	58
Tablo 3- 2: Modellerin 10-Fold çapraz doğrulama elde edilen genel sınıflandırma performansları	59
Tablo 3- 3: BEiT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları	62
Tablo 3- 4: DEiT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları	65
Tablo 3- 5: LeViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları	68
Tablo 3- 6: MaxViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları	70
Tablo 3- 7: MobileViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları	73
Tablo 3- 8: ViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları	76
Tablo 3- 9: MetaFormer Tabanlı Modellerin Performans Sonuçları.....	80
Tablo 3- 10: MetaFormer Tabanlı Özgün Hibrit Modellerin Performans Sonuçları.....	83
Tablo 3- 11: Önerilen metaformer modelinin performans sonuçları	84

SİMGE VE KISALTMA LİSTESİ

Simgeler

σ	: Standart sapma
μ	: Ortalama
ϵ	: Sayısal kararlılık
γ	: MLP genişletme oranı

Açıklama

Kisaltmalar

B	: Mini-batch boyutu
C	: Kanal sayısı
H, W	: Görüntünün yükseklik (Height) ve genişlik (Width) boyutları
x	: Girdi tensörü
x'	: Skorla modüle edilmiş çıkış tensörü (attention etkili çıkış)
x_gray	: RGB girdiden gri seviyeye dönüştürülmüş tensör
x_flat	: Yeniden şekillendirilmiş düzleştirilmiş tensör
x_mlp	: MLP çıktısı
x_out	: MLP sonrası BatchNorm ve residual ile birleştirilmiş çıkış
diff	: Pürüzlük fark tensörü (orijinal – ortalama havuzlu görüntü)
lap	: Kenar bilgisi (Laplace filtresi sonrası tensör)
score	: Normalleştirilmiş ve sigmoid aktivasyon uygulanmış skor tensörü
score_ref	: 1×1 konvolüsyonla rafine edilmiş skor tensörü
W_1, W_2	: İlk ve ikinci lineer katmana ait ağırlık matrisi
b_1, b_2	: Lineer katmanlara ait bias vektörleri
h, h'	: Aktivasyon sonrası geçici MLP çıktıları
BN	: Batch Normalization işlemi
Conv1x1	: 1×1 konvolüsyon katmanı
GELU	: Gaussian Error Linear Unit aktivasyon fonksiyonu
StarReLU	: Transformer mimarilerinde kullanılan optimize edilmiş aktivasyon
MLP	: Çok katmanlı algılayıcı (Multi-Layer Perceptron)
FLOPs	: Kayan noktalı aritmetik işlemlerin sayısı

MACs : Çarp-topla işlem sayısı
Patch Size : Görüntü yamasının boyutu
Stride : Konvolüsyon adımı



1. BÖLÜM: GİRİŞ VE LİTERATÜR İNCELEMESİ

1.1 Giriş

Zeytin ağacı (*Olea europaea* L.), sert çekirdekli meyveler grubuna dâhil, daimi yeşil yapraklı, uzun ömürlü ve derin kök yapısı sayesinde kurak iklim koşullarına yüksek adaptasyon kabiliyeti gösteren bir türdür [1]. Genetik olarak geniş varyasyon gösteren bu tür, farklı ekolojik kuşaklara yayılabilmesi ve iklimsel değişkenliğe karşı dayanıklılığı sayesinde Akdeniz tarımsal peyzajının omurgasını oluşturmaktadır [2] .

Dünya genelinde zeytin üretimi büyük oranda Akdeniz iklim kuşağına sahip ülkelerde yoğunlaşmakta bu bağlamda İspanya, İtalya, Yunanistan, Türkiye ve Fas en yüksek üretim hacmine sahip beş ülke arasında yer almaktadır [3]. 2023 verilerine göre dünya zeytin üretiminin yaklaşık %70'inden fazlası bu ülkeler tarafından gerçekleştirilmektedir. Türkiye, toplam üretimde dördüncü sırada yer almakta; özellikle Ege ve Marmara bölgelerinde yoğunlaşan üretim alanları ile stratejik bir konumda bulunmaktadır [4].

Zeytin üretimi, sofralık zeytin ve zeytinyağı olmak üzere iki temel işleme kategorisine ayrılmakta aynı zamanda yan ürünleriyle de yüksek katma değer potansiyeli sunmaktadır [5]. Zeytinyağı, içerdiği oleik asit, fenolik bileşikler ve E vitamini nedeniyle fonksiyonel gıda niteliği taşımakta özellikle kalp-damar sağlığı başta olmak üzere birçok kronik hastalığın önlenmesinde rol oynamaktadır [6]. Bunun yanı sıra zeytin yaprağı, çekirdeği ve karasuyu gibi işleme yan ürünleri; farmakolojik, antimikrobiyal, kozmetik ve hayvancılığa yönelik farklı sektörlerde kullanılmaktadır [4].

Zeytinin çok yönlü faydaları, onu yalnızca tarımsal bir üretim nesnesi değil aynı zamanda ekonomik büyüme, halk sağlığı, çevresel sürdürülebilirlik ve kültürel devamlılık açısından bütüncül değerlendirilmeyi gerektiren stratejik bir kaynak haline getirmiştir [7]. Bu nedenle zeytine dair yapılacak analizler, salt tarımsal verimlilik ölçütleriyle sınırlı kalmamalı ekolojik direnç, yerel bilgi sistemleri ve katma değerli işleme potansiyeli gibi alanları da kapsayacak şekilde çok boyutlu bir çerçevede ele alınmalıdır [3]. Ancak bu çok boyutlu yapının korunması ve sürdürülebilir kılınması, üretim sürecini etkileyen doğal ve biyotik risklerin doğru şekilde tanımlanmasını ve yönetilmesini zorunlu kılmaktadır. Özellikle iklim değişikliğinin etkisiyle birlikte Akdeniz kuşağında biyotik stres faktörlerinin yayılımı artmakta bu durum, zeytin üretimini daha kırılgan bir hâle getirmektedir [8]. Fungal, bakteriyel ve viral hastalıkların yanı sıra zararlılar da bu risklerin başında gelmekte olup, özellikle ürün kalitesi üzerinde kalıcı etkiler

bırakabilmektedir. Türkiye’de yapılan güncel saha çalışmalarında, sıcaklık ve nem gibi mikroklimatik değişkenlerin zararlı popülasyonlarını doğrudan etkilediği ve bazı bölgelerde risk düzeyini artırdığı raporlanmıştır [9].

Zeytin meyvesini doğrudan etkileyen en önemli zararlılardan biri olan zeytin sineği (*Bactrocera oleae*), yumurtlama döneminde meyveye verdiği zarar yoluyla hem doğrudan meyve dokusunu tahrip etmekte hem de ikincil mikrobiyal enfeksiyonlara zemin hazırlayarak ürünün genel kalitesini ve zeytinyağının kimyasal ile duyuşsal özelliklerini ciddi biçimde olumsuz etkilemektedir [10,11]. Popülasyon yoğunluğu; sıcaklık, nem ve meyve varlığı gibi çevresel faktörlere bağılı olarak değişkenlik göstermekte olup, yıllık iklim koşulları sinek aktivitesini doğrudan belirleyici bir rol oynamaktadır [12].

Larvalar, meyve içerisinde oluşturdıkları galeriler aracılığıyla hem doğrudan besin kaybına neden olmakta hem de ikincil enfeksiyonlara zemin hazırlayarak ürünün ticari değerini düşürmektedir. Bu zararlıya karşı yürütölen mücadele yöntemlerinde, özellikle organik üretim sistemlerinde biyoteknik yaklaşımlar –feromon tuzakları, cezbedici yem tuzakları gibi– öncelikli olarak tercih edilmekte bunun yanı sıra çevre dostu biyolojik ajanlar ve kitle imha yöntemleri de entegre mücadele programları kapsamında etkili biçimde kullanılmaktadır [13].

Zeytin sineği zararını takiben meyvede sıkça gözlenen fungal patojenlerden en belirgin olanı, antraknoz etmeni olan *Colletotrichum spp.*’dir. Bu patojen, meyvede çökük lekeler, renk değişimi ve çürümeye neden olarak özellikle yağlık çeşitlerde ciddi kalite kayıpları yaratmaktadır. Nemli ve sıcak koşullarda hızla yayılan bu hastalık, enfekte meyveler zamanında toplanmazsa salgına dönüşebilir [14]. Olgunluk döneminde oluşın fiziksel hasarlar ile birleşen yüzeysel mantar enfeksiyonları da meyve dokusunda ciddi bozulmalara yol açar. Bu durum hem sofralık hem de yağlık zeytinlerin kalitesini doğrudan etkiler. Fungal etmenlerin neden olduğı renk değişimi, çürüme ve koku bozulmaları, zeytinyağında istenmeyen aromalara ve kalite düşüşüne sebep olur [15]. Sonuç olarak, zeytin hastalıkları yalnızca bitkinin fizyolojik sağlığını değıl, aynı zamanda ürünün ekonomik değerini ve üretim sürecinin ekolojik sürdürülebilirliğini tehdit etmektedir. Bu nedenle, hastalık yönetimi; zararlı etmenlerin fenolojik döngüleri, iklimle ilişkileri ve konukçu bitki bağlamında çok boyutlu olarak ele alınmalıdır [16]. Bu bütöncöl yaklaşım, yalnızca patojen kontrolünü değıl, aynı zamanda üretim zincirinin diğör kritik aşamalarında kalite korumasını da içermelidir. Bu bağlamda zeytin hasadı, ürün kalitesi ve sonraki üretim dönemleri açısından kritik bir aşamadır. Hasat, meyvenin kullanım amacı ve olgunluk evresine göre farklı yöntemlerle gerçekleştirilmekte olup yeşil sofralık zeytinler genellikle Eylül-Ekim aylarında, siyah sofralıklar ve yağlık türler ise Kasım-Aralık döneminde toplanmaktadır. Yağ kalitesinin korunması açısından

hasadın zamanında yapılması önemlidir. Ağaç altına düşen meyveler, toprakla temas sonucu yüksek asitli yağlara neden olduğundan ayrı toplanmalı ve farklı değerlendirilmelidir. Geleneksel yerden toplama yöntemi düşük kaliteli yağlara neden olurken, doğrudan ağaçtan elle toplama kalite açısından en uygun yöntemdir ancak yüksek işgücü gereksinimi nedeniyle ekonomik açıdan sınırlayıcıdır [17]. Öte yandan, sırkla silkme ve çırpma gibi yarı mekanik yöntemler, kısa vadede verimli görünmekle birlikte, ağaç dallarında kırılmalara, gelecek yılın sürgünlerine zarar verilmesine ve meyvede zedelenmelere yol açarak hem ürün verimini hem de yağın duyuşal özelliklerini olumsuz etkilemektedir [18].

Bu geleneksel ve yarı mekanik yöntemlere alternatif olarak geliştirilen mekanik hasat teknikleri, zeytin üretiminde işçilik maliyetlerini azaltması, hasat süresini kısaltması ve geniş alanlarda verimli toplama olanağı sunması açısından önemli avantajlar sağlamaktadır. Özellikle süper yoğun (superintensive) dikim sistemlerinde, üzüm hasat makinelerine benzer şekilde çalışan mekanik sistemler, zeytin hücrelerinde oluşan içsel dokusal kırılmalar sayesinde yağ salımını kolaylaştırmakta ve daha yüksek fiziki ekstraksiyon oranlarına sahip zeytinyağlarının elde edilmesine olanak tanımaktadır [19]. Ancak bu yöntemler, meyvede gözle görülmeyen içsel zedelenmelere neden olarak mikrobiyal bulaşmalara karşı duyarlılığı artırmakta ve bu durum serbest asitlik, peroksit sayısı, oksidatif stabilite gibi temel kalite göstergelerinde olumsuz değişimlere yol açabilmektedir. Özellikle zeytin sineği (*Bactrocera oleae*) enfestasyonları sonucu oluşan larva çıkış delikleri ve dokusal bozulmalar ile *Colletotrichum spp.* gibi mantar enfeksiyonlarının neden olduğu yüzey çürükleri ve renk değişimleri, sofralık ve yağlık zeytinlerin kalitesinde ciddi kayıplara neden olmaktadır. Bu tür bozulmuş meyvelerin işlenmesi sonucunda ise raf ömrü kısa, istenmeyen aromalara sahip ve duyuşal nitelikleri düşük yağlar elde edilmektedir [20].

Bu tez kapsamında, zeytin meyvesinde görülen biyotik hastalık belirtilerinin (özellikle *Bactrocera oleae* ısırıkları ve fungal enfeksiyonlar) derin öğrenme temelli yaklaşımlar aracılığıyla otomatik olarak sınıflandırılmasına yönelik kapsamlı bir araştırma yürütülmüştür. Çalışmada, önceden eğitilmiş mimarilerle transfer öğrenme uygulamalarından, dikkat mekanizmalarına dayalı Transformer modellerine ve MetaFormer tabanlı özgün yapıların tasarımına uzanan çok katmanlı bir modelleme süreci izlenmiştir. Tez süresince gerçekleştirilen başlıca deneysel adımlar ve özgün katkılar aşağıda özetlenmiştir:

i) Transfer öğrenme temelli derin öğrenme uygulamaları gerçekleştirilmiştir. Dokuz farklı CNN tabanlı önceden eğitilmiş model (VGG16, ResNet50, InceptionV3, InceptionResNetV2, DenseNet121, Xception, MobileNetV2, NASNetMobile ve EfficientNetB0),

zeytin meyvesi görüntüleri üzerinde transfer öğrenme yöntemiyle yeniden eğitilmiştir. Bu modeller, Hold-Out ve 10 katlı çapraz doğrulama (K-Fold) stratejileri kullanılarak değerlendirilmiştir.

ii) Transformer temelli modellerin sınıflandırma performansları incelenmiştir. Son yıllarda dikkat temelli görsel modeller arasında öne çıkan ViT, DeiT III, BEiT, LeViT, MaxViT ve MobileViT mimarileri, her biri 10 farklı optimizasyon algoritmasıyla eğitilmiş ve sınıflandırma görevindeki başarımları karşılaştırmalı olarak analiz edilmiştir.

iii) MetaFormer mimarisine dayalı hazır modeller test edilmiştir. Literatürde önerilmiş MetaFormer tabanlı CaFormer, ConvFormer ve IdentityFormer modelleri, aynı zeytin meyvesi veri seti üzerinde uygulanmış ve performansları değerlendirilmiştir.

iv) MetaFormer iskelet yapısı temel alınarak özgün hibrit modeller geliştirilmiştir. ConvFormer, RandFormer, IdentityFormer ve Transformer bloklarının farklı dizilimleriyle oluşturulan toplam altı adet yeni MetaFormer tabanlı hibrit mimari tasarlanmıştır. Her bir yapı, sınıflandırma başarımı, mimari verimlilik ve açıklanabilirlik gibi ölçütler açısından kapsamlı biçimde analiz edilmiştir.

v) Yeni bir token mixer modülü tasarlanarak model entegrasyonu sağlanmıştır. Lokal yoğunluk ve kenar duyarlılığına dayalı skor üretimi gerçekleştiren *DeformationScore* isimli yeni bir token mixer geliştirilmiş ve bu modül MetaFormer iskeletine entegre edilerek *DeformationScoreFormer* adlı özgün bir model sunulmuştur.

vi) Tüm modellerin çıktıları çok boyutlu biçimde analiz edilmiştir. Her bir model, doğruluk oranı, F1 skoru, karışıklık matrisi, eğitim ve kayıp eğrileri gibi performans metrikleri aracılığıyla değerlendirilmiş ve modeller arası başarı farkları görsel ve istatistiksel olarak raporlanmıştır.

vii) Elde edilen bulgular, tarımsal uygulamalar ve karar destek sistemleri bağlamında tartışılmıştır.

1.2 Tez İçeriği

Bu tez çalışmasının organizasyonu aşağıdaki gibi maddeler halinde sunulmuştur:

- Birinci Bölüm: Araştırmanın problemi tanımlanmış, çalışmanın amacı, kapsamı ve sınırlılıkları ortaya konmuştur. Ayrıca izlenen yöntemsel yaklaşım ve çalışmanın özgün yönlerine ilişkin genel bir çerçeve çizilmiştir.

- İkinci Bölüm: Kullanılan materyaller detaylandırılmış, veri setinin oluşturulma süreci, zeytin meyvesi görüntülerinin elde edilme koşulları ve görüntü işleme adımları (segmentasyon, morfolojik işlemler, yerleştirme vb.) açıklanmıştır. Ayrıca kullanılan CNN, Transformer ve MetaFormer mimarileri ile önerilen hibrit yapılar teknik olarak tanıtılmıştır.
- Üçüncü Bölüm: Bu bölümde, farklı derin öğrenme modelleriyle yürütülen deneyler sonucunda elde edilen sınıflandırma performanslarının nicel bulguları sunulmuştur. Doğruluk, F1 skoru, karışıklık matrisi ve eğitim/kayıp eğrileri gibi metrikler kullanılarak model başarımı kapsamlı şekilde değerlendirilmiştir. Elde edilen sonuçlar doğrultusunda, uygulanan model mimarilerinin güçlü ve zayıf yönleri ile kullanılan yapıların sınırlı veri üzerindeki etkileri tartışılmıştır. Bölüm sonunda, genel bir değerlendirme yapılmış; çalışmanın öne çıkan bulguları özetlenmiş ve geliştirilen yöntemlerin uygulama potansiyelleri vurgulanmıştır. Ayrıca, gelecekte yapılabilecek geliştirmeler ve araştırmalara yönelik önerilere de yer verilmiştir.

1.3 Tezin Amacı ve Katkısı

Bu çalışmanın temel amacı, zeytin meyvesinde yaygın olarak karşılaşılan bazı hastalıkların görsel veriler üzerinden derin öğrenme yöntemleriyle sınıflandırılabilirliğini araştırmaktır. Bu kapsamda, çeşitli önceden eğitilmiş CNN ve Transformer mimarileri değerlendirilmiş; ayrıca Metaformer iskeletine dayalı altı yeni hibrit yapı önerilmiştir. Çalışmanın özgün katkılarından biri, DeformationScore adlı yeni bir token karıştırıcı modülünün geliştirilmesi ve bu modülün MetaFormer yapısına entegre edilerek yeni bir mimarinin ortaya konmasıdır.

Deneysel çalışmalarla elde edilen sonuçların, benzer sınıflandırma problemlerine yönelik gelecekteki çalışmalarda model seçimi ve mimari tercihleri açısından yol gösterici olabileceği düşünülmektedir.

1.4 Literatür Taraması

Son yıllarda zeytin tarımında derin öğrenme ve bilgisayarlı görü tekniklerinin kullanımına yönelik çeşitli önemli çalışmalar yapılmıştır. Örneğin, Chacón-Maldonado vd. (2025), zeytin sineği (*Bactrocera oleae*) popülasyon tahmini için otomatik hiperparametre optimizasyonlu bir autoencoder ve derin ileri beslemeli sinir ağından oluşan hibrit bir model geliştirmiştir. Uydu

görüntüleri, bitki örtüsü indeksleri ve meteorolojik verilerle eğitilen sistem, SHAP analiziyle yorumlanabilir hâle getirilmiş ve klasik yöntemlere kıyasla %9,7–61,8 aralığında daha düşük hata oranları sunmuştur [21]. Benzer şekilde, Miho vd. (2024), zeytin meyvesindeki *Colletotrichum spp.* enfeksiyonunun fenolik bileşikler üzerindeki etkisini incelemiş, dirençli çeşitlerde aldehitik ve demetillenmiş fenollerin baskın olduğunu ortaya koymuştur [22].

Mahmood ve Alsalem (2024) ise wavelet tabanlı ön işleme ve EfficientNet-B7, DenseNet-201, ResNet-152V2 gibi modellerle öznetelik füzyonu uygulayarak %99,72 doğruluk elde etmiştir [23]. Hassan vd. (2024), modifiye edilmiş K-Means kümeleme ve hibrit ANN-SVM sınıflandırıcı kullanarak konveyör bant üzerindeki 15.000 meyveyi %97,25 doğrulukla sınıflandırmıştır [24].

Hastalık teşhisi konusunda Diker vd. (2024), halkalı leke hastalığına yönelik oluşturdukları görüntü veri kümesinde ResNet101 ve MobileNet ile derin öznetelik çıkarımı yapmış, geleneksel sınıflandırıcılarla %98,63 doğruluk sağlamıştır [25]. Macías-Macías vd. (2023), Mask R-CNN ile sofralık zeytinlerde olgunluk ve hasar tespitini %93,5 doğrulukla gerçekleştirerek kalite kontrol süreçlerini iyileştirmiştir [26]. El Akhal vd. (2023) ise EfficientNetB0 ve lojistik regresyon kombinasyonlu otuz farklı hibrit modelle %96,14 doğruluk oranına ulaşmıştır [27].

Dikkat mekanizmalarının performansa etkisini inceleyen Alshammari vd. (2022), Vision Transformer (ViT) ve CNN füzyonuyla zeytin yaprak hastalıklarında %96 (çok sınıflı) ve %97 (ikili sınıflı) doğruluk elde etmiştir [28]. Mimari füzyonun önemini vurgulayan Ksibi vd. (2022), MobileNet ve ResNet50 tabanlı MobiRes-Net modeliyle %97,08 başarımlı sağlamıştır [29]. Raouhi vd. (2022), 5.571 zeytin yaprağı görüntüsü üzerinde yaptığı karşılaştırmalı çalışmada MobileNet ve RMSprop kombinasyonunun en yüksek performansı sunduğunu belirtmiştir [30].

Veri artırma tekniklerinin etkinliğini araştıran Dikici vd. (2022), AlexNet modelinde %7,56'lık bir doğruluk artışı gözlemlemiştir [31]. Sinha ve Shekhawat (2020), histogram eşikleme ve doku öznetelikleriyle benzer görünümlü hastalıkların ayırt edilebileceğini kanıtlamıştır [32]. Alruwaili vd. (2019), transfer öğrenme ve veri çoğaltma yöntemleriyle sınırlı veriyle yüksek performanslı sınıflandırma yapılabileceğini göstermiştir [33].

Erken dönem çalışmalardan Haff vd. (2016), X-ışını görüntüleme ve Bayes tabanlı bir algoritmayla *Bactrocera oleae* istihlasını tespit ederek sağlam meyvelerde %90 doğruluk sağlamıştır [34]. Kuncan vd. (2013), HSV ve RGB renk uzayına dayalı bir görüntü işleme sistemi geliştirerek endüstriyel otomasyona katkıda bulunmuştur [35]. Riquelme vd. (2008), yapay görme tabanlı bir sistemle sekiz kalite sınıfını %75–97 doğrulukla ayırt etmiş [36], Ayora-Cañada vd. (2005) ise FT-NIR spektroskopisiyle tahribatsız analiz yaparak %92 doğruluk oranına ulaşmıştır [37].

Literatüre baktığımızda önemli besin ürünü olan pirinç bitkisi hastalıkları yanı sıra kuru üzüm[38], patates[39], mısır [40], kayısı[41], yağ gülü[42], domates[43] gibi çeşitli hastalıklarının tespiti ve sınıflandırılması hatta meyve ve sebzelerin tazelik durumu[44] gibi konulara derin öğrenme teknikleriyle ilgili çalışmalar mevcuttur.

Bu kapsamda, zeytin meyvesi ve yaprağına yönelik gerçekleştirilen çeşitli derin öğrenme ve görüntü işleme temelli çalışmalar, yöntemsel yaklaşımları, kullanım amaçları ve elde edilen başarımları bakımından karşılaştırmalı olarak Tablo 1.1’de özetlenmiştir.

Tablo 1.1: Zeytin Tarımında Derin Öğrenme ve Görüntü İşleme Yöntemlerine Dayalı Literatür Özeti

Yazar (Yıl)	Yöntem / Model	Konu / Uygulama	Başarım	Açıklama
Chacón-Maldonado vd. (2025)	Autoencoder + DNN	Zeytin sineği popülasyon tahmini	%9,7–61,8 daha düşük hata	SHAP ile açıklanabilirlik
Miho vd. (2024)	Kimyasal analiz	Colletotrichum spp. etkisi	Nitel sonuç	Fenolik bileşik dağılımı
Mahmood ve Alsalem (2024)	Wavelet + EfficientNet-B7, DenseNet-201, ResNet-152V2	Öznitelik füzyonu ile sınıflandırma	%99,72	Çoklu model entegrasyonu
Hassan vd. (2024)	Modifiye K-Means + ANN-SVM	Konveyör bantta sınıflandırma	%97,25	Gerçek zamanlı uygulama
Diker vd. (2024)	ResNet101, MobileNet + klasik sınıflayıcı	Halkalı leke teşhisi	%98,63	Derin öznitelik çıkarımı
Macías-Macías vd. (2023)	Mask R-CNN	Olgunluk ve hasar tespiti	%93,8	Kalite kontrol otomasyonu
El Akhal vd. (2023)	EfficientNetB0 + LR (30 model)	Hibrit modelleme ile sınıflama	%96,1	Model çeşitliliği
Alshammari vd. (2022)	ViT + CNN	Zeytin yaprak hastalıkları	%97	CNN, ViT ve hibrit modellerin karşılaştırması
Ksibi vd. (2022)	MobiRes-Net (MobileNet + ResNet50)	Yaprak hastalıkları teşhisi	%99,1	Füzyon mimarisi

Raouhi vd. (2022)	MobileNet + RMSprop	Yaprak görüntüleri sınıflandırması	%99,46	Optimizer karşılaştırması
Dikici vd. (2022)	AlexNet	Veri artırma etkisi	+%7,56	Augmentasyon katkısı
Sinha ve Shekhawat (2020)	Histogram eşikleme + doku öznelilikleri	Benzer hastalıkların ayrımı	Nitel başarı	İkili görüntü ayrımı
Alruwaili vd. (2019)	Transfer learning + augmentasyon	Az veride sınıflama	Yüksek doğruluk	Veri çoğaltma katkısı
Haff vd. (2016)	X-ray + Bayes	Zeytin sineği tespiti	%90	Tahribatsız değerlendirme
Kuncan vd. (2013)	HSV ve RGB uzayı	Görüntü işleme ile sınıflama	Nitel sonuç	Endüstriyel katkı
Riquelme vd. (2008)	Yapay görme	Kalite sınıfı ayrımı	%75–97	Sekiz sınıflı değerlendirme
Ayora-Cañada vd. (2005)	FT-NIR spektroskopisi	Tahribatsız analiz	%92	Spektral temelli

Tablo 1.1 incelendiğinde, zeytin tarımında gerçekleştirilen yapay zekâ temelli çalışmaların büyük ölçüde yaprak hastalıklarının teşhisi ve kalite sınıflandırması üzerine yoğunlaştığı görülmektedir. Öte yandan doğrudan zeytin meyvesine odaklanan ve sinek ısırığı ile mantar enfeksiyonlarının birlikte sınıflandırıldığı çok sınıflı çalışmalara literatürde sınırlı sayıda rastlanmaktadır. Bu durum, meyve temelli özgül hastalıkların tespiti için derin öğrenme yöntemlerinin henüz yeterince araştırılmadığını ortaya koymaktadır.

Bu bağlamda, yürütülen bu tez çalışması; zeytin meyvesine özgü iki farklı hastalık semptomunu (*Bactrocera oleae* kaynaklı sinek ısırığı ve *Colletotrichum spp.* kaynaklı mantar enfeksiyonu) tek bir sistem içinde sınıflandırmayı hedefleyen, farklı derin öğrenme mimarilerinin (CNN, Transformer ve MetaFormer tabanlı) performanslarını karşılaştırmalı olarak değerlendiren bir yaklaşım sunmaktadır.

2. BÖLÜM: MATERYAL VE YÖNTEM

2.1 Veri Seti Tanımı ve Görüntü Toplama Süreci

Bu çalışmada kullanılan görüntü verileri, Balıkesir ili Bandırma ilçesine bağlı Edincik bölgesinde yetiştirilen zeytin ağaçlarından elde edilmiştir. Görüntülerin temin edildiği zeytin meyveleri, Marmarabirlik kooperatifine sofralık ve yağlık olarak teslim edilen ürünlerin toplandığı ağaçlardan seçilmiş olup, saha çalışmaları bir ziraat mühendisi eşliğinde titizlikle yürütülmüştür. Ekim ayı içerisinde gerçekleştirilen bu saha çalışması kapsamında, farklı gelişim evrelerine ve sağlık durumlarına sahip toplam 1644 adet zeytin meyvesi toplanarak görüntülenmiştir.

Aşağıda sunulan Şekil 2.1, veri setinin oluşturulması sürecinde sahada yürütülen görüntüleme çalışmalarından bazı anlık kareleri içermektedir. Bu görseller, veri çeşitliliğini sağlamak amacıyla farklı ağaçlardan ve çeşitli sağlık durumlarındaki zeytin meyvelerinden örnekler toplandığını göstermektedir.



Şekil 2- 1: Veri seti için zeytin meyvesi toplama süreci

Toplanan zeytinler, hastalık belirtilerine göre etiketlenmiş ve üç temel sınıfa ayrılmıştır: sağlıklı meyveler, zeytin sineği zararlısı tarafından hasar görmüş meyveler, ve mantar enfeksiyonuna maruz kalmış meyveler. Bu sınıflama hem görsel belirtilere hem de ziraat mühendisinin uzman görüşüne dayalı olarak gerçekleştirilmiştir. Sınıflandırılmış veriler, sonraki aşamalarda kullanılmak üzere dijital ortama aktarılmış ve analiz için hazırlanmıştır.

2.2 Görüntü İşleme ve Ön İşleme

Zeytin örnekleri üç temel kategoriye ayrılarak analiz edilmiştir: sinek zararına uğramış (275 çift görüntü), mantar hastalığına sahip (287 çift görüntü) ve sağlıklı (292 çift görüntü) örnekler. Görüntüleme süreci, tutarlı ve kontrol edilebilir bir ışıklandırma ortamı sağlamak üzere özel olarak tasarlanmış kapalı bir kutu içerisinde gerçekleştirilmiştir. Kutu iç yüzeyine yerleştirilen LED ışık şeritleri sayesinde homojen aydınlatma sağlanmış, dış ortamdan gelen ışık parazitleri tamamen engellenmiştir. Kamera, zeytinlerin tam üstüne gelecek şekilde 52 cm yükseğe konumlandırılarak her bir numunenin belirli ve sabit bir açıdan görüntülenmesi sağlanmıştır. Zeytin meyveleri ve yaprakları kutu içerisine rastlantısal olarak yerleştirilmiş, bu sayede görüntülerdeki konumsal sapmalardan doğabilecek yanlışlıklar önlenmiştir. Şekil 2-2’de, zeytin meyvelerinin görüntülenmesi amacıyla özel olarak tasarlanmış kapalı kutunun yapısı sunulmaktadır. Elde edilen tüm görseller dijital ortama otomatik biçimde aktarılmış ve ilerleyen aşamalarda yürütülecek olan ön işleme ile sınıflandırma işlemleri için kullanılmaya hazır hâle getirilmiştir.



Şekil 2- 2: Zeytin meyvelerinin sınıflandırılması amacıyla kullanılan kapalı görüntüleme kutusu

Bu çalışmada zeytin meyvesine ait görüntüler, sınıflandırma öncesinde çeşitli görüntü işleme tekniklerine tabi tutulmuştur. İlk olarak, RGB (BGR) formatındaki renkli görüntüler, renk bilgisini yoğunluk bilgisine indirgemek amacıyla gri tonlamaya çevrilmiş, ardından Otsu yöntemiyle ikili formata dönüştürülerek segmentasyon işlemi gerçekleştirilmiştir. Histogram tabanlı bu yöntem, sınıf içi varyansı minimize eden optimal eşik değerini otomatik olarak belirlemekte olup [45] çalışmada nesnenin arka plandan daha açık tonlarda olduğu için

THRESH_BINARY_INV eşikleme tercih edilmiştir. Böylece meyve alanı beyaz, arka plan siyah olarak elde edilmiştir. İkili görüntüdeki boşluklar flood-fill algoritmasıyla doldurulmuş, böylece nesneler daha kompakt bir hâle getirilmiştir [46]. Ardından, küçük boşlukların ve gürültülerin giderilmesi için morfolojik kapatma işlemi uygulanmış; genişletme ve aşındırma adımlarının birleşimiyle daha düzgün nesne sınırları sağlanmıştır [47]. Segmentasyon sonrası bağlantılı bileşen analizi (connected component analysis) yapılarak, görüntüdeki her nesnenin sınırları ve topolojik ilişkileri belirlenmiş, böylece birden fazla zeytin bulunan karelerde yalnızca hedef nesneye odaklanılması sağlanmıştır [48]. Son olarak, tespit edilen meyve nesneleri sınırlayıcı kutular (bounding box) çerçevesinde kırpılarak ayrı görüntü dosyaları olarak kaydedilmiştir. Bu kutular, nesnenin görüntüdeki konumunu belirlemek amacıyla kullanılan ve bilgisayarla görme uygulamalarında standart olan yapılar olup nesne yerleştirme ve sınıflandırma süreçlerinde kritik rol oynamaktadır [49].

Şekil 2-3'te, açıklanan bu iş akışının tamamlanmasıyla elde edilen ve sınıflandırma modellerinin eğitimi için kullanılmaya hazır hâle getirilen tekil zeytin meyvesi örneklerine ait görseller sunulmaktadır. Hazırlanan bu görseller, derin öğrenme tabanlı modellerin öğrenme sürecinde yüksek düzeyde sınıf ayrımı, görsel tutarlılık ve standartlaştırılmış çerçeveleme sağlayacak biçimde optimize edilmiştir.



Şekil 2- 3: Segmentasyon ve ön işleme süreci sonrası elde edilen, sınıflandırma için kullanılmaya hazır örnek zeytin meyvesi görselleri (soldan sağa: sinek ısırıklı, mantar hastalıklı, sağlıklı).

2.2.1 Evrişimsel Sinir Ağları

Evrişimsel sinir ağları (CNN), özellikle görsel veri üzerinde öznitelik çıkarımı ve sınıflandırma görevlerinde yaygın biçimde kullanılan derin öğrenme mimarileridir. Bu yapılar, geleneksel tam bağlantılı sinir ağlarından farklı olarak, yerel bağlantılılık ve ağırlık paylaşımı prensiplerine dayanır. Bu sayede, yüksek boyutlu görüntü verilerinde parametre sayısını önemli ölçüde azaltarak hem hesaplama verimliliği sağlar hem de öğrenme sürecini derinleştirir [50, 60].

CNN mimarileri, ardışık olarak düzenlenmiş dört temel katmandan oluşur: evrişim (convolution), aktivasyon (non-linearity), ortaklama (pooling) ve tam bağlantılı (fully connected) katmanlar [52]. Evrişim katmanları, giriş görüntüsü üzerinde kaydırılan filtreler yardımıyla belirli öznelilikleri (kenar, köşe, desen gibi) algılar. Bu katmanların ürettiği özellik haritaları (feature maps), görüntüdeki yerel örüntülerin öğrenilmesini mümkün kılar [53].

Evrişim sonrası aktivasyon fonksiyonu uygulanarak modele doğrusal olmayanlık kazandırılır; bu, ağın daha karmaşık ilişkileri öğrenmesini sağlar [54]. Ortaklama katmanları ise elde edilen özellik haritalarını belirli oranlarda alt örnekleyerek boyutlarını küçültür, böylece hem hesaplama maliyetini azaltır hem de modelin gürültüye karşı daha dayanıklı olmasını sağlar [55].

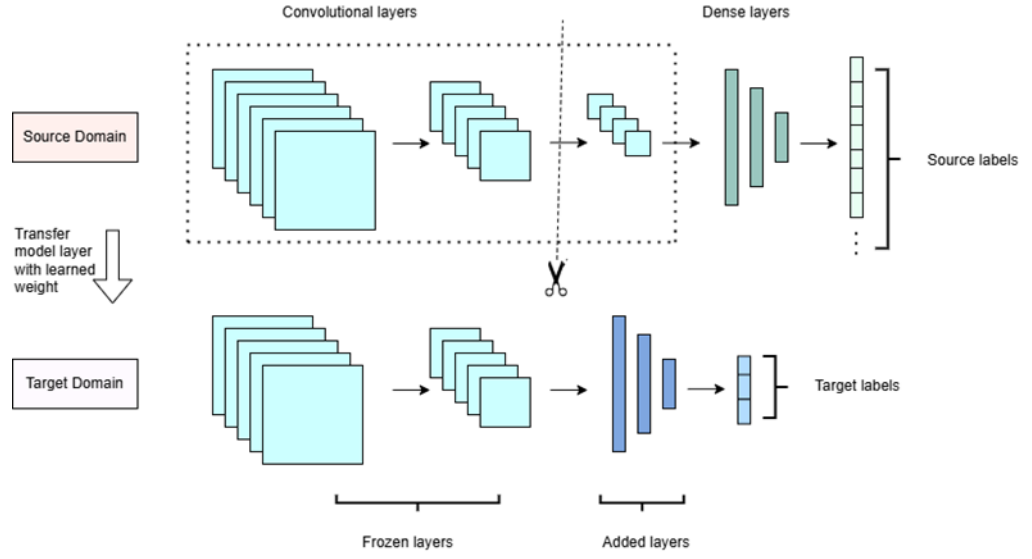
Son katman olan tam bağlantılı katman, önceki katmanlardan elde edilen soyut temsilleri bir araya getirerek nihai sınıflandırma kararının verilmesini sağlar. Bu bölüm genellikle softmax gibi bir aktivasyon fonksiyonu ile tamamlanarak, modelin her sınıf için bir olasılık değeri üretmesi sağlanır [50].

CNN mimarilerinin başarısı, özellik çıkarımının öğrenilebilir filtrelerle otomatikleştirilmesine ve çok katmanlı temsiller yoluyla karmaşık görsel yapıların hiyerarşik olarak modellenmesine dayanır [56]. Bu yapı sayesinde, CNN'ler yalnızca sınıflandırma görevlerinde değil; aynı zamanda nesne algılama, segmentasyon, yüz tanıma, medikal görüntü analizi ve tarımsal hastalık tespiti gibi çok sayıda uygulama alanında yüksek başarıyla kullanılmaktadır [53, 65].

2.3 CNN Tabanlı Transfer Öğrenme Modelleri

Transfer öğrenme (transfer learning), bir görevde edinilen bilginin, benzer özellikteki başka bir göreve aktarılması prensibine dayanan makine öğrenmesi yöntemidir [59]. Derin öğrenme alanında yaygın olarak benimsenen bu yöntem, geniş ölçekli veri kümeleri üzerinde önceden eğitilmiş sinir ağı modellerinin, sınırlı veriye sahip özel problem alanlarına adaptasyonunu mümkün kılmaktadır. Bu yaklaşım sayesinde model eğitim süreci önemli ölçüde kısalmakta, düşük veri hacimlerinde dahi yüksek genelleme performansı elde edilebilmekte ve hesaplama kaynakları optimize edilebilmektedir [60].

Özellikle görsel sınıflandırma problemlerinde transfer öğrenme, özellik çıkarımı için temel katmanların yeniden kullanılmasını mümkün kılar; yalnızca son katmanlar, hedef sınıflara göre güncellenerek yeniden eğitilir [61]. Bu mimari yaklaşım Şekil 2-4'te şematik olarak özetlenmiştir.



Şekil 2- 4: Transfer öğrenme mimarisinin genel yapısı

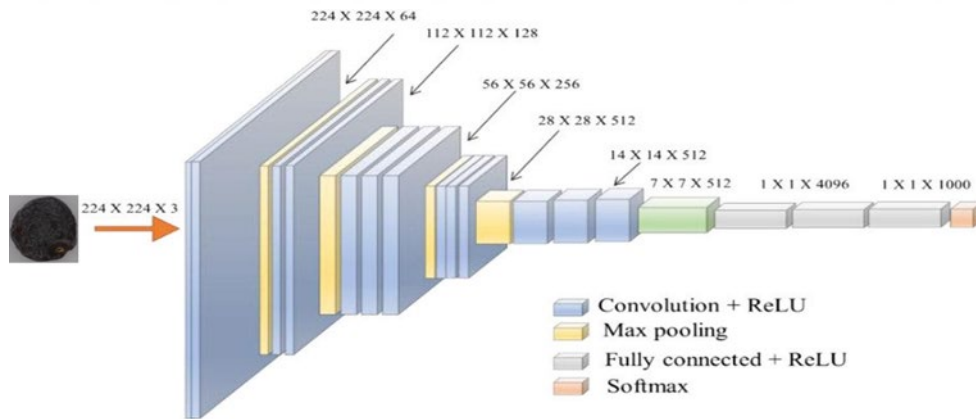
Bu yaklaşım etiketli veri sayısının sınırlı olduğu ya da görüntü elde etmenin zor olduğu alanlarda kritik avantajlar sunmaktadır. Tarım, bu alanların başında gelmektedir. Çünkü birçok tarımsal problemde (örneğin bitki hastalıkları, zararlı tespiti, ürün kalitesi izleme vb.) görsel veri toplama süreçleri zaman alıcı, mevsimsel ve çoğu zaman uzmanlık gerektiricidir. Ayrıca doğal ışık, meyve durumu, arka plan ve çevresel koşullar gibi değişkenler, sıfırdan model eğitimi için gereken veri bütünlüğünü zorlaştırmaktadır. Bu nedenle transfer öğrenme, tarımsal görüntü işleme uygulamalarında hem araştırma sürecini hızlandırmakta hem de daha küçük veri kümeleriyle güvenilir sonuçlar üretmeyi mümkün kılmaktadır. Son yıllarda yapılan çalışmalar, transfer öğrenmenin bu alandaki performansını desteklemekte; önceden eğitilmiş CNN tabanlı modellerin, çeşitli tarım ürünlerine ait sınıflandırma problemlerinde başarılı biçimde uygulandığını göstermektedir [8, 65, 69].

Bu doğrultuda, tez kapsamında tarımsal görüntü sınıflandırma görevine uyarlanmak üzere, önceden ImageNet veri kümesi üzerinde eğitilmiş olan çeşitli CNN tabanlı transfer öğrenme modelleri değerlendirilmiştir. Bu modeller arasında MobileNet, NASNetMobile, DenseNet121, Xception, VGG16, ResNet50, InceptionV3, EfficientNetB0 ve InceptionResNetV2 mimarileri yer almakta olup, her biri farklı yapısal özellikleri ve hesaplama gereksinimleri doğrultusunda karşılaştırmalı olarak analiz edilmiştir.

2.3.1 VGG16

VGG16, Oxford Üniversitesi Görsel Geometri Grubu tarafından geliştirilmiş ve Simonyan ile Zisserman (2014) tarafından tanıtılmış derin konvolüsyonel sinir ağı mimarisidir. Model, ImageNet Büyük Ölçekli Görüntü Tanıma Yarışması (ILSVRC-2014) sırasında yüksek doğruluk elde ederek bilgisayarla görü alanında temel modellerden biri olarak kabul edilmiştir. Mimari, toplamda 16 ağırlıklı katmandan oluşmakta ve esas olarak ardışık 3×3 boyutundaki konvolüsyon filtrelerinin tekrarı prensibine dayanır. Bu tasarım, modelin derinliğini artırırken parametre paylaşımı yoluyla hesaplama verimliliğini korumaya olanak tanır [63].

Şekil 2-5'te modelin genel mimarisi verilen VGG16, giriş görüntüsünü beş ardışık konvolüsyon bloğundan geçirir; ilk iki blokta her biri iki adet, sonraki üç blokta ise üç adet 3×3 filtreli konvolüsyon katmanı ve ardından max-pooling işlemi uygulanır. Bu yapı, temel görsel özelliklerin giderek daha soyut temsillere dönüştürülmesini sağlar. Ardından üç tam bağlantılı katman ve son olarak softmax aktivasyonlu sınıflandırıcı gelir. VGG16'nın toplam parametre sayısı 138 milyondan fazla olup, yüksek hesaplama kaynağı gerektirir ancak mimarisinin sadeliği ve tekrarlayan yapısı, transfer öğrenme uygulamalarında yaygın şekilde tercih edilmesini sağlar [71, 72].



Şekil 2- 5: VGG16 modelinin genel mimari yapısı

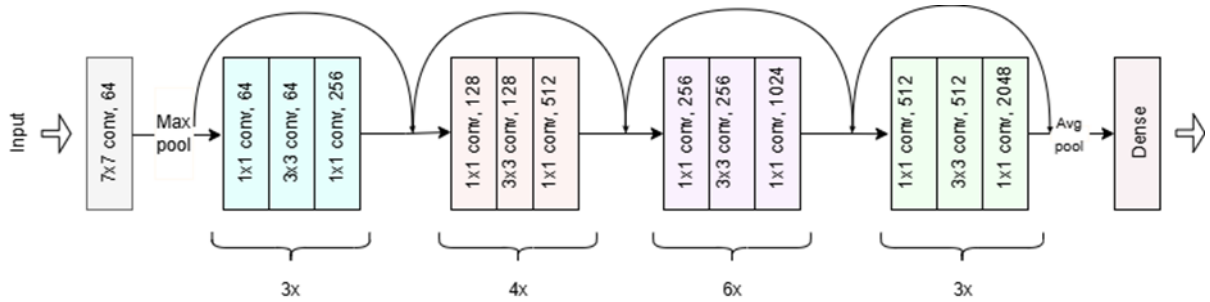
Transfer öğrenme süreci, konvolüsyon bloklarından elde edilen $7 \times 7 \times 512$ boyutundaki öznetelik haritasının çıkarılmasının ardından başlar. Orijinal modelde bu noktadan sonra gelen üç tam bağlantılı katman (ikisi 4096 nöron içerir) ve softmax sınıflandırıcı kaldırılır (`include_top=False`) ve yerine hedef görev için daha hafif sınıflandırıcılar eklenir. Modelin yeniden eğitimi ise yalnızca bu üst katmanlar üzerinde gerçekleştirilir [73, 74]. Böylelikle,

konvolüsyon blokları sabitlenirken sınıflandırıcılar hedef göreve uyarlanır. Bu durum eğitim süresini kısaltır ve sınırlı veriyle aşırı uyumu önlemeye yardımcı olur [61].

2.3.2 ResNet50

ResNet50, derin sinir ağlarında öğrenme zorluklarını aşmak amacıyla He ve arkadaşları (2016) tarafından geliştirilen Residual Network (ResNet) ailesinin 50 katmanlı bir versiyonudur. Bu mimarinin temel yeniliği, derin ağlarda karşılaşılan aşırı uyum (overfitting) ve gradyan kaybolması (vanishing gradient) problemlerini azaltmak için artık öğrenme (residual learning) mekanizmasını kullanmasıdır. ResNet, klasik konvolüsyonel katmanların yanı sıra “skip connection” olarak adlandırılan atlama bağlantılarına sahiptir. Bu bağlantılar, her katmanın doğrudan çıktı üretmek yerine bir önceki katmandan gelen bilgiyi değiştirerek modellemesini sağlar ve böylece öğrenme sürecinin daha etkin gerçekleşmesine olanak tanır [68].

Toplam 50 katmandan oluşan ResNet50 mimarisi, 1×1 , 3×3 ve tekrar 1×1 boyutundaki konvolüsyonlardan oluşan “bottleneck” bloklarının ardışık diziliminden meydana gelir. Bottleneck yapısı, derin ağların daha verimli eğitilmesini sağlarken, parametre sayısını ve hesaplama maliyetini önemli ölçüde azaltır. Her blokta orijinal girişin kopyası öğrenilen dönüşüme doğrudan eklenerek bilgi kaybı en aza indirilir ve optimizasyon süreci hızlandırılır [69]. Bu yapısal düzen, Şekil 2-6’da blok bazında şematik olarak gösterilmiştir.



Şekil 2- 6: ResNet50 mimarisinin blok temelli yapısı

Transfer öğrenme bağlamında ResNet50, yüksek genelleme kapasitesi ve esnek modüler yapısı nedeniyle çok sayıda görsel görevde tercih edilmektedir. Modelin erken konvolüsyonel katmanları, kenar, doku ve renk varyasyonu gibi genel görsel özellikleri yakaladığından genellikle sabitlenir ve yalnızca son sınıflandırma katmanları hedef veri setine göre yeniden eğitilir [60]. Ayrıca, ResNet50 mimarisi katman dondurma, ince ayar (fine-tuning) veya ara katmanlardan

öznitelik çıkarımı gibi farklı transfer öğrenme stratejilerine kolayca uyarlanabilir [61]. Bu çalışmada önceden eğitilmiş ResNet50 modelinin tüm konvolüsyonel blokları sabitlenmiş ve sadece sınıflandırıcı kısmı yeniden yapılandırılarak eğitim süreci gerçekleştirilmiştir.

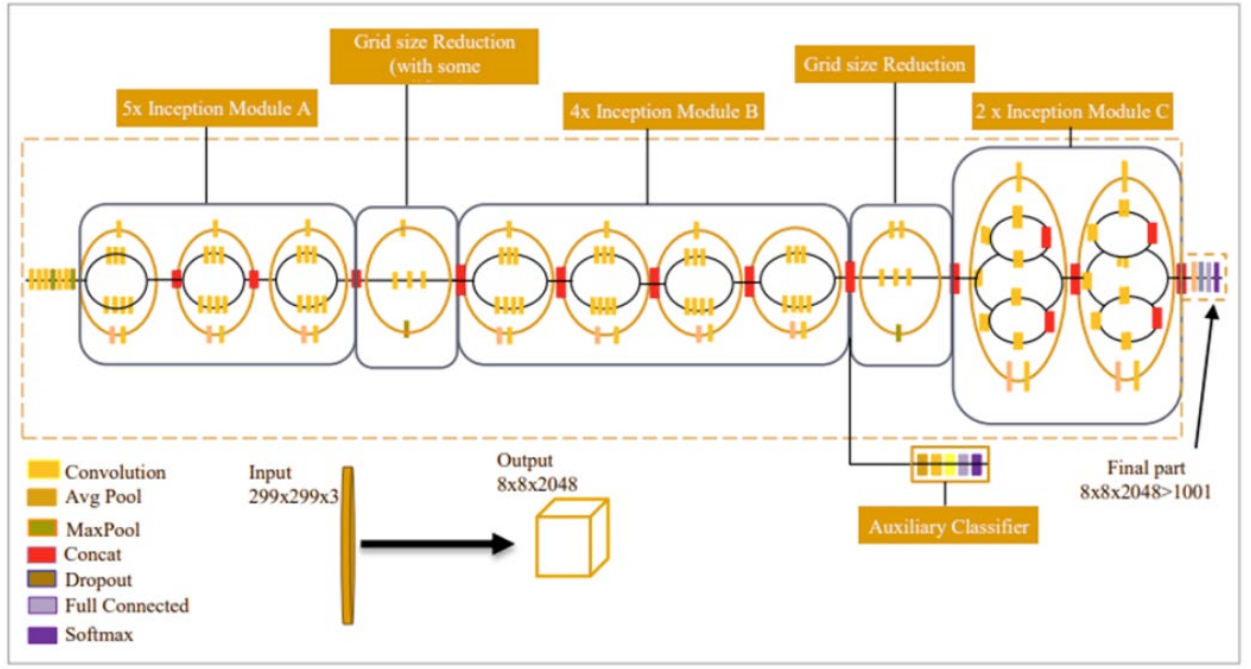
2.3.3 InceptionV3

InceptionV3, Google Brain ekibi tarafından geliştirilmiş ve ilk olarak Szegedy ve arkadaşları (2015, 2016) tarafından tanıtılmış derin bir konvolüsyonel sinir ağı mimarisidir. Model, yüksek sınıflandırma doğruluğunu korurken hesaplama maliyetini azaltmaya yönelik yapısal yenilikleri ile ön plana çıkmaktadır. Inception mimarisi, farklı boyutlardaki (1×1 , 3×3 , 5×5) konvolüsyon filtrelerini paralel olarak kullanarak çok yönlü öznitelik çıkarımı sağlar. InceptionV3'te ise, bu yapı daha modüler ve hesaplı hale getirilmiştir [70].

Büyük boyutlu konvolüsyonlar yerine ardışık iki 3×3 filtre kullanımı ve 3×3 filtrelerin 1×3 ile 3×1 boyutlarında ayrıştırılması (asymmetric factorization), işlem verimliliğini artırırken yönsel bilgilerin çıkarılmasına katkı sağlamaktadır [71].

Modelde ayrıca batch normalization kullanılarak içsel dağılım değişimi minimize edilmiş ve eğitim süreci stabilize edilmiştir [72]. Derin katmanlarda öğrenmeyi desteklemek için auxiliary classifier yapıları eklenmiş ve aşırı güvenli tahminlerin neden olduğu overfitting riski label smoothing yöntemiyle azaltılmıştır [80, 81, 82].

InceptionV3'ün temel bileşenleri, çok ölçekli Inception modülleri (A, B ve C) ve bu modüller arasında yer alan grid size reduction yapılarından oluşur. Bu yapı, hem ayrıştırılmış konvolüsyonlar (factorized convolutions) hem de paralel dallanmış çekirdek kombinasyonları ile çok çeşitli özniteliklerin eş zamanlı olarak çıkarılmasına olanak sağlar. Şekil 2.7'de, modelin temel bileşenleri ve katmanlar arasındaki veri akışı detaylı biçimde gösterilmiştir [76].



Şekil 2- 7: InceptionV3 mimarisinin katman temelli genel yapısı.

Deneysel çalışmalar, InceptionV3'ün faktörize edilmiş yapısı ve optimizasyon teknikleri sayesinde önceki Inception sürümlerine kıyasla yaklaşık %30 daha düşük hesaplama maliyetiyle benzer veya daha yüksek sınıflandırma doğruluğu sağladığını göstermektedir [77, 84].

Keras ve TensorFlow gibi popüler kütüphanelerde, önceden ImageNet veri kümesi üzerinde eğitilmiş InceptionV3 modeli `include_top=False` parametresi ile çağrıldığında orijinal sınıflandırma katmanı çıkarılır ve modelin çıktısı $8 \times 8 \times 2048$ boyutunda bir öznitelik haritası olur. Bu harita genellikle global average pooling katmanı ile $1 \times 1 \times 2048$ boyutuna indirgenir ve hedef sınıflandırıcıya beslenir [70].

Bu çalışmada da benzer bir transfer öğrenme stratejisi izlenmiş; InceptionV3'ün tüm konvolüsyonel blokları sabitlenmiş, sadece sınıflandırıcı katman yeniden yapılandırılarak eğitilmiştir. Bu yaklaşım, özellikle küçük veri kümelerinde aşırı öğrenmeyi önlemekte ve eğitim süresini kısaltmaktadır [67, 74].

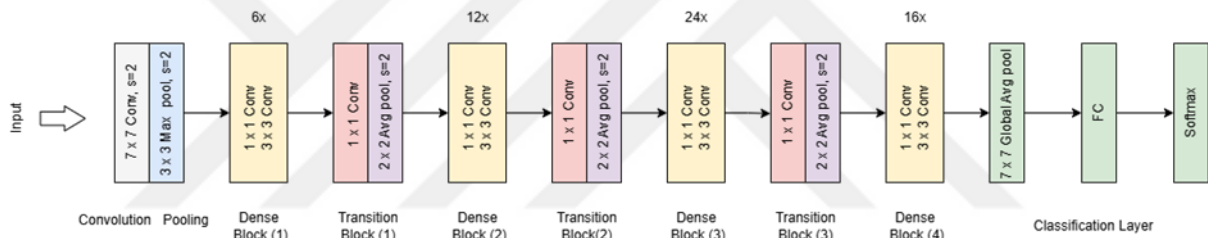
2.3.4 DenseNet121

DenseNet121, her katmanın kendisinden önceki tüm katmanlardan gelen çıktı haritalarını giriş olarak almasını sağlayan yoğun bağlantı prensibi temelinde geliştirilmiş derin bir konvolüsyonel sinir ağıdır. Bu yoğun bağlantı yapısı, derin ağlarda gradyan yayılımını iyileştirerek kaybolma (vanishing gradient) sorununu minimize eder ve böylece öğrenme sürecinin stabilitesini

artırır [78]. Modeldeki dense block yapıları, özelliklerin tekrarlı kullanımını artırarak mimarinin öğrenme kapasitesini genişletir [79].

DenseNet121, dört adet dense block ve bu bloklar arasında yer alan üç transition layer'dan oluşur. Transition layerlar, özellik haritalarının boyutlarının kademeli olarak azaltılmasını ve parametre verimliliğinin sağlanmasını mümkün kılar [80]. Ayrıca, hesaplama maliyetini düşürmek amacıyla her dense block içinde 1×1 konvolüsyon kullanılan bottleneck katmanları bulunur [81].

Modelin mimari akışı, Şekil 2-8'de şematik olarak gösterilmiştir. Şekilde, giriş katmanından sonra sırasıyla dört dense block, aralarındaki transition layer'lar, en sonunda ise global average pooling ve tam bağlantılı sınıflandırıcı katmanlar yer almaktadır [82]. Her dense block, sırasıyla 6, 12, 24 ve 16 adet konvolüsyon katmanından oluşur ve bu yapı modelin toplam 121 katmana ulaşmasını sağlar. Growth rate parametresi, her yeni katmanın eklediği özellik haritası sayısını kontrol altında tutar [83]. Doğrudan bağlantılar sayesinde katmanlar arasında etkin bilgi akışı ve özellik paylaşımı gerçekleştirilerek öğrenme verimliliği artırılır [79].



Şekil 2- 8: DenseNet121 modelinin genel mimari akışı

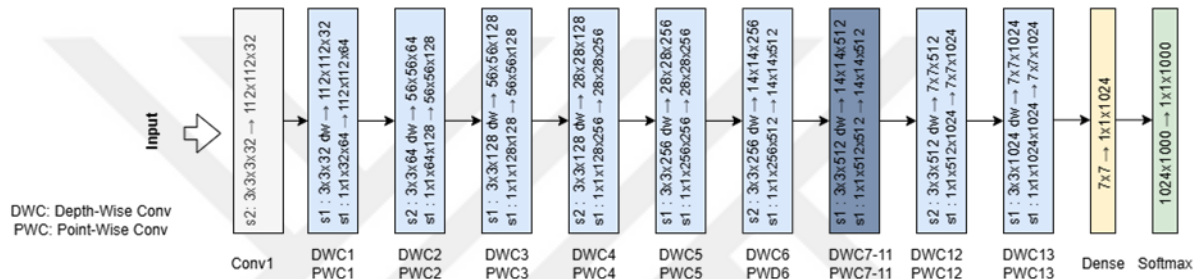
Transfer öğrenme uygulamalarında DenseNet121, genellikle ImageNet gibi büyük veri kümelerinde önceden eğitilmiş ağırlıklar kullanılarak başlatılır ve böylece yeni görevlerde hızlı adaptasyon sağlanır [84]. Bu süreçte, modelin temel konvolüsyon katmanları çoğunlukla dondurulur (freeze edilir) ve yalnızca son sınıflandırıcı katman hedef veri setine göre yeniden eğitilir [85]. Yoğun bağlantı yapısı, transfer öğrenme ile sınırlı veriyle dahi yüksek doğruluk elde edilmesine imkan verir [78]. Ayrıca parametre verimliliği ve hızlı eğitim süresi, DenseNet121'in transfer öğrenmede tercih edilmesini destekleyen önemli unsurlar arasında yer alır [80].

2.3.5 NASNet Mobile

NASNet Mobile, Google tarafından otomatik ağ mimarisi arama (Neural Architecture Search, NAS) yöntemi kullanılarak geliştirilmiş ve özellikle mobil cihazlarda yüksek doğrulukla

Derinlik ayırık konvolüsyon iki bileşenden oluşur: (1) Depthwise convolution, her giriş kanalına ayrı 3×3 filtre uygulayarak uzamsal bilgi çıkarırken; (2) Pointwise convolution olarak adlandırılan 1×1 konvolüsyon, tüm kanalları birleştirerek kanal arası ilişkileri öğrenir. Bu ikili yapı, klasik konvolüsyon işleminin hesaplama maliyetini yaklaşık 8-9 kat azaltmakla beraber benzer doğruluk seviyeleri sağlar [98, 99].

MobileNet'in yapısal akışı, Şekil 2-10'da blok bazında gösterilmiştir. Giriş görüntüsü ilk olarak standart 3×3 konvolüsyon (Conv1) katmanından geçirilerek $112 \times 112 \times 32$ boyutunda öznitelik haritası elde edilir. Bunu takiben ardışık 3×3 derinlik ayırık konvolüsyon (DWC) ve 1×1 noktasal konvolüsyon (PWC) katmanlarından oluşan 13 blok yer alır.



Şekil 2- 10: MobileNetV1 mimarisinin blok temelli yapısı

Her blok, uzamsal özellik çıkarımı ve kanal ilişkilerini modelleyerek $14 \times 14 \times 512$ boyutuna kadar öznitelik haritalarını daraltır. Sonrasında 7×7 boyutundaki alanda global average pooling (GAP) uygulanarak $1 \times 1 \times 1024$ boyutuna indirgeme yapılır. Son olarak, 1024 boyutlu temsil sağlayan tam bağlantılı (Dense) katman ve softmax sınıflandırıcı yer alır.

Mobil cihazlarda hızlı çıkarım gereksinimi ve sınırlı bellek koşulları göz önüne alındığında, MobileNet mimarisi, transfer öğrenme senaryolarında yaygın biçimde kullanılmakta ve birçok görüntü sınıflandırma görevinde yüksek başarı sağlamaktadır.

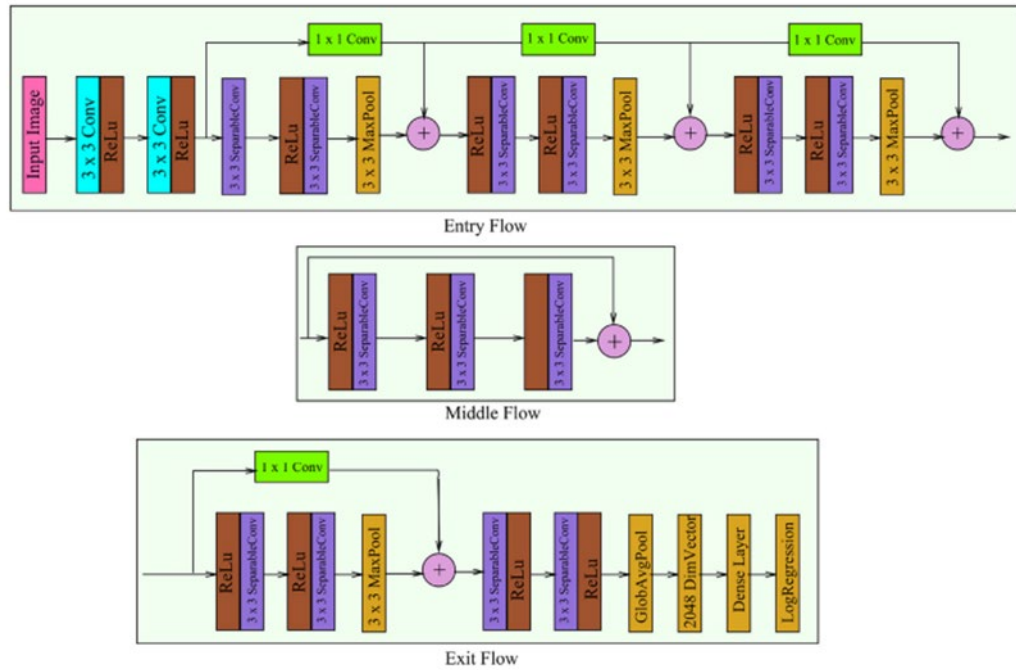
2.3.7 Xception Mimarisi

Xception, derin ayırık evrişim (depthwise separable convolution) temelli yapısıyla klasik Inception mimarilerinden daha hesaplama açısından verimli ve performanslı bir ağ oluşturmak amacıyla geliştirilmiştir [93]. Modelin temel prensibi, standart konvolüsyonların yerine, her uzamsal kanalın ayrı ayrı işlendiği depthwise konvolüsyon ve ardından gelen pointwise (1×1)

konvolüsyonların kullanılmasıdır [91]. Bu yaklaşım, parametre sayısını ve işlem maliyetini önemli ölçüde azaltırken, yüksek doğruluk ve model kapasitesinin korunmasını sağlar [94].

Xception mimarisi, toplam 36 konvolüsyonel katmandan oluşmakta olup, mimari giriş (entry), orta (middle) ve çıkış (exit) akış blokları olarak üç temel bölüme ayrılmıştır [93]. Her blokta, art arda gelen depthwise separable convolution ve batch normalization katmanları birlikte kullanılarak bilgi kaybı minimize edilmektedir [70].

Şekil 2-11’de, Xception mimarisinin blok diyagramı yer almaktadır. Burada modelin girişinden çıkışa kadar Entry Flow, Middle Flow ve Exit Flow blokları ile bu bloklarda kullanılan 3×3 konvolüsyonlar, ReLU aktivasyonları, depthwise separable convolution katmanları ve max pooling işlemleri gösterilmektedir. Ayrıca residual bağlantılar ve modelin sonunda yer alan global average pooling, tam bağlantılı (dense) katman ve softmax sınıflandırıcı diyagram üzerinde net olarak işaretlenmiştir [95].



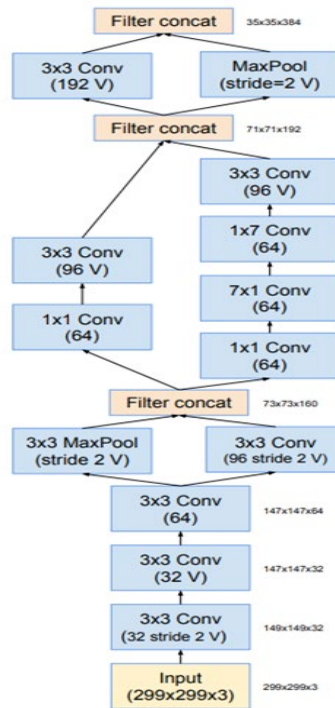
Şekil 2- 11: Xception mimarisinin blok diyagramı

Transfer öğrenme kapsamında Xception, genellikle ImageNet gibi büyük ölçekli veri kümelerinde önceden eğitilmiş ağırlıklarla başlatılır ve sadece son sınıflandırıcı katman hedef veri setine göre yeniden eğitilir.

2.3.8 InceptionResNetV2

InceptionResNetV2, klasik Inception mimarisini residual bağlantılarla birleştirerek daha derin ve verimli bir ağ yapısı ortaya koymak amacıyla geliştirilmiştir [70]. Residual bağlantılar, derin katmanlara gradyanların daha etkin aktarılmasını sağlayarak ağın eğitilebilirliğini artırmakta ve öğrenme sürecindeki zorlukları azaltmaktadır [68]. Inception bloklarının paralel konvolüsyon yolları farklı ölçeklerde zengin özellik çıkarımı sağlarken, residual bağlantıların entegrasyonu öğrenme dinamiklerinin iyileştirilmesine katkıda bulunur [93]. Her bir Inception-ResNet bloğunda giriş ve çıkış arasında kısa yol (skip connection) kullanılarak hem model derinliği artırılmış hem de parametre verimliliği sağlanmaktadır [70].

Şekil 2.12’de InceptionResNetV2 modelinin giriş katmanlarına (stem) ait mimari akış diyagramı sunulmaktadır. Bu şema, modelin ilk evresini oluşturan ardışık 3×3 konvolüsyon katmanları, max pooling işlemleri ve paralel konvolüsyon yollarının (filter concat) yapısını göstermektedir. $299 \times 299 \times 3$ boyutundaki giriş görüntüsü, bu katmanlarda işlem görerek farklı filtre boyutları ile çoklu özellik çıkarımı sağlanmakta ve uzamsal çözünürlük kademeli olarak azaltılmaktadır.



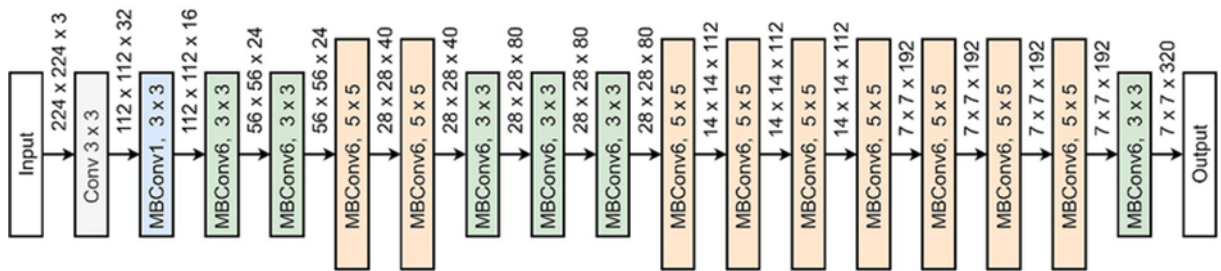
Şekil 2- 12: InceptionResNetV2 modelinin mimari akış diyagramı

Transfer öğrenme uygulamalarında InceptionResNetV2, genellikle ImageNet gibi geniş veri kümelerinde önceden eğitilmiş ağırlıklarla başlatılır ve sadece son sınıflandırıcı katman hedef veri setine göre yeniden eğitilir [96]. Residual bağlantılar sayesinde model, sınırlı veri ile hızlı ve etkili öğrenme sergileyebilmektedir [97].

2.3.9 EfficientNetB0

EfficientNetB0, model doğruluğunu artırırken parametre sayısı ve hesaplama maliyetini minimumda tutmayı amaçlayan ölçeklenebilir bir konvolüsyonel sinir ağı mimarisidir. Google Brain ekibi tarafından geliştirilmiş olup, Tan ve Le (2019) tarafından önerilen “compound scaling” yaklaşımı ile önceki mimarilere kıyasla önemli bir verimlilik sunmaktadır. Bu strateji, ağı derinliği, genişliği ve giriş çözünürlüğünün birlikte dengeli biçimde ölçeklenmesini sağlar ve böylece küçük modellerden büyük modellere geçişte doğruluk artışı ile hesaplama verimliliği aynı anda elde edilir [67].

EfficientNetB0, NAS (Neural Architecture Search) algoritmasıyla optimize edilmiş temel modeldir. Mimari, Mobile Inverted Bottleneck Convolution (MBConv) blokları üzerine kuruludur. Her blokta, genişletme faktörlü 1×1 konvolüsyon, ardından 3×3 derinlik ayırık konvolüsyon ve tekrar 1×1 konvolüsyon ile daraltma katmanı bulunur. Ayrıca, her MBConv bloğunda squeeze-and-excitation (SE) mekanizması yer almakta ve bu sayede kanal bazlı dikkat (channel-wise attention) uygulanmaktadır [74, 105]. Şekil 2-13'te, EfficientNetB0 mimarisinin blok temelli yapısı gösterilmektedir.



Şekil 2- 13: EfficientNetB0 mimarisinin blok temelli yapısı

Model, yedi aşamalı blok yapısından oluşur; bu bloklarda stride ile çözünürlük düşürülür, kanal sayısı artırılır ve Swish aktivasyon fonksiyonu ile batch normalization gibi normalizasyon

teknikleri kullanılır. Son aşamada küresel ortalama havuzlama (GAP) ve tam bağlantılı (dense) sınıflandırıcı katman yer almaktadır.

Transfer öğrenme bağlamında EfficientNetB0, ImageNet gibi büyük veri kümeleri üzerinde önceden eğitilmiş ağırlıklarla kullanılmakta ve farklı görevlerde hızlı adaptasyon sağlamaktadır. Özellikle veri hacminin kısıtlı olduğu durumlarda, modelin erken blokları sabitlenerek (freeze) yalnızca sınıflandırıcı katmanlar yeniden eğitilebilmekte veya tüm ağ düşük öğrenme oranıyla güncellenebilmektedir. Bu parametrik verimlilik, EfficientNetB0'nin mobil cihazlardan bulut tabanlı sistemlere kadar geniş platformlarda uygulanmasını mümkün kılmaktadır [83], [99].

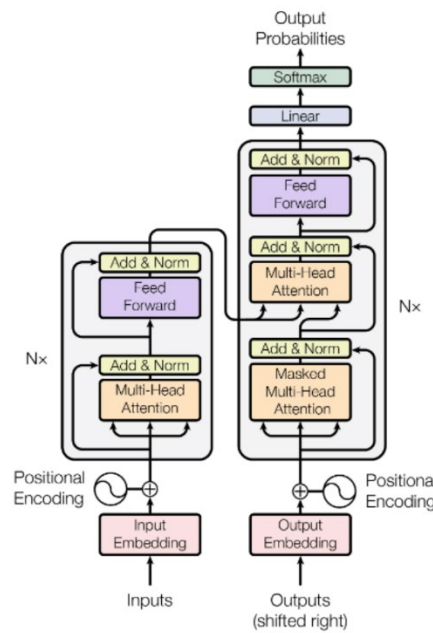
2.4 Transformer Tabanlı Modeller

Derin öğrenme teknikleri, son yıllarda yapay zekâ alanında çığır açan gelişmelere öncülük etmiş ve özellikle görsel veri analizi ile sınıflandırma konularında klasik yöntemlerin önüne geçmiştir [100]. Bu gelişmelerin merkezinde yer alan konvolüsyonel sinir ağları (CNN), yerel özelliklerin etkili çıkarımı ve hiyerarşik temsiller oluşturmadaki başarılarıyla bilgisayarla görünün temel yapı taşlarından biri haline gelmiştir [64]. Ancak CNN mimarileri, sınırlı alansal algılama kapasitesi ve uzun menzilli bağımlılıkları modelleme güçlüğü gibi yapısal kısıtlamalar taşımaktadır. Derinleşen ağlarda, uzak pikseller arasındaki ilişkilerin öğrenilmesi zorlaşmakta ve bu durum uzun menzilli bağımlılıkların modellenmesini engellemektedir [101]. Sabit boyuttaki filtreler ve lokal receptive field yapıları nedeniyle, CNN'ler tüm görüntüdeki küresel bağlamı doğrudan işleyememekte bu ise özellikle karmaşık yapısal örüntüler içeren görsellerde, örneğin tarımsal görüntülerde performansın sınırlandırılmasına neden olmaktadır [102].

Bu sınırlamaların aşılması amacıyla geliştirilen Transformer mimarileri, ilk olarak doğal dil işleme alanında dikkat mekanizmaları üzerinden uzun menzilli bağımlılıkları modellemedeki başarısıyla öne çıkmıştır [103]. Transformer'lar, konvolüsyonel filtreler yerine tamamen dikkat (attention) tabanlı işlemleri esas alarak, giriş verisindeki her bir birim arasındaki ilişkiyi küresel bağlamda değerlendirebilme yeteneği kazandırmıştır. Bu özellik, özellikle uzamsal örüntülerin bütünsel biçimde ele alınmasını gerektiren görsel veri analizlerinde büyük bir avantaj sunmaktadır [104]. Görüntüler üzerinde doğrudan uygulanabilmesi için, bu mimarilerin giriş katmanları yeniden yapılandırılmış; piksellerden oluşan görüntüler, sabit boyutlu parçalara (patch) ayrılarak birer sıraya dönüştürülmüş ve ardından lineer gömülere çevrilerek modele beslenmiştir [102]. Bu dönüşüm sayesinde, daha önce yalnızca sıralı verilere uygulanabilen Transformer mimarileri,

Vision Transformer (ViT) gibi yapılarla birlikte görsel sınıflandırma görevlerinde de yüksek başarı sağlamaya başlamıştır.

Şekil 2.14'te, Vaswani ve arkadaşları (2017) tarafından önerilen orijinal Transformer mimarisi yer almaktadır. Bu yapı, encoder ve decoder bloklarından oluşan çift yönlü bir sistem sunmakta; her iki bölümde de çoklu başlı dikkat (multi-head attention), katman normu (layer normalization) ve ileri beslemeli ağ (feed-forward network) bileşenleri yer almaktadır. Encoder, giriş dizisinin tüm pozisyonlarını dikkate alarak bağlamsal temsiller üretirken; decoder, hem önceki çıktıları hem de encoder'dan gelen temsilleri kullanarak yeni çıktılar üretmektedir.



Şekil 2- 14: Transformer mimarisi

Bu mimari, görsel veri gibi sıralı olmayan girdilere doğrudan uygulanmasa da ViT gibi modellerde yalnızca encoder kısmı kullanılarak başarıyla uyarlanmıştır. Görsel Transformer mimarilerinin neredeyse tamamı bu temel yapının encoder bileşenini merkeze alarak geliştirilmiştir [57, 109].

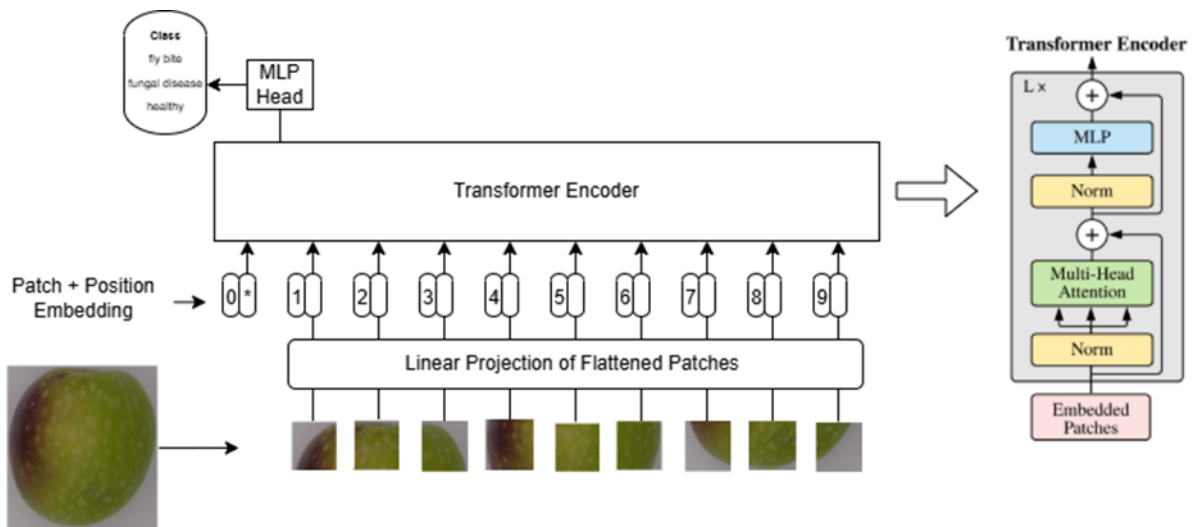
Bu bağlamda çalışma kapsamında görsel sınıflandırma alanında yüksek performanslarıyla öne çıkan BEiT, DeiT, LeViT, MaxViT, MobileViT ve ViT olmak üzere altı farklı transformer tabanlı derin öğrenme mimarileri deneysel olarak değerlendirilmiştir. Söz konusu modeller, ilgili resmi kütüphanelerden temin edilen önceden eğitilmiş ağırlıklarla başlatılmış ve çalışma gereksinimlerine uygun olarak üç sınıflı etiketleme (sinek ısırığı, mantar hastalığı, sağlıklı) görevine uyarlanmıştır.

2.4.1 Vision Transformer (ViT)

Vision Transformer (ViT), görüntü sınıflandırma görevlerinde evrimsel sinir ağlarına (CNN) alternatif olarak saf Transformer mimarisinin uygulanabileceğini ortaya koyan bir modeldir. ViT, giriş görüntüsünü sabit boyutlu parçalara (örneğin 16×16 piksellik yamalar) böler, her bir parçayı düzleştirir ve doğrusal bir projeksiyon aracılığıyla sabit uzunlukta vektörlere dönüştürür [102]. Bu yama temsilleri, Transformer'ın dizisel yapısına uyumlu hâle getirilmek üzere pozisyonel kodlamalarla zenginleştirilir ve böylece modelin konumsal farkındalığı artırılır [103].

ViT mimarisinin yapısal temelini sadece encoder blokları oluşturur. Bu bloklarda çoklu başlı dikkat (multi-head attention), katman normu ve ileri beslemeli ağlar (MLP) ardışık olarak yer alır [102]. Bu yapı sayesinde ViT, her yamanın tüm diğer yama temsilleriyle bağlamsal ilişkilerini eşzamanlı olarak işler ve böylece küresel görsel örüntüleri etkin biçimde öğrenir [105]. Görüntü sınıflandırma alanında, bu küresel dikkat mekanizması, CNN'lerin lokal filtrelerine kıyasla daha geniş kapsama alanı sağlayarak üstün performans sunar [106].

Şekil 2-15'te yer alan yapısal akışta görüldüğü gibi, süreç şu adımlarla ilerler: Öncelikle her yama lineer bir katmandan geçirilerek temsile dönüştürülür ve pozisyonel kodlamalar eklenerek modelin uzamsal farkındalığı korunur [103]. Şekilde "0*" ile işaretlenen ilk giriş, sınıflandırma için özel olarak tanımlanmış [CLS] token'ıdır. Bu token'ın özel bir rolü vardır: encoder sürecinde diğer tüm yama temsilleriyle etkileşimde bulunarak görüntünün küresel özetini temsil eder [107].



Şekil 2- 15: Vision Transformer mimarisinin yapısal akışı

Patch embedding ve pozisyonel kodlama işlemlerinin tamamlanmasının ardından, tüm temsiller paralel şekilde transformer encoder bloklarına gönderilir. Her encoder bloğu sırasıyla katman normu, çoklu başlı self-attention ve ileri beslemeli MLP yapısından oluşur. Bu bloklar L defa tekrarlanır ve modelin derinliği, bu tekrar sayısı ile belirlenir. Her encoder katmanı içinde dikkat mekanizması, her token'ın (yama) diğer tüm token'larla ilişkisini eşzamanlı olarak değerlendirir.

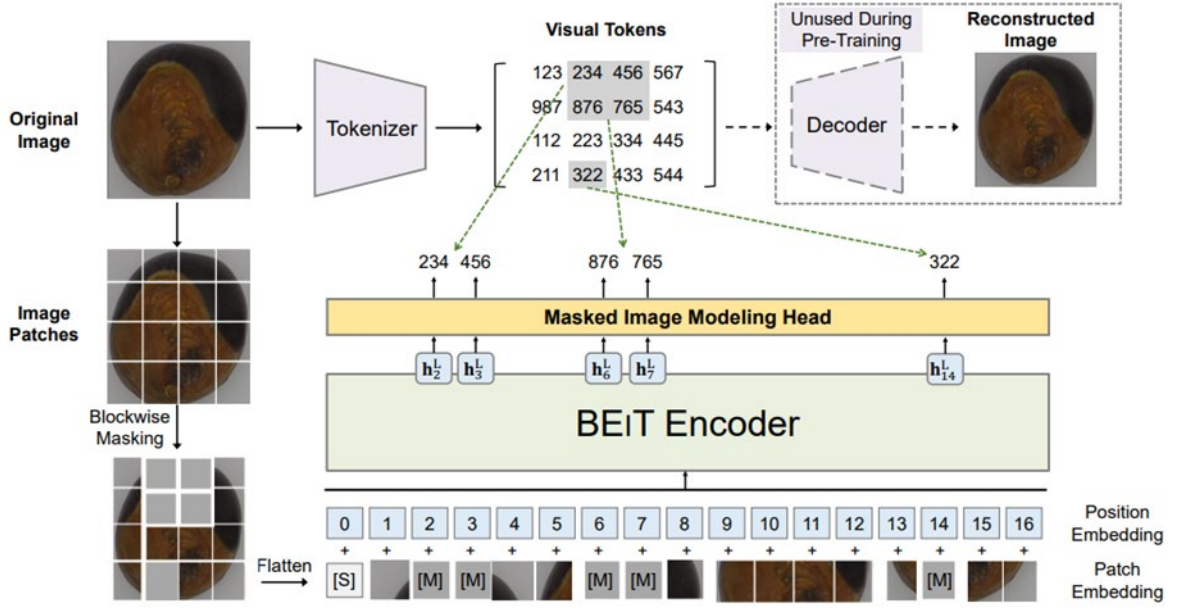
Encoder çıktısı, yalnızca [CLS] token'ına karşılık gelen vektör üzerinden sınıflandırıcı katmana (MLP head) aktarılır. Bu katman, görüntünün ait olduğu sınıfı (ör. “sinek ısırtığı”, “mantar hastalığı”, “sağlıklı”) tahmin etmek üzere eğitilir. Böylece, tüm işlem boyunca tek bir doğrusal sınıflandırma yapılırken, karar küresel bağlamsal ilişkiler temelinde verilir.

2.4.2 BERT Pre-Training of Image Transformers (BEiT)

BEiT modeli, doğal dil işleme alanında büyük başarı elde eden BERT mimarisinden esinlenerek, görsel veriler üzerinde benzer biçimde maskeli modelleme yapılabileceğini gösteren öncü bir çalışmadır [108]. Bu modelin temel hedefi, bazı bölümleri (yamaları) rastgele gizlenmiş bir görüntüden, bu bölgelerin içeriklerini bağlamsal ilişkilerle yeniden tahmin etmektir [112, 113].

BEiT mimarisi, ViT benzeri bir yapıda, giriş görüntüsünü sabit boyutlu yamalara ayırır; her bir yama düzleştirilerek lineer projeksiyonla vektörel temsile dönüştürülür ve pozisyonel kodlamalar eklenerek modele sunulur [108]. Ancak ViT'den farklı olarak, BEiT ilk aşamada doğrudan sınıflandırma görevine değil, maskeli görüntü modellemesine (Masked Image Modeling)[111] odaklanır. Bu bağlamda, bazı yamalar maskeleyme stratejisiyle gizlenir ve modelden bu gizli bölgelerin içeriklerini tahmin etmesi beklenir. Tahmin sürecinde hedef olarak, önceden eğitilmiş bir dVAE (discrete Variational Autoencoder) tarafından oluşturulan görsel token'lar kullanılır[108].

Şekil 2-16'da BEiT modelinin ön-eğitim süreci şematik olarak sunulmaktadır. Giriş görüntüsü yamalara bölünüp her biri gömülendikten sonra (patch embedding), belirli bir oranı rastgele maskeleyme işlemine tabi tutulur.



Şekil 2- 16: BEiT modelinin yapısal akışı

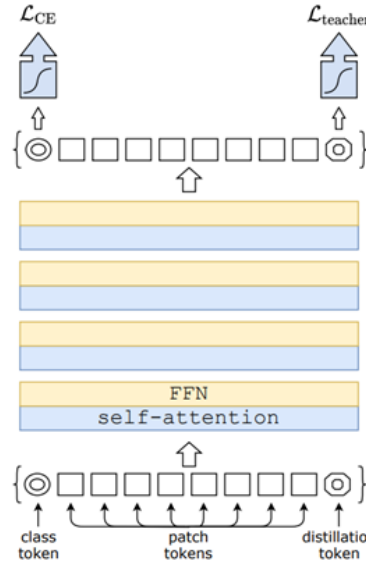
Maskeleme sonrası ortaya çıkan [M] token'ları, ViT tabanlı encoder üzerinden işlenir. Encoder'ın ürettiği ara temsiller, masked image modeling head üzerinden görsel token'lara eşleştirilerek maske altındaki bölgelerin ne olduğuna ilişkin tahminler yapılır. Şekilde görüldüğü üzere, bu işlemde decoder katmanı yer alsa da yalnızca hedef sembollerin referans olarak kullanılması söz konusudur.

2.4.3 Data-efficient Image Transformer (DeiT III)

DeiT III modeli, Vision Transformer (ViT) mimarisi temel alınarak geliştirilmiş ve herhangi bir dışsal öğretici model ya da distillation token kullanılmaksızın, yalnızca denetimli öğrenme yöntemiyle eğitilen yüksek verimliliğe sahip görsel Transformer mimarilerinden biri olarak öne çıkmaktadır [112]. Bu modelde, 384×384 çözünürlükteki giriş görüntüsü, 16×16 piksellik sabit boyutlu yamalara ayrılarak toplam 576 adet patch elde edilir. Her bir yama, sabit uzunlukta bir vektöre gömülme ve pozisyonel kodlamalarla zenginleştirilerek Transformer encoder bloklarına iletilmektedir [102], [112]. DeiT III, encoder katmanlarında klasik ViT bileşenleri olan çoklu başlı dikkat (multi-head attention), katman normu (layer normalization), artık bağlantılar (residual connections) ve ileri beslemeli MLP yapısını korumaktadır [112].

DeiT III'ün eğitim süreci; learnable scale, layer scale, stochastic depth ve exponential moving average (EMA) gibi modern tekniklerle desteklenmiştir. Bu teknikler, modelin

optimizasyon sürecinde istikrarı artırmakta ve özellikle learnable scale mekanizması sayesinde her encoder bloğunun katkısı adaptif biçimde ayarlanarak daha dengeli bir gradyan akışı sağlanmaktadır [113, 116].



Şekil 2- 17: DeiT mimarisinin yapısal şeması

DeiT III, ilk sürümde kullanılan distillation token yaklaşımından farklı olarak, eğitimi yalnızca denetimli öğrenme temelli olarak yürütür. Şekil 2-17’de gösterildiği gibi, klasik DeiT yapısında distillation token, dışsal bir öğretici model aracılığıyla bilgi aktarımı için kullanılmakta ve hem [CLS] token hem de distillation token üzerinden kayıplar hesaplanmaktadır [107]. Ancak DeiT III, bu yapıyı sadeleştirerek yalnızca [CLS] token’ı üzerinden eğitimi sürdürmekte ve böylece hem eğitim süreci basitleşmekte hem de öğreticiye olan dışsal bağımlılık ortadan kaldırılmaktadır [112].

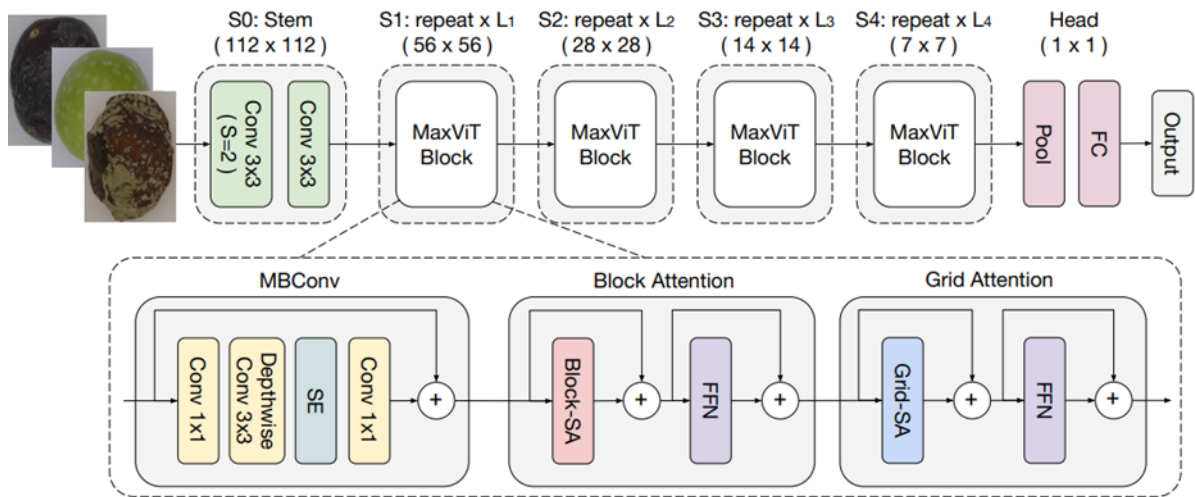
2.4.4 Mobile Vision Transformer (MobileViT v2)

MobileViT v2, mobil cihazlar gibi sınırlı kaynaklara sahip ortamlarda çalışmak üzere tasarlanmış, konvolüsyonel sinir ağlarının yerel özellik çıkarım kapasitesini, Transformer mimarisinin küresel bağlam öğrenme gücüyle birleştiren hibrit bir modeldir [114]. Bu mimaride, her MobileViT bloğu yerel ve küresel temsilleri ardışık olarak işler. İlk olarak, konvolüsyonel katmanlar düşük maliyetli yerel özellik çıkarımı gerçekleştirir ve ardından bu temsiller,

EfficientNet mimarisinden uyarlanmış olup 1×1 konvolüsyon, derinlik ayırık 3×3 konvolüsyon ve Squeeze-and-Excitation modüllerini içermektedir. Bu yapı sayesinde düşük parametrelili ve yüksek verimli yerel temsiller üretilmektedir [83].

Dikkat mekanizmaları iki aşamalı bir yaklaşım izlemektedir. İlk olarak pencere tabanlı dikkat (blockwise window attention), sabit boyutlu bölgelere odaklanarak her pencere içindeki örüntüleri işler. Ardından ızgara dikkat (grid attention) devreye girerek tüm görüntü üzerinde ekstenel taramalarla (dikey ve yatay) küresel bağlamı yakalar [118].

Şekil 2-19’da MaxViT mimarisinin katman bazlı yapısı ve bilgi akışı şematik olarak sunulmaktadır. Giriş görüntüsü öncelikle S0 (stem) evresinde 3×3 konvolüsyon ile 112×112 boyutuna indirgenirken, takip eden S1–S4 blokları boyunca konvolüsyonel ve dikkat mekanizmaları eş zamanlı olarak çalışmaktadır. Her blok kendi çözünürlük seviyesinde birkaç kez tekrarlanarak derinlik artırılmakta ve modelin genel öğrenme kapasitesi güçlendirilmektedir [116].

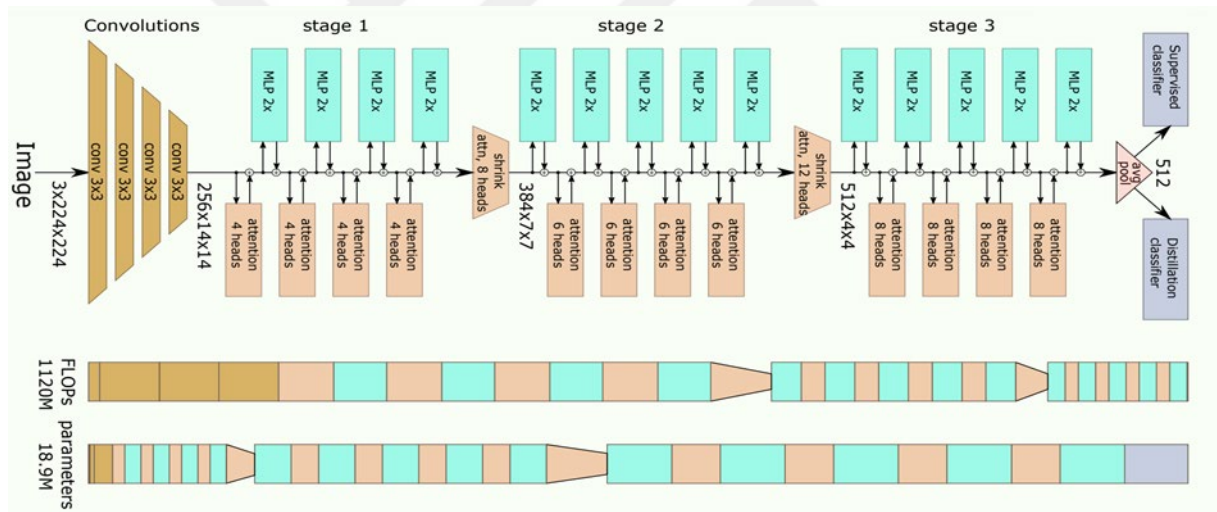


Şekil 2- 19: MaxViT mimarisinin şematik yapısı.

Modelin son katmanında global average pooling ve tam bağlantılı sınıflandırıcı bulunmaktadır. Bu yapı, öğrenilen tüm çok katmanlı temsilleri özetleyerek nihai sınıf tahminini gerçekleştirmektedir. Özellikle sınırlı etiketli veriyle eğitilen sistemlerde, MaxViT mimarisinin sunduğu dikkat çeşitliliği genelleme kabiliyetini önemli ölçüde artırmaktadır [117].

2.4.6 Lightweight & Efficient Vision Transformer (LeViT)

LeViT mimarisi, Vision Transformer (ViT) modellerinin sunduğu yüksek doğruluk seviyesini korurken, mobil cihazlar ve sınırlı kaynaklara sahip sistemlerde çalışabilecek şekilde optimize edilmiş, verimli bir yapıdır. Şekil 2-20’de, LeViT mimarisinin katman yapısı ve işlem akışı şematik olarak sunulmaktadır. Bu üç evreli hiyerarşik yapısında sırasıyla 4, 6 ve 8 başlıklı çoklu dikkat (multi-head attention) katmanları ile SiLU aktivasyonlu MLP blokları dönüşümlü olarak yer almakta ve evre geçişlerinde kullanılan “shrink” modülleri sayesinde uzamsal çözünürlük azaltılırken gömme (embedding) boyutu kademeli olarak artırılmaktadır (256→384→512). Bu tasarım, çok düzeyli bağlamsal temsil öğrenimini etkin biçimde desteklemektedir. Ayrıca, klasik ViT’lerde kullanılan MLP yapıları yerine SiLU tabanlı lineer projeksiyonların tercih edilmesi ve dikkat mekanizmalarının hızlandırılmış biçimde uygulanması, LeViT’nin düşük gecikmeli uygulamalarda yüksek verimlilik sunmasını sağlamaktadır [119].



Şekil 2- 20: LeViT mimarisinin katman yapısını ve evreleme sistemini gösteren şematik yapı

Modelin çıkış aşamasında, 512 boyutlu global temsil vektörü ortalama havuzlama (average pooling) yöntemiyle özetlenmekte ve iki ayrı çıkış birimine yönlendirilmektedir. Bunlardan biri doğrudan sınıflandırma amacıyla kullanılan supervised classifier, diğeri ise öğretici (teacher) modelden gelen sinyalleri değerlendiren distillation classifier birimidir [120].

2.4.7 Transformer Tabanlı Modeller İçin Deneysel Ayarlar/Optimizasyon Parametreleri

Optimizasyon yöntemleri, ağırlık ürettiği çıktı ile gerçek değer arasındaki farkı yani hatayı en aza indirmek amacıyla kullanılan tekniklerdir. Derin öğrenme süreçlerinde, hata fonksiyonunun mutlak minimumuna ulaşılması bu yöntemler sayesinde mümkün olmaktadır. Böylece öğrenme sürecinin daha sağlıklı ve verimli şekilde gerçekleşmesi hedeflenir. Optimizasyon algoritmalarının kullanımında en önemli faktörlerden biri ise öğrenme oranıdır. Bu çalışmada, tüm derin öğrenme modellerinin eğitimi sırasında kullanılan SGD, Adam, AdamW, RMSprop, Adagrad, Adadelat, Adamax, NAdam ve RAdam gibi optimizasyon algoritmalarının parametreleri, PyTorch kütüphanesinin torch.optim modülünde tanımlanan ve resmi dokümantasyonda belirtilen varsayılan değerler esas alınarak belirlenmiştir.

Tüm optimizelerde öğrenme oranı (lr), moment parametresi (momentum), epsilon (eps), ağırlık çürüme oranı (weight_decay) gibi temel hiperparametreler PyTorch'un en güncel sürümünde yer alan default değerleriyle uygulanmıştır. Kodda bu parametreler şeffaflık ve tekrarlanabilirlik amacıyla açıkça belirtilmiştir.

Transformer tabanlı modellerde Adafactor, PyTorch'un kendi optimizasyon kütüphanesinde bulunmamakta; HuggingFace Transformers kütüphanesi (transformers) aracılığıyla eklenmiştir. Adafactor'ın uygulanmasında HuggingFace'in güncel dokümantasyonunda önerdiği parametreler (scale_parameter=True, relative_step=True, warmup_init=False, lr=None) kullanılmıştır.

Böylece bu bölümde sunulan tüm optimizasyonlar hem PyTorch hem de HuggingFace ekosistemlerinde yaygın olarak önerilen, bilimsel çalışmalarda tekrarlanabilirliği yüksek parametre ayarlarıyla karşılaştırmalı olarak kullanılmıştır.

SGD (Stochastic Gradient Descent - Rastgele Gradyan Alçalması), derin öğrenme algoritmalarında optimal parametre değerini bulmak için yinelemeli (iteratif) olarak çalışan bir optimizasyon yöntemidir. Hata değerini azaltmak için parametre değerlerini sık sık günceller. Her yinelemede eğitim verisi için yeni bir tahmin değeri hesaplandığı için yavaş çalışır. Bu özelliğinden dolayı büyük veri kümelerinde kullanımı uygun değildir. En önemli avantajı, verimli ve uygulanması kolay bir yöntem olmasıdır [121].

AdaGrad optimizasyon yöntemi, her adımda farklı bir öğrenme oranı kullanarak bu sorunu aşmayı hedefler. SGD yönteminde sabit bir öğrenme oranı kullanılırken, AdaGrad geçmişteki gradyanların karelerinin toplamına dayalı bir öğrenme oranı hesaplar [122].

RMSProp (Root Mean Squared Propagation), sabit öğrenme oranı problemine çözüm olarak önerilmiştir. AdaGrad'dan, gradyan toplamı yöntemi açısından farklılık gösterir. Geçmiş

gradyan bilgilerini göz ardı ederek, yalnızca son gradyan bilgisi ile işlem yapar. Gradyanlar, üssel olarak ağırlıklandırılmış ortalama alınarak hesaplanır. Zamanla değişen bir öğrenme oranı kullanır ve her parametre için ayrı bir değer uygulanır. Bu yönüyle gradyan tabanlı bir optimizasyon yöntemidir [127, 128].

AdaDelta, AdaGrad'ın öğrenme oranı sorununu aşmak için önerilmiş bir optimizasyon yöntemidir. Çok fazla yineleme durumunda, öğrenme oranının çok küçük olması yavaş yakınsama problemine neden olur. AdaDelta yöntemi kullanıldığında, belirli bir öğrenme oranı seçme zorunluluğu yoktur. Bunun yerine, mevcut ağırlıklarla güncellenmiş ağırlıklar arasındaki farkı gösteren delta değerinin karelerinin momentum toplamları kullanılır. Bu sayede, üssel olarak azalan ortalama değer alınması, yavaşlık sorununu ortadan kaldırır [126, 129].

Adam (Adaptive Moment Estimation - Uyumlu Moment Tahmini) yöntemi esas olarak derin öğrenme çalışmaları için tasarlanmıştır. RMSProp yöntemine momentumu ekleyerek en uygun değeri elde etmeyi amaçlar. Tarihsel gradyanların üssel azalan ortalamasını tutar ve her bir parametre için uyarlanabilir öğrenme oranları ayarlar. En önemli özelliği, farklı parametreler için bireysel uyarlanabilir öğrenme oranlarını bulmasıdır [130, 131, 132].

Nadam (Nesterov-accelerated Adaptive Moment Estimation), Adam ve Nesterov Momentumu'nun birleştirilmesiyle oluşturulmuştur. Gradyanların hareketli ortalamasına dayanan üssel büyüme yöntemi kullanır. Böylece model eğitimi sırasında öğrenme süreci hızlandırılır. Nadam, Adam'a göre daha hızlı yakınsar. Uygulamasının basit ve hesaplamalarının verimli olması nedeniyle sıkça tercih edilir [133, 134].

AdaMax (Adaptive Max Pooling Optimizer), sonsuz norm temelli Adam optimizasyon yönteminin geliştirilmiş bir versiyonudur. En önemli avantajlarından biri, hiperparametre seçimine duyarlılığıdır. Bu sayede daha güvenilir çözümler sunar [131].

Adafactor, özellikle büyük dil modelleri ve sınırlı bellekli cihazlarda verimli eğitim amacıyla geliştirilmiş bir optimizasyon yöntemidir. Bu yöntem, Adam'a benzer şekilde moment tahminleri kullanır ancak ikinci momentin (squared gradients) tümünü değil, yalnızca satır ve sütun ortalamalarını tutarak bellek gereksinimini büyük ölçüde azaltır. Bu özelliği sayesinde büyük parametre matrisleriyle çalışan modellerde önemli bir performans ve hafıza kazancı sağlar. Ayrıca Adafactor, otomatik olarak öğrenme oranı ayarlama yeteneğine sahiptir ve düşük kaynaklı koşullarda yüksek stabilite ile çalışır [132].

RAdam (Rectified Adam), klasik Adam algoritmasının düşük iterasyon sayılarında dengesiz öğrenme oranı üretme sorununu çözmek amacıyla geliştirilmiştir. Bu yöntem, adaptif öğrenme oranını düzeltici bir mekanizma ile dengeleyerek öğrenme sürecinde daha kararlı ve hızlı

yakınsama sağlar. Özellikle küçük veri kümelerinde ya da erken epoch aşamalarında performans dalgalanmasını azaltır [133].

AdamW (Adam with Weight Decay) ise klasik Adam algoritmasında ağırlık çürümesinin (weight decay) yanlış uygulanmasını düzelterek geliştirilmiş bir versiyondur. Adam algoritmasında, ağırlık çürütme öğrenme oranına bağlı olarak yanlış etki yaratabilirken, AdamW bu etkiyi ayrı olarak ele alarak düzenlenileştirmenin daha doğru şekilde uygulanmasını sağlar. Böylece aşırı öğrenmeyi (overfitting) engellemede daha başarılı sonuçlar verir ve özellikle transformer tabanlı modellerde yaygın olarak tercih edilir [134].

2.5 MetaFormer Tabanlı Yapılar

Son yıllarda, görsel sınıflandırma görevlerinde Transformer mimarileri, özellikle dikkat (attention) mekanizmalarının sunduğu yüksek doğruluk oranları sayesinde önemli bir ivme kazanmıştır [102]. Ancak bu modellerin yüksek hesaplama maliyeti ve bellek gereksinimi, özellikle mobil ve gömülü sistemlerde uygulanabilirliğini sınırlamaktadır. Bu sınırlamalara alternatif olarak geliştirilen Metaformer çerçevesi, Transformer mimarilerinin başarısının yalnızca self-attention mekanizmasından değil, daha genel bir mimari iskeletten kaynaklandığını öne sürerek yeni bir yaklaşım sunmuştur [135].

Metaformer, temel olarak üçlü bir işlem dizisi üzerine inşa edilmiştir: (1) normalizasyon katmanı, (2) token mixer modülü ve (3) kanal boyutlu çok katmanlı algılayıcı (Channel MLP). Bu yapıdaki en kritik bileşen olan token mixer, girdideki uzamsal ilişkileri modellemekten sorumludur ve bu noktada self-attention zorunlu değildir. Yu ve arkadaşları [135] token mixer yerine ortalama havuzlama (average pooling), konvolüsyon (convolution) veya hatta parametresiz işlemler gibi alternatiflerin de kullanılabileceğini ve bu yapıların dikkat temelli modellere yakın performans gösterebildiğini ortaya koymuştur. Böylece Metaformer, dikkat mekanizmasını zorunlu olmaktan çıkarak hesaplama verimliliğini ve mimari esnekliği artırmayı hedeflemektedir.

Metaformer çerçevesi, belirli bir modelden ziyade Transformer mimarisinden soyutlanmış evrensel bir yapı iskeleti olarak tanımlanır ve başarının temelini belirli işlemlerden çok yapısal organizasyona atfeder. Bu yaklaşımda ilk adım, giriş verisinin bir gömme (embedding) işlemiyle token dizisine dönüştürülmesidir:

$$X = \text{InputEmbedding}(I) \quad (2.1)$$

Burada $X \in \mathbb{R}^{N \times C}$, sıralı token'lerden oluşan ve uzunluğu N, kanal boyutu C olan bir tensörü ifade eder. Bu gömülen token dizisi, her biri iki alt bloktan oluşan ardışık MetaFormer

bloklarından geçirilir. İlk alt blok, token'lar arası bilgi etkileşimini sağlayan bir karıştırıcı (token mixer) içerir:

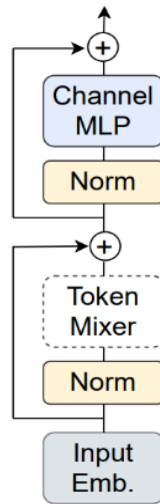
$$X' = X + \text{TokenMixer}(\text{Norm}_1(X)) \quad (2.2)$$

İkinci alt blokta ise, her bir token üzerinde kanal düzeyinde dönüşüm gerçekleştiren ve öğrenilebilir ağırlıklara sahip bir MLP (çok katmanlı algılayıcı) yer alır:

$$X'' = X' + \sigma(\text{Norm}_2(X')W_1)W_2 \quad (2.3)$$

Bu yapıda, $\text{Norm}_1(\cdot)$ ve $\text{Norm}_2(\cdot)$ genellikle LayerNorm olarak kullanılan normalizasyon işlevleridir; $\sigma(\cdot)$ bir aktivasyon fonksiyonudur (örneğin GELU); $W_1 \in \mathbb{R}^{C \times rC}$ ve $W_2 \in \mathbb{R}^{rC \times C}$ ise MLP'nin öğrenilebilir ağırlıklarıdır. Token mixer, token'lar arasında bağlamsal bilgi aktarımı sağlamakla yükümlü modüler bir bileşendir ve self-attention, uzamsal MLP, havuzlama (pooling), rastgele projeksiyon (random mixing) ya da yalnızca kimlik eşlemesi (identity mapping) gibi birçok biçimde tanımlanabilir [124, 125].

MetaFormer'ın bu genel mimari iskeleti Şekil 2-21'de sunulmaktadır. Girdi, ilk olarak bir embedding katmanından (Input Embedding) geçirilir. Ardından katman normu ve bir token mixer uygulanarak özellikler karıştırılır. Bu işlemin çıktısı, kalıntı bağlantı yoluyla girdiye eklenerek ikinci işlem birimine aktarılır. Bu işlem bloğu da yeniden katman normu ve bir kanal boyutlu MLP katmanından oluşur. Bu ardışık bloklar, MetaFormer'ın evrensel çekirdek yapısını oluşturur.



Şekil 2- 21: MetaFormer yapısının genel mimari iskeleti

Bu mimari çerçeve, seçilecek token mixer türüne bağlı olarak farklı alt yapıların tasarlanmasına olanak tanır. Örneğin, PoolFormer modelinde token mixer olarak yalnızca ortalama havuzlama işlemi kullanılırken; ConvFormer yapısında konvolüsyon temelli token karıştırıcılar tercih edilmektedir. CAFormer gibi hibrit modellerde ise hem konvolüsyon hem de attention tabanlı bileşenler bir arada yer almaktadır [136].

Yeni nesil token mixer tasarımlarının, örneğin istatistiksel filtreler [137], difüzyon skorlayıcılar [138] ve topolojik operatörler [139] gibi özgün yapılarla MetaFormer iskeleti üzerinde uygulanabilir oluşu, bu yapının gelecekteki gelişim potansiyelini de göstermektedir.

Bu yönüyle MetaFormer, yalnızca bir mimari değil, bir üst mimari şablon (meta-architecture) olarak konumlanmakta ve farklı token mixer bileşenleriyle modüler olarak genişletilebilen bir çerçeve sunmaktadır.

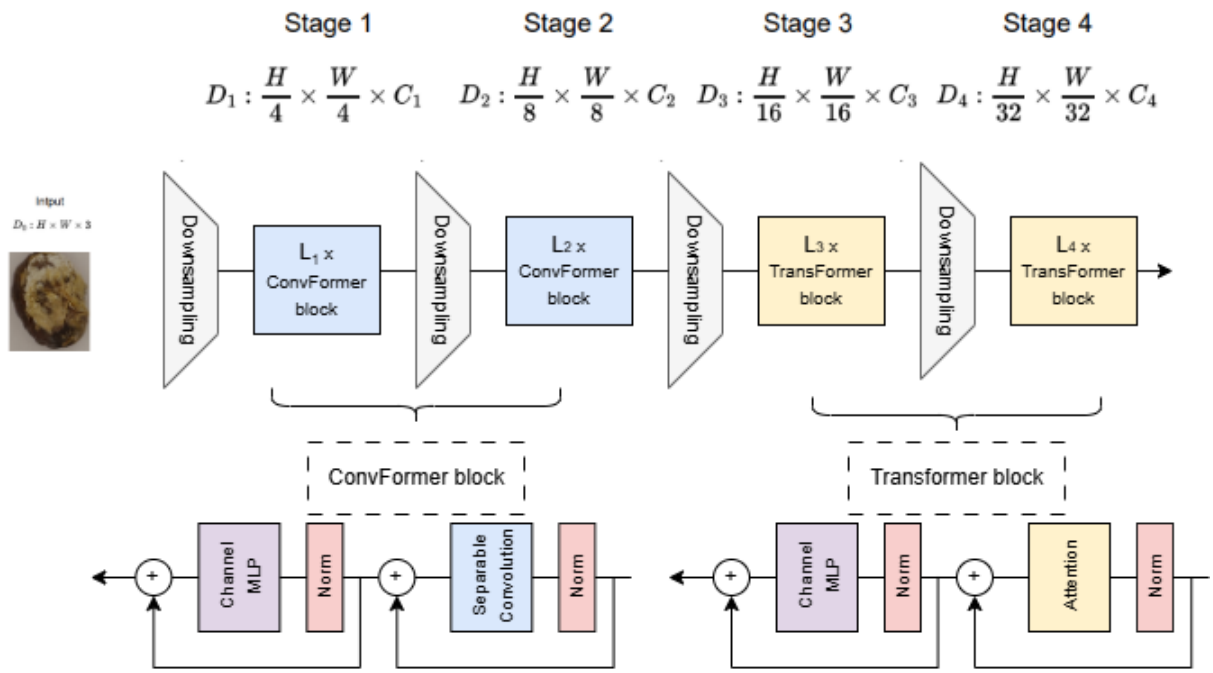
2.5.1 CAFormer

CAFormer, MetaFormer mimarisinin hem konvolüsyonel hem de dikkat (attention) tabanlı karıştırıcıları birlikte kullanan hibrit bir versiyonudur. Bu yapı, son yıllarda konvolüsyon ve dikkat mekanizmalarının birbirini tamamlayıcı özellikleri nedeniyle birlikte kullanılmasının yaygınlaştığı bir literatür zeminine dayanmaktadır [129, 130]. Yu ve arkadaşları (2024), MetaFormer iskeletine dayalı olarak geliştirdikleri CAFormer modelinde, her blokta konvolüsyon ve attention karıştırıcılarını ardışık biçimde entegre ederek hem yerel hem de küresel düzeyde bilgi çıkarımı sağlamıştır.

Model, dört aşamalı bir hiyerarşik yapıya sahiptir ve her aşamada sıralı biçimde yerleştirilen convolution + attention token mixer blokları kullanılmıştır. Özellikle ilk iki aşamada konvolüsyonel işlem baskınken, daha üst düzey katmanlarda self-attention bileşenleri öne çıkmaktadır. Bu yapı, bilgi temsili zenginliğini artırırken aynı zamanda hesaplama verimliliğini korumayı amaçlamaktadır. Bu yapılandırma, Transformer tabanlı modellerde önerilen "early convolutional embedding" stratejileriyle de uyumludur [120].

CAFormer mimarisi, ImageNet-1K gibi büyük ölçekli görüntü veri kümelerinde test edilmiş; CAFormer-S18 ve CAFormer-S36 gibi varyantları hem doğruluk hem de işlem verimliliği bakımından Vision Transformer ve ResNet gibi modellerle rekabetçi sonuçlar elde etmiştir. Özellikle CAFormer-S36, düşük FLOPs ve parametre sayısı ile dikkat çekici sınıflandırma başarımı sağlamıştır. Bu yönüyle gömülü sistemler ve kaynak kısıtlı uygulamalar için umut vadeden bir model olarak değerlendirilmektedir [136].

Şekil 2.22’de sunulan mimari şema, MetaFormer manifestosunda tanımlanan “norm → token mixer → norm → channel MLP” blok yapısına bağlı kalınarak tasarlanmıştır. Burada her bir token mixer bileşeni, sırasıyla konvolüsyonel ve dikkat tabanlı işlemlerden oluşmaktadır. İlk iki aşamada yer alan “ConvFormer block” yapıları, ayrılabilir konvolüsyonlara (depthwise separable convolutions) dayanmakta olup, yerel uzamsal örüntülerin çıkarımında etkilidir [74, 98]. İzleyen iki aşamada ise vanilla self-attention uygulanarak uzun menzilli bağlamsal ilişki modellenmesi gerçekleştirilir [103].



Şekil 2- 22: CAFormer mimarisinin dört aşamalı genel şeması

Her aşamadan önce yerleştirilen downsampling modülleri, giriş çözünürlüğünü azaltırken kanal genişliğini artırır. Bu sayede hem işlemsel yoğunluk azaltılır hem de soyutlama düzeyi kademeli olarak yükseltilir. Bloklar içerisinde kullanılan residual bağlantılar, ağırların derinleşmesine rağmen gradyan akışını koruyarak öğrenme sürecini stabilize eder [68].

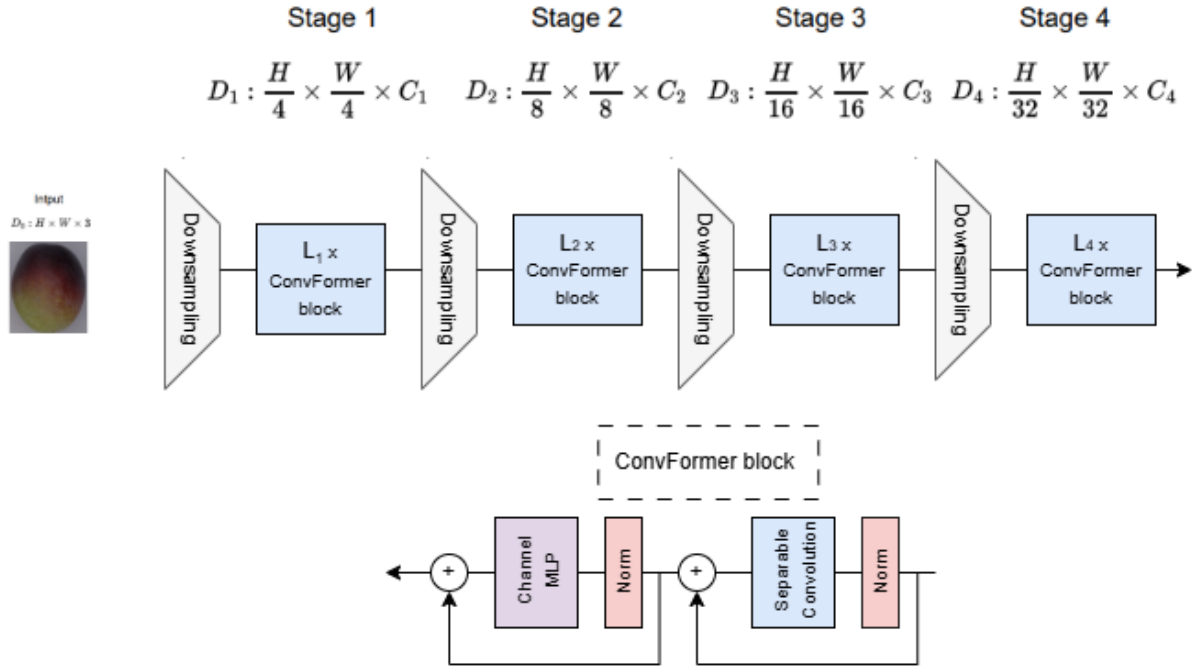
CAFormer, MetaFormer mimarisinin modülerliğini korurken, farklı bilgi işleme paradigmasını tek çatı altında birleştiren dikkat-konvolüsyon hibrit modellerin potansiyelini ortaya koyar. Bu yönüyle, yalnızca attention tabanlı değil, “attention-optional” modellerin de yüksek başarı sağlayabileceğini göstermektedir [136].

2.5.2 ConvFormer

ConvFormer, MetaFormer mimarisinin dikkat (attention) mekanizmasını tamamen dışlayarak yalnızca konvolüsyon tabanlı bir token mixer kullanan versiyonudur. Yu ve arkadaşları (2024), MetaFormer iskeletinin başarısının temelinde yalnızca self-attention değil, işlem sıralamasının yapısal organizasyonu olduğunu savunarak, bu mimaride token mixer bileşeni yerine konvolüsyonel işlemlerin etkili biçimde kullanılabileceğini göstermiştir. ConvFormer bu bağlamda, MetaFormer manifestosunun “attention-free” yaklaşıma dayalı temel uygulamalarından biri olarak değerlendirilir [136].

Modelde kullanılan token mixer bileşeni, depthwise separable convolution ve group convolution gibi hafif konvolüsyonel yapılarla tasarlanmıştır. Bu bileşenler, hem yerel düzeyde bilgi çıkarımı sağlar hem de parametre sayısını ve işlem yükünü önemli ölçüde azaltır. Bu nedenle ConvFormer, mobil cihazlar, gömülü sistemler ve uçtan uca hesaplama gibi kaynak kısıtlı uygulamalar için güçlü bir alternatif olarak önerilmektedir [74, 98].

ConvFormer mimarisi, klasik MetaFormer blok dizilimini korur: LayerNorm → Token Mixer (Conv) → LayerNorm → Channel MLP. Bu bloklar, residual bağlantılar ile desteklenerek gradyan akışını güçlendirir ve derin yapılarla eğitim kararlılığını artırır [68]. Dört aşamalı hiyerarşik yapısı Şekil 2-23’te gösterildiği üzere, her aşamada downsampling işlemleri uygulanarak uzamsal çözünürlük azaltılırken kanal genişliği artırılır. Bu yapı, modelin soyutlama düzeyini kademeli olarak yükseltir ve klasik konvolüsyonel ağlardaki "özellik hiyerarşisi" anlayışına paralel bir şekilde işler [63].



Şekil 2- 23: ConvFormer mimarisinin dört aşamalı genel yapısı

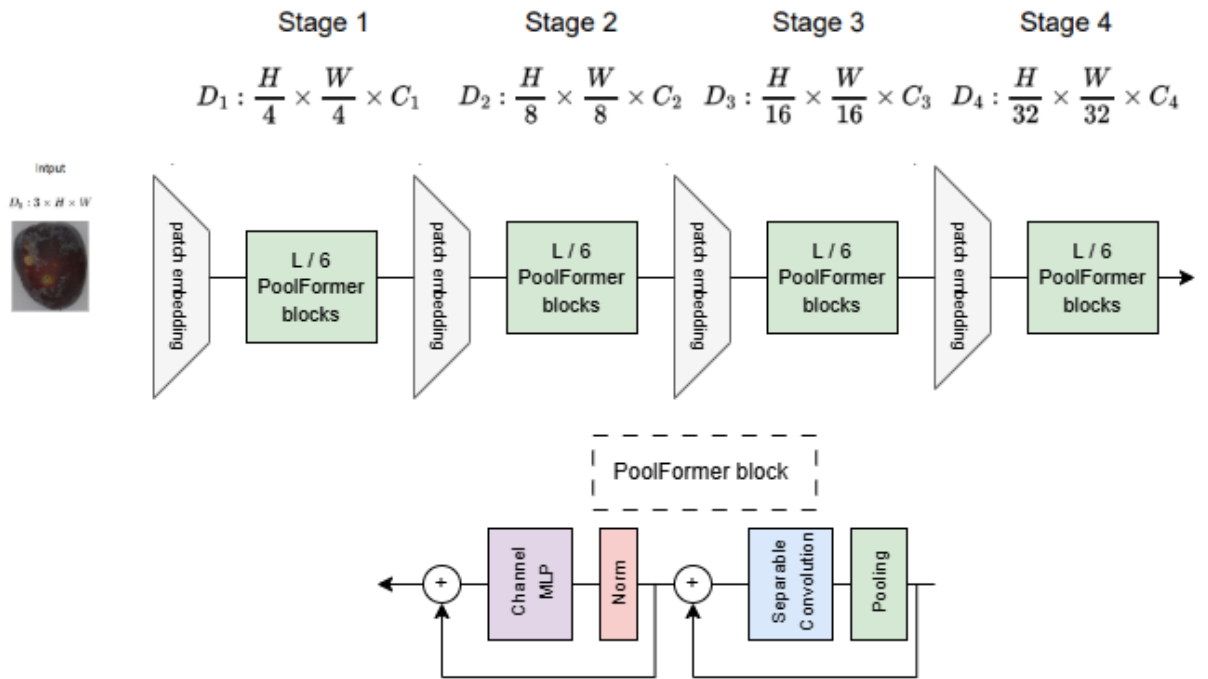
Araştırmalar, dikkat mekanizması olmaksızın konvolüsyon bazlı token mixer'ların yalnızca hesaplama verimliliği değil, aynı zamanda lokal yapısal örüntülerin daha etkili temsil edilmesi açısından da avantaj sunduğunu göstermektedir [142].

2.5.3 PoolFormer

PoolFormer, MetaFormer mimarisi temel alınarak geliştirilmiş, tamamen parametresiz ve attention-free bir modeldir. Yu ve arkadaşları (2022), self-attention veya konvolüsyon gibi parametrelili işlemler yerine yalnızca basit bir ortalama havuzlama işlemi ile token etkileşimini gerçekleştiren $\text{Token Pooling}(x) = \text{AvgPool}(x) - x$ yapısını önererek, hesaplama maliyeti düşük ancak etkili bir mimari ortaya koymuşlardır. Klasik MetaFormer blok dizilimi olan Norm $\rightarrow$ Token Mixer (Pooling) $\rightarrow$ Norm $\rightarrow$ Channel MLP sırası bu modelde de korunmaktadır. Her blok, parametresiz pooling işlemini residual bağlantılarla birleştirerek çalışmaktadır. Bu yapı, özellikle mobil cihazlar, gömülü sistemler ve düşük kaynaklı uygulamalar için büyük avantaj sağlamakta aynı zamanda öğrenilebilir parametre içermemesi sayesinde modelin hesaplama süresi ve bellek gereksinimi ciddi şekilde azalmaktadır. Yu ve arkadaşları, ImageNet-1K üzerinde gerçekleştirdikleri deneylerde PoolFormer-S36 ve M36 gibi varyantların %81'in üzerinde doğruluk elde ettiğini rapor etmiş ve bu sonuç, dikkat mekanizması içermeyen bir mimarinin dahi

yüksek başarı gösterebileceğini ortaya koymuştur. Özellikle lokal dokulara dayalı örüntülerin ayırt edici olduğu tarımsal görüntü analizleri gibi görevlerde, PoolFormer gibi sade ve verimli yapılar güçlü bir alternatif olarak değerlendirilmektedir[135].

Şekil 2-24’te sunulan mimari şema, dört ardışık aşamadan oluşan PoolFormer yapısını göstermektedir. Her aşama, sabit sayıda blok içerir ve aşamalar arasında patch embedding ve downsampling işlemleri uygulanarak uzamsal çözünürlük azaltılır, kanal sayısı artırılır. Modelin tüm aşamalarında yalnızca havuzlama tabanlı token mixer kullanılmakta, bu yönüyle PoolFormer, Metaformer manifestosunun “başarı dikkat değil, yapıdan gelir” ilkesini doğrudan temsil etmektedir.



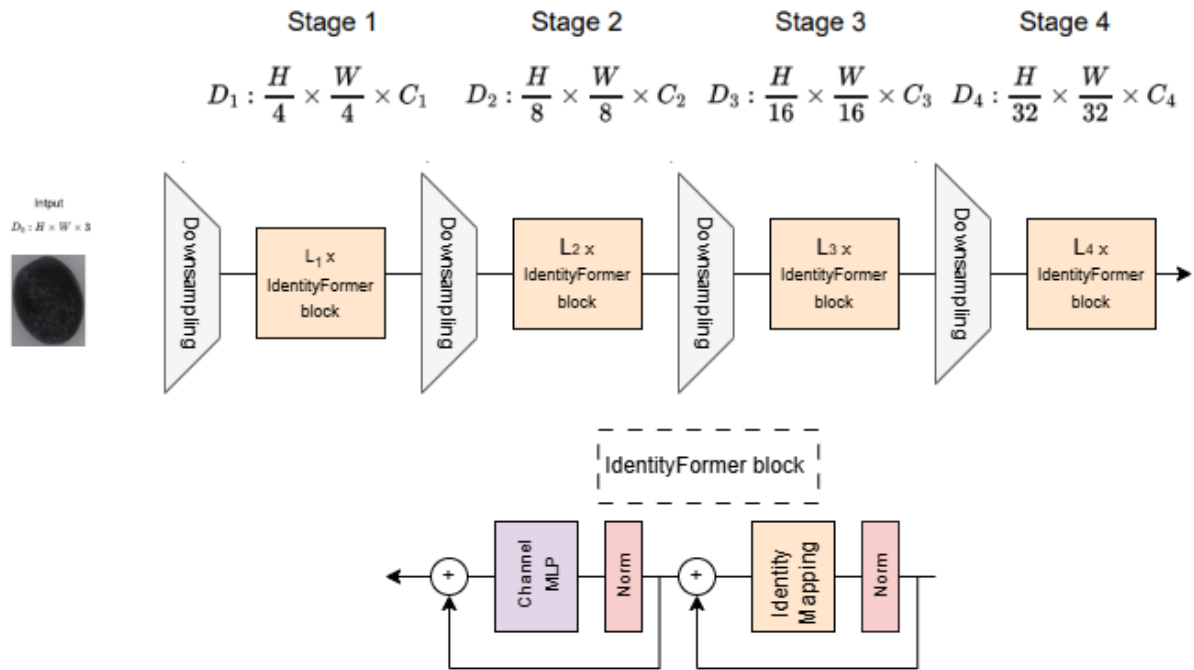
Şekil 2- 24: PoolFormer mimarisinin dört aşamalı genel yapısı

2.5.4 IdentityFormer

IdentityFormer, MetaFormer mimarisine dayalı olarak geliştirilen ve token mixer bileşenini tamamen pasif bırakarak çalışan en sade versiyondur. Yu ve arkadaşları (2022), bu yapıda self-attention, konvolüsyon veya havuzlama gibi işlemler yerine doğrudan kimlik fonksiyonu ($x \rightarrow x$) kullanan bir mimari önererek, yapının başarısının dikkat mekanizmasına değil, norm $\rightarrow$ mixer $\rightarrow$ norm $\rightarrow$ MLP işlem sırasına bağlı olduğunu deneysel olarak ortaya koymuşlardır. Modelde token mixer bileşeninde hiçbir uzamsal işlem yapılmaz; giriş doğrudan

çıktıya aktarılır. Aşamalar arasında yer alan downsampling katmanları (çoğunlukla patch embedding) ile uzamsal boyut küçültülürken kanal genişliği artırılır. Model, yalnızca kanal düzeyinde bilgi etkileşimi sağlayan MLP blokları üzerinden işlem yapmakta, öznetelik aktarımını yatay (uzamsal) değil, dikey (kanal) ekseninde gerçekleştirmektedir. Bu yönüyle IdentityFormer, kaynak kısıtlı ortamlarda çalışabilecek parametresiz ve düşük maliyetli sınıflandırma çözümleri arasında dikkat çeken bir yapı sunmaktadır [136].

Şekil 2-25’te, dört aşamalı IdentityFormer mimarisi şematik olarak sunulmakta olup, her blokta token mixer olarak kimlik dönüşümünün kullanıldığı sade yapısı vurgulanmaktadır.

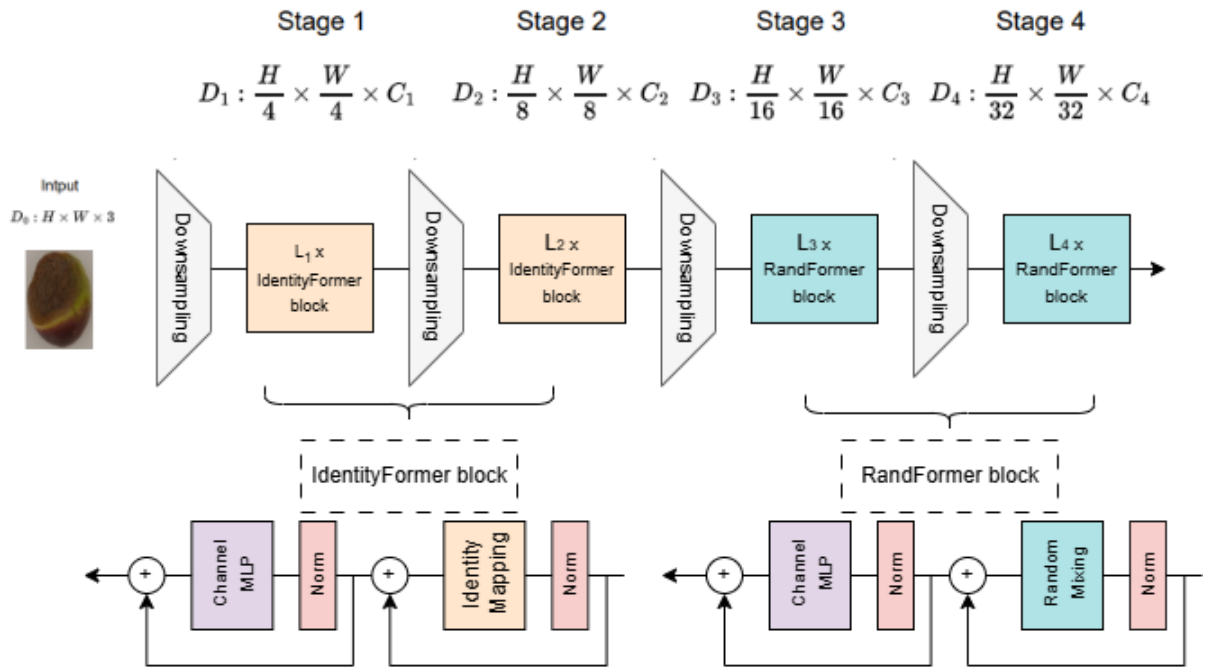


Şekil 2- 25: IdentityFormer mimarisinin dört aşamalı genel yapısı

2.5.5 RandFormer

RandFormer, MetaFormer mimarisine dayalı olarak geliştirilen ve token mixer bileşeni olarak tamamen sabit bir rastgele etkileşim matrisi kullanan deneysel bir modeldir. Yu ve arkadaşları (2022), "Random Mixer" adıyla sundukları bu yapıda, token'lar arası bilgi alışverişi öğrenilebilir bir yapı yerine, sabit ve önceden örneklenmiş rastgele bağlantı örüntüleriyle gerçekleştirilmiştir. Mimari olarak klasik MetaFormer işlem sırası olan LayerNorm → Token Mixer → LayerNorm → Channel MLP düzeni korunmuş, ancak token mixer olarak sabit rastgele

bağlantılar uygulanmıştır. Bu düzenleme, eğitim sürecinde güncellenmeyen, dolayısıyla parametre sayısını artırmayan bir yapı sağlar. Şekil 2-26’da RandFormer mimarisinin dört aşamalı yapısı şematik olarak sunulmakta olup ilk iki aşamada identity mapping kullanan IdentityFormer blokları, son iki aşamada ise sabit rastgele bağlantılara sahip RandFormer blokları yer almaktadır. Bu yapılandırma hem düşük seviyeli temsillerin sade biçimde çıkarılmasını hem de üst düzey soyutlama katmanlarında çeşitlilik sağlayan bilgi aktarımını hedeflemektedir. Özellikle rastgele etkileşimlerin aşırı uyumu önleyici etkisi sayesinde, tarımsal görüntüler gibi örüntü çeşitliliğine duyarlı sınıflandırma görevlerinde etkili olabileceği değerlendirilmektedir [136].



Şekil 2- 26: RandFormer mimarisinin dört aşamalı genel yapısı

2.6 MetaFormer Tabanlı Özgün Blok Kombinasyonları

Bu bölümde, Metaformer mimarisi sabit iskeleti korunarak farklı token mixer bloklarının sıra ve tür bakımından yeniden düzenlenmesiyle oluşturulan özgün hibrit yapılar sunulmaktadır. Bu modellerde kullanılan temel bloklar; Identity Mapping, Random Mixing, Separable Convolution ve Self-Attention olarak belirlenmiş ve her biri LayerNorm → Token Mixer → LayerNorm → Channel MLP düzenine sadık kalarak yerleştirilmiştir [136]. Amaç, yalnızca belirli bir token mixer türüne bağlı mimari yapıların ötesine geçerek, her bir bloğun öğrenme üzerindeki

katkısını ayrı ayrı test edebilecek şekilde modüler, esnek ve görev uyarlanabilir bir çerçeve geliştirmektedir.

Geliştirilen her blok tipi, belirli bilişsel ve yapısal roller esas alınarak tasarlanmıştır. ConvFormer bloklarında, yerel örüntü çıkarımı için ayrılabilir konvolüsyon (separable convolution) ve kanal düzeyli MLP bir arada kullanılmıştır. Bu yapı, girişin önce normalleştirilip ardından Channel MLP ile dikey ekseninde işlenmesini, ardından da SeparableConv ile yatay (uzamsal) örüntülerin çıkarılmasını sağlar. RandFormer bloklarında ise token mixer olarak kullanılan bileşen, kanalların sabit rastgele permütasyonu yoluyla bilgi çeşitliliği ve regularization sağlamayı amaçlamaktadır. IdentityFormer blokları, yalnızca norm $\rightarrow$ MLP $\rightarrow$ norm işlemleriyle çalışır ve token mixer işlevi $x \rightarrow x$ biçiminde, doğrudan bilgi aktarımı yapar. Böylece bilgi korunumunu sağlayan düşük müdahaleli geçiş blokları elde edilmiştir. Transformer blokları ise klasik self-attention ile küresel ilişkilendirme sağlar. Her blok yapısı hem teorik işlev hem de uygulamada hesaplama maliyeti, parametre sayısı ve etkileşim tipi göz önünde bulundurularak modellenmiştir.

Bloklar, MetaFormer manifestosunun önerdiği norm $\rightarrow$ token mixer $\rightarrow$ norm $\rightarrow$ channel MLP düzenine [136] Transformer mimarilerindeki self-attention kullanımının bağlamsal ilişki modelleme avantajlarına [103] ve bilgi kaybını önleyen kimlik aktarımı yaklaşımlarına [75, 125] dayanan genel mimari ilkeler doğrultusunda tasarlanmıştır. ChannelMLP, SeparableConv, RandomMixOp gibi bileşenlerin sıralanışı ve iç işleyişleri bu tez kapsamında görev adaptasyonu ve düşük parametrik yapı hedefleri doğrultusunda optimize edilmiştir. Bu yönüyle geliştirilen ConvFormer ve RandFormer blokları, MetaFormer manifestosunun önerdiği işlem dizisine uygun özgün yorumlar sunmaktadır.

Kullanılan bu blokların teorik dayanaklarını ve mimari işlevlerini özetleyen yapısal ilkeler, Tablo 2-1’de detaylı şekilde sunulmuştur.

Tablo 2- 1: Blok Türlerinin Bilişsel Roller ve Literatür Dayanakları

Mimari İlke	Açıklama	Kaynak
Modüler yapı (norm $\rightarrow$ mixer $\rightarrow$ norm $\rightarrow$ MLP)	MetaFormer manifestosu ile tanımlanan işlem dizisi	[135]
Separable convolution kullanımının etkinliği	Yerel örüntü çıkarımı ve verimlilik	[91]
Channel MLP yapısı	Kanal düzeyli bilgi işleme	[143]

Random Mixing yaklaşımlarının parametrik olmayan yapı olarak kullanımı	Rastlantısal permütasyonlar ile regularization etkisi	[136]
Blok düzeyinde açıklanabilirlik ve bilgi akışı	LayerNorm ve residual bağlantı kullanımı	[68, 102]
Self-Attention ile küresel bağlam modelleme	Uzun menzilli token bağı kurarak bağlamsal öğrenme	[103]
Identity Mapping ($x \rightarrow x$)	Müdahalesiz bilgi aktarımı ve geçişte denge	[136]

Özgün hibrit yapılar, sabit bir Metaformer omurgası üzerinde dört ardışık aşama halinde tasarlanmıştır. Her aşamada farklı bir token mixer bloğu uygulanmış ve *downsampling* işlemleri ile modelin derinliği arttırılmıştır. Bloklar arasında uygulanan sıra ve tip kombinasyonları sistematik olarak değiştirilerek toplam altı farklı hibrit model varyantı oluşturulmuştur.

Önerilen yapılar, literatürde yaygın olarak kullanılan sabit blok dizilimli modellerden farklı olarak görev-adaptif potansiyele sahiptir. Sabit karıştırıcı tiplerinin sınırlılıkları yerine, öğrenme görevine göre değiştirilebilir blok sıralamaları sayesinde hem açıklanabilirlik artmakta hem de blok düzeyinde mimari optimizasyon mümkün hale gelmektedir.

Geliştirilen özgün hibrit modellerin aşamalı blok dizilimlerine ilişkin yapılandırmalar, Şekil 2-27’de şematik olarak sunulmuştur. Her bir modelin blok sıralaması, bilgi işleme sürecinde farklı stratejileri vurgulamak üzere planlanmıştır.

Model 1, ilk aşamada düşük karmaşıklığa sahip bir IdentityFormer bloğu ile başlamakta, ardından rastlantısallık getiren RandFormer ve yerel örüntü çıkarımını sağlayan ConvFormer bloklarını kullanmakta, son aşamada ise küresel bağlamı modelleyen Transformer bloğu ile tamamlanmaktadır. Bu yapı, basit başlangıç ve artan kompleksite yaklaşımını yansıtmaktadır.

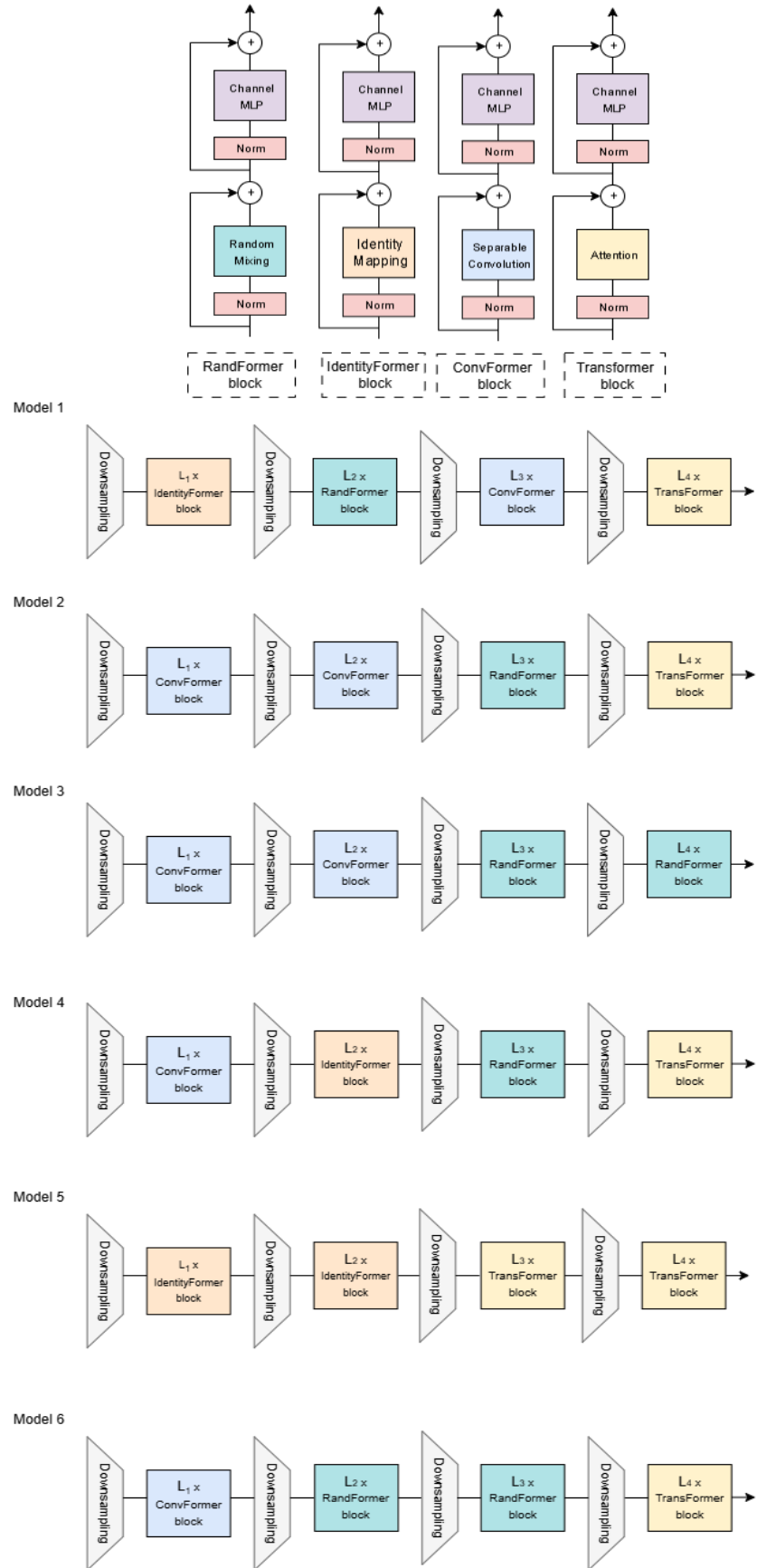
Model 2, ilk iki aşamada ConvFormer bloklarını kullanarak yoğun bir yerel özellik çıkarımı gerçekleştirmekte, üçüncü aşamada rastgele karışım ile çeşitlilik katmakta ve son olarak Transformer bloğu ile global dikkat öğrenimini sağlamaktadır. Bu model, aşamalı olarak lokalden küresele geçişi hedeflemektedir.

Model 3, yine yerel desenleri öğrenmeye odaklı iki ConvFormer bloğuyla başlamaktadır. Sonraki iki aşamada art arda kullanılan RandFormer blokları sayesinde öğrenilen temsillere rastlantısallık ve çeşitlilik kazandırılmaktadır. Bu yaklaşım, güçlü yerel çıkarımı, dikkat yerine çeşitlendirme ile desteklemeyi amaçlamaktadır.

Model 4, ilk aşamada ConvFormer, ardından daha basit bir yapı olan IdentityFormer ile modelin erken aşamalarındaki parametre yükünü hafifletmekte; bunu takiben RandFormer ve Transformer bloklarıyla hem çeşitlilik hem de bağlamsal bilgi kazanımı sağlamaktadır. Bu model, düşük maliyetli başlangıç ve dengeli karmaşıklık artışı sunar.

Model 5, ilk iki aşamada art arda iki IdentityFormer bloğu kullanarak minimum bilgi işleme ile başlamakta, sonrasında ise ardışık iki Transformer bloğu ile doğrudan yüksek düzeyde küresel dikkat öğrenimine geçmektedir. Bu yapı, derin semantik ilişkileri daha erken safhada modellemeyi hedeflemektedir.

Model 6 ise ilk aşamada ConvFormer bloğu ile başlamakta, ardından art arda iki RandFormer bloğu ile çeşitlilik kazandırmakta ve son olarak Transformer bloğu ile global özellikleri öğrenmektedir. Bu model, yerel örüntüler üzerine rastgele karışımı entegre ederek son aşamada dikkat ile bütünleştirme yaklaşımını benimsemektedir.



Şekil 2- 27: Geliştirilen özgün hibrit modellerin aşamalı blok dizilimleri

Bu modeller, veri seti boyutunun küçük olduğu ve genel yapının aşırı öğrenmeye yatkın olduğu durumlarda modüler ve düşük parametrik çözümler üretmek amacıyla geliştirilmiştir. Blok türlerinin bu şekilde düzenlenmesi, ileride geliştirilecek otomatik blok seçimi (*AutoBlock Selection*) gibi yöntemler için de kuramsal bir temel oluşturma potansiyeline sahiptir. Görev uyarlamalı mimari düzenleme, özellikle mobil cihazlar, tarımsal görüntü sistemleri, sınırlı veriyle çalışan sağlık uygulamaları, sınır güvenliği, İHA görüntü analizi, askeri keşif, afet yönetimi gibi gerçek zamanlı sınıflandırma gerektiren kritik görevlerde düşük kaynak kullanımıyla yüksek genelleme başarımı elde etmeyi mümkün kılabilir. Bu gibi senaryolarda, aşırı öğrenmeye dirençli, açıklanabilir, hafif ve adapte edilebilir mimarilere duyulan ihtiyaç, bu tezde önerilen hibrit blok kombinasyonlarının uygulanabilirliğini artırmaktadır.

Bu bölümde geliştirilen mimariler, bu doğrultuda önerilmiş yapısal şablonlardır ve performans değerlendirmeleri izleyen bölümlerde ayrıntılı biçimde sunulacaktır.

2.6.1 Hiperparametre Ayarları

Bu çalışmada önerilen altı farklı MetaFormer tabanlı özgün mimari, aynı eğitim koşulları altında eğitilerek performansları karşılaştırmalı olarak analiz edilmiştir. Tüm modeller, giriş verisinin boyutu, eğitim süreci ve optimizasyon stratejileri bakımından eşit koşullarda test edilmiştir. Böylece performans farklarının yalnızca mimari yapıdaki blok dizilimlerinden kaynaklandığı varsayımı korunmuştur.

Model eğitim sürecinde kullanılan mimari yapılandırmalar Tablo 2-2’de ve ortak hiperparametre ayarları da Tablo 2-3’te sunulmuştur. Bu ayarlar MetaFormer iskeletine dayalı yapılar arasında doğrudan karşılaştırma yapılabilmesini sağlamak amacıyla sabit tutulmuştur.

Tablo 2- 2: Geliştirilen özgün hibrit modellerin katman temelli yapısal parametreleri

Aşama (Stage)	Çıkış Boyutu	Patch Size / Stride	Gömme Boyutu (Embed Dim)	Blok Sayısı (L)	MLP Ratio
1	$H/4 \times W/4$	$7 \times 7 / 4$	64	2	2
2	$H/16 \times W/16$	$3 \times 3 / 2$	128	2	2
3	$H/64 \times W/64$	$3 \times 3 / 2$	256	2	2
4	$H/64 \times W/64$	$3 \times 3 / 2$	512	2	2

Tablo 2- 3: Geliştirilen özgün hibrit modellerin ortak hiperparametre ayarları

Ayar	Değer
Epochs	50
Batch Size	16
Optimizer	Adam
Learning Rate	1e-4
Weight Decay	1e-5
Scheduler	ReduceLROnPlateau
Early Stopping	Patience=7
Loss Function	CrossEntropy
Input Resolution	224×224
Data Augmentations	Flip, Jitter, Rotate
Dropout (MLP içinde)	0.1
Normalizasyon	Post-Norm (BatchNorm2d)

Tablo 2- 4: Geliştirilen özgün hibrit modellerin parametre ve FLOPs değerleri

Model Adı	Parametre Sayısı (M)	FLOPs (GMac)
Model 1	9.39	4.75
Model 2	9.11	3.92
Model 3	7.01	3.43
Model 4	9.26	4.77
Model 5	9.78	5.69
Model 6	9.26	4.77

Tüm özgün modellerde bu yapısal konfigürasyon korunmuştur. Farklılık yalnızca kullanılan blok türlerinin (IdentityFormer, RandFormer, ConvFormer, Transformer) aşamalara göre farklı sırada yerleştirilmesidir.

2.7 Önerilen Model Mimarisi: DeformationScoreFormer

Son yıllarda görsel tanıma ve sınıflandırma görevlerinde Transformer tabanlı yapılar, özellikle self-attention mekanizmasının sunduğu uzun menzilli bağımlılık modelleme kapasitesi

sayesinde önemli başarılar elde etmiştir [102]. Ancak bu yapıların yüksek hesaplama maliyeti, belleğe olan yoğun gereksinimi ve düşük veri rejimlerinde performans düşüşü gibi sınırlılıkları, özellikle kaynak kısıtlı sistemlerde kullanımını güçleştirmektedir [101].

Bu bağlamda, Metaformer manifestosu [135] dikkat mekanizmasının mutlak bir gereklilik olmadığını, aksine LayerNorm → Token Mixer → LayerNorm → Channel MLP yapısının mimaride asıl belirleyici rolü oynadığını savunarak, çok daha esnek ve hesaplamasal olarak verimli yapıların önünü açmıştır. ConvFormer, PoolFormer ve RandFormer gibi Metaformer türevleri bu doğrultuda self-attention yerine alternatif karıştırıcılar ile etkili sonuçlar sunmuştur.

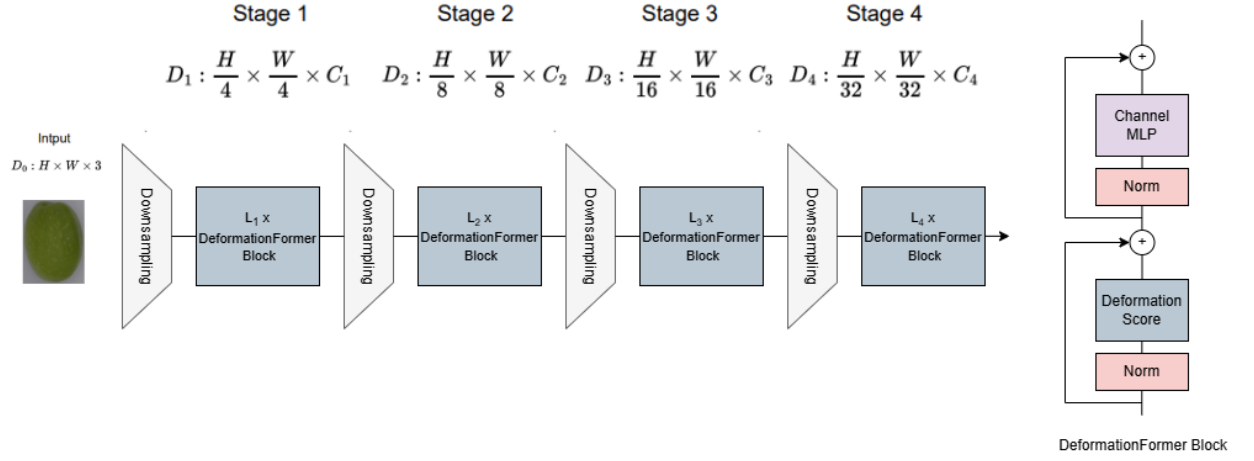
Bu tez çalışması kapsamında önerilen DeformationScoreFormer, MetaFormer iskeletine sadık kalmakla birlikte, özellikle tarımsal görüntü sınıflandırma gibi düşük çözünürlüklü ve görsel pürüzlere duyarlı alanlara özgü ihtiyaçlara cevap verebilmek üzere tasarlanmıştır. Tarım görüntülerinde, özellikle zeytin meyvesi gibi doğal objelerin yüzeyinde lokal bozulmalar (örneğin sinek ısırığı veya mantar lekesi) belirgin ve küçük ölçekli olabilir. Bu nedenle hem yerel örüntü çıkarımı yapabilen hem de hafif, modüler ve açıklanabilir bir yapıya sahip bir model mimarisi geliştirmek, bu alanda önemli bir gereklilik olarak ortaya çıkmaktadır.

DeformationScoreFormer’ın temel özgülüğü, her bloğa entegre edilen DeformationScore adlı yeni bir token mixer tasarımıdır. Bu yapı, her pikselin çevresine göre pürüzlük (smoothness) ve kenar yoğunluğu gibi düşük seviyeli yapısal bilgileri kullanarak öznitelik haritası üzerinde adaptif bir önemlilik skoru üretmektedir. Bu skor, feature map üzerine dinamik bir şekilde modülasyon uygulanmasını sağlar. Böylece model, self-attention’ın hesaplama yükünü taşımadan dikkat benzeri bir öne çıkarma işlemi gerçekleştirebilir.

Sonuç olarak DeformationScoreFormer, MetaFormer ilkeleriyle uyumlu, düşük parametrelili, açıklanabilir ve özellikle tarımsal görsel görevler için optimize edilmiş yenilikçi bir mimari olarak önerilmektedir.

2.7.1 Mimari Genel Bakış

Önerilen DeformationScoreFormer mimarisi, MetaFormer yapısal manifestosunu temel alarak tasarlanmış, modüler ve hiyerarşik bir görüntü sınıflandırma modelidir. Bu mimari ve model akışının genel yapısı, Şekil 2-28’te şematik olarak sunulmuştur. Model, girdi görüntüsünü çok aşamalı şekilde işleyen ve düşük seviyeden yüksek seviyeye doğru öznitelik derinliğini artıran bir yapı üzerine kuruludur.



Şekil 2- 28: DeformationScoreFormer mimarisinin aşama bazlı genel yapısı

2.7.2 DeformationFormer Bloğu

Bu tez kapsamında önerilen modelin temel yapı taşı, MetaFormer mimarisi çerçevesinde kurgulanan DeformationFormer bloğudur. Blok yapısı, iki paralel işlem yolundan oluşmaktadır: (i) token mixer yolu ve (ii) kanal MLP yolu. Her bir yol, özgün öznitelik dönüşümleri gerçekleştirerek öğrenme kapasitesini artırmakta; süreç sonunda her iki yolun çıktısı, residual bağlantılar aracılığıyla blok girişine eklenerek bilgi kaybı önlenmekte ve gradyan akışı desteklenmektedir. Tüm yapı, post-norm düzenine uygun olarak geliştirilmiştir.

2.7.2.1 Token Mixer Yolu

DeformationScore Token Mixer, MetaFormer manifestosunun "attention-free" ilkesine uygun olarak tasarlanmış özgün bir yapıdır. Bu modül, her pikselin lokal bağlamını ve kenar bilgisini kullanarak görelî önemini belirler ve bu skoru giriş tensörüne modülasyon yoluyla uygular. Böylece, evrimsel işlem öncesinde bilgi ağırlıklandırılarak dikkat etkisi kazandırılır. Bu yapı, özellikle yaprak, meyve ve yüzey dokularındaki varyasyonları hassas biçimde öne çıkarma amacı taşır.

Algoritma 1’de modülün işleyişine dair algoritmik özet sunulmuştur:

Algorithm 1: DeformationScore Token Mixer

Input: Feature map $x \in \mathbb{R}^{B \times C \times H \times W}$ **Output:** Modulated feature map $x' \in \mathbb{R}^{B \times C \times H \times W}$

```
1  $x_{gray} \leftarrow \frac{1}{C} \sum_{i=1}^C x_i$   
   // Kanal ortalaması alınarak gri tensör elde edilir.  
2  $smooth \leftarrow \text{AvgPool2d}(x_{gray})$   
3  $diff \leftarrow |x_{gray} - smooth|$   
   // Lokal pürüzlük bilgisi çıkarılır.  
4  $lap \leftarrow |Laplace(x_{gray})|$   
   // Kenar bilgisi hesaplanır.  
5  $score \leftarrow \sigma \left( \frac{diff + lap - \mu}{\sigma + \epsilon} \right)$   
   // Skor normalize edilerek sigmoid ile ölçeklenir.  
6  $score_{ref} \leftarrow \text{Conv}_{1 \times 1}(score)$   
   // 1x1 konvolüsyon ile refine edilir.  
7  $x' \leftarrow x + x \cdot score_{ref}$   
   // Girdi tensörü, skor ile module edilir.  
8 return:  $x'$ 
```

2.7.2.2 Kanal MLP Yolu

ChannelMLP modülü, her bir konvolüsyonel bloğun sonunda kanal boyutundaki ilişkileri modellemek amacıyla kullanılmaktadır. Girdi verisi $[B, C, H, W]$ boyutundan $[BHW, C]$ şekline dönüştürülmekte, bu düzlemde iki adet tam bağlantılı katman (Linear) üzerinden işlenmektedir. İlk lineer katmandan sonra GELU aktivasyon fonksiyonu ve Dropout uygulanmakta ardından ikinci lineer katman ile tekrar orijinal kanal boyutuna dönülmektedir. Çıkış, $[B, H, W, C]$ boyutuna çevrilip tekrar $[B, C, H, W]$ biçimine dönüştürülür.

Bu yapı, Transformer mimarisinden ilhamla oluşturulan ve Metaformer manifestosuna dayalı modellerde yaygın olarak kullanılan bir kanal temelli MLP tasarımıdır. ChannelMLP, özellikle konvolüsyonel blokların ardından eklenen hafif ve açıklanabilir bir kanal işlemcisi olarak hem öğrenme kapasitesini artırmakta hem de parametre verimliliğini korumaktadır.

Bu klasik yapı Metaformer manifestosunda önerilen post-norm ilkesine uygun olarak konumlandırılmıştır yani MLP işlemi tamamlandıktan sonra normalizasyon ve residual birleştirme gerçekleştirilmiştir. Bu sayede modelin öğrenme süreci daha dengeli ve genellenebilir hale getirilmiştir.

2.7.3 Aşama Tabanlı Mimari ve Çıkış Boyutları

DeformationScoreFormer mimarisi, dört aşamalı hiyerarşik bir yapı üzerine inşa edilmiştir. Bu yapı, klasik ConvNet mimarilerinden esinlenerek her aşamada hem çözünürlüğün düşürülmesini hem de kanal sayısının artırılmasını amaçlamaktadır. Girdi çözünürlüğü 224×224 olan bir görüntü, sırasıyla Stem ve Downsample katmanlarından geçerek daha düşük çözünürlüklü fakat daha derin özellik temsillerine dönüştürülür.

Her aşama, belirli sayıda DeformationFormerBlock içerir. Bu bloklar, hem yerel pürüzlük ve kenar bilgisini içeren özgün token mixer yapısını hem de kanal boyutundaki ilişkileri modelleyen Channel MLP bileşenini içermektedir.

Tablo 2-5’de her bir aşamanın çıkış boyutları, gömme (embedding) boyutları ve kullanılan blok sayıları özetlenmiştir:

Tablo 2- 5: DeformationScoreFormer mimarisinin katman düzeyinde yapısal özellikleri

Stage	Çıkış Boyutu	Patch Size / Stride	Embed Dim	Block (L)	Token Mixer	MLP Ratio
1	$H/4 \times W/4$	$7 \times 7 / 4$	64	2	DeformationScore	2
2	$H/16 \times W/16$	$3 \times 3 / 2$	128	2	DeformationScore	2
3	$H/64 \times W/64$	$3 \times 3 / 2$	256	2	DeformationScore	2
4	$H/64 \times W/64$	$3 \times 3 / 2$	512	2	DeformationScore	2
Toplam Parametre Sayısı $\approx 10.63 \text{ M}$					MACs $\approx 1.55 \text{ GMac}$	

Bu yapı sayesinde model hem yerel hem de küresel bağlamları çok katmanlı bir düzlemde işleyebilmektedir. Özellikle tarımsal görüntülerde sıkça karşılaşılan küçük alanlı hasar izleri ve geniş yüzeysel anomalilerin birlikte modellenmesi açısından bu tür çok ölçekli yaklaşımlar kritik öneme sahiptir.

Modelin toplam parametre sayısı 10,63 milyon, FLOPs değeri ise 1.55 GMac civarındadır. Bu değerler, sınırlı donanım kaynaklarında dahi verimli çalışabilecek şekilde optimize edilmiş bir yapı elde edildiğini göstermektedir.

2.7.4 Eğitim Yapılandırmaları ve Hiperparametreler

DeformationScoreFormer mimarisi, zeytin meyvesi görüntülerinden hastalık sınıflandırması görevine uygun olarak optimize edilmiştir. Bu bölümde modelin eğitim süreci, kullanılan hiperparametreler ve veri artırma stratejileri açıklanmaktadır.

Tablo 2- 6: DeformationScoreFormer mimarisinin eğitim yapılandırmaları

Ayar	Değer
Epochs	50
Batch size	16
Optimizer	Adam
Learning Rate	1e-4
Weight Decay	1e-5
Scheduler	ReduceLROnPlateau
Early Stopping	patience=7
Loss Function	CrossEntropy
Input Resolution	224x224
Data Augmentations	Flip, Jitter, Rotate
Dropout	0.1 (MLP içinde)

Modelde kullanılan tüm normalization katmanları, işlem yollarının ardından uygulanacak şekilde yapılandırılmıştır (post-norm). Bu tercih, MetaFormer mimarilerinde önerilen 'mixer → norm' ve 'mlp → norm' düzenine uygunluğu pekiştirmektedir.

2.7.5 Modelin Matematiksel Açıklaması

Önerilen DeformationScoreFormer mimarisi, MetaFormer ailesinin genel matematiksel çerçevesine sadık kalmakla birlikte, her bloğun içinde hem özgün bir token mixer (DeformationScore) hem de klasik bir Channel MLP bileşeni barındırmaktadır. Her blok, iki ayrı yol üzerinden bilgi işlemesi gerçekleştirmekte ve bu yolların çıktıları residual bağlantılar ile özgün girişe geri eklenmektedir. Bu yapı, aşağıda detaylandırılan işlemler dizisini takip eder:

(1) DeformationScore Token Mixer

İlk yol, görsel girdiye göre yerel bağlamı dikkate alan bir modülasyon skoru üretir. Bu yol aşağıdaki işlemlerle tanımlanır:

Aşama	İşlem Açıklaması	Formül	Çıktı Boyutu
Grayscale Dönüşüm	RGB girişten ortalama alınarak gri seviye tensör oluşturulur.	$x_{gray} = \frac{1}{C} \sum_{i=1}^C x_i$	$[B, 1, H, W]$ (2.4)
Pürüzlük (Smoothness)	Gri tensöre ortalama havuzlama uygulanır ve orijinal tensörle fark alınır.	$diff = x_{gray} - AvgPool(x_{gray}) $	$[B, 1, H, W]$ (2.5)

Kenar Çıkarımı	Gri tensöre Laplace filtresi uygulanarak kenar bilgisi elde edilir.	$lap = Laplace(x_{gray}) $	$[B, 1, H, W]$	(2.6)
Skor Hesaplama	Diff ve lap toplanır, normalize edilir ve sigmoid aktivasyon uygulanır.	$score = \sigma\left(\frac{ diff + lap -\mu}{\sigma+\epsilon}\right)$	$[B, 1, H, W]$	(2.7)
Rafine Skor Üretimi	Skor, 1×1 konvolüsyon filtresiyle rafine edilir.	$score_{ref} = Conv_{1 \times 1}(score)$	$[B, 1, H, W]$	(2.8)
Modülasyon ve Çıkış	Rafine skor, giriş tensörüne uygulanarak dikkat etkisiyle modüle edilir.	$x' = x + x \cdot score_{ref}$	$[B, C, H, W]$	(2.9)

(2) Kanal MLP Yolu

İkinci yol, kanal içi ilişkilerin modellenmesini sağlar ve klasik MLP yapısıyla gerçekleştirilir:

Aşama	İşlem Açıklaması	Formül	Çıktı Boyutu	
Yeniden Şekillendirme	Girdi tensörü $[B, C, H, W]$ formatından $[B \times H \times W, C]$ formatına çevrilir.	$x_{flat} = Reshape(x') \in \mathbb{R}^{(B \cdot H \cdot W)C}$	$(B \times H \times W), C$	(2.10)
İlk Lineer Katman + Aktivasyon	Kanal sayısı $\gamma=2$ katsayısıyla genişletilir ve GELU aktivasyon uygulanır.	$h = GELU(x_{flat}W_1 + b_1), W_1 \in \mathbb{R}^{C \times \gamma C}$	$(B \times H \times W), \gamma C$	(2.11)
Dropout	Overfitting'i azaltmak için dropout uygulanır.	$h' = Dropout(h), p = 0.1$	$(B \times H \times W), \gamma C$	(2.12)
İkinci Lineer Katman	Kanal boyutu tekrar C 'ye indirgenir.	$x_{mlp} = h'W_2 + b_2, W_2 \in \mathbb{R}^{C \times \gamma C}$	$(B \times H \times W), C$	(2.13)
Geri Şekillendirme	Çıktı yeniden $[B, C, H, W]$ formatına çevrilir.	$x_{mlp} \rightarrow \mathbb{R}^{B \times C \times H \times W}$	$[B, C, H, W]$	(2.14)

BatchNorm +	Batch normalization	$x_{out} = BN(x_{mlp}) + x$	$[B, C, H, W]$	(2.15)
Residual	uygulanır ve residual bağlantı ile girişle toplanır.			

(3) Blok Çıkışı

Her DeformationFormer bloğu, iki yolun ardışık uygulanması sonucunda elde edilen çıktıyı şu şekilde oluşturur:

$$x_{block} = F_{mlp}(F_{mixer}(x)) + x \quad (2.16)$$

2.8 Değerlendirme Metrikleri

Karışıklık matrisi, bir sınıflandırma modelinin, doğru etiketleri bilinen bir veri kümesi üzerindeki performansını değerlendirmek amacıyla sıklıkla kullanılan tablo biçiminde bir gösterimdir. Bu yapı, modelin yaptığı tahminleri özetleyerek doğruluk düzeyine ilişkin net bir görünüm sunar ve iyileştirmeye açık yönlerin tespit edilmesine katkı sağlar. Matriste satırlar gerçek sınıfları, sütunlar ise modelin tahmin ettiği sınıfları temsil eder böylece modelin farklı sınıfları ne ölçüde ayırt edebildiği kolaylıkla analiz edilebilir.

Karışıklık matrisi, aynı zamanda doğru pozitif (TP), yanlış pozitif (FP), doğru negatif (TN) ve yanlış negatif (FN) gibi önemli sınıflandırma çıktılarının da görünür hale gelmesini sağlar. Bu sayede modelin güvenilirliği ve başarımı daha bütüncül biçimde değerlendirilebilir [132, 133, 134].

Bu matris üzerinden türetilen temel performans metrikleri şunlardır:

- Doğruluk (Accuracy): Modelin yaptığı tüm tahminler içindeki doğru sınıflandırmaların oranını ifade eder. Modelin genel başarısını ölçen temel bir göstergedir.

- Kesinlik (Precision): Modelin pozitif olarak tahmin ettiği örneklerden kaç tanesinin gerçekten pozitif olduğunu gösterir. Yanlış pozitiflerin etkisini azaltmak açısından önemlidir.

- Duyarlılık / Yakalama Oranı (Recall / Sensitivity): Gerçek pozitif örneklerin model tarafından ne kadarının doğru şekilde tanındığını gösterir. Özellikle eksik tespitin kritik olduğu durumlarda önemlidir.

- F1 Skoru (F1 Score): Precision ve Recall metriklerinin harmonik ortalamasıdır. Bu metrik, modelin hem doğruluk hem de duyarlılık açısından dengeli bir performans gösterip göstermediğini değerlendirir.

3. BULGULAR

3.1 Transfer Öğrenme Tabanlı Mimarilerinin Karşılaştırmalı Analizi

Bu çalışmada, transfer öğrenme temelli derin öğrenme mimarilerinin çok sınıflı görüntü sınıflandırma görevindeki başarımlarını karşılaştırmak amacıyla, kapsamlı bir ön işleme ve modelleme süreci yürütülmüştür. İlk aşamada, tüm görüntüler, önceden eğitilmiş modellerin beklediği giriş boyutlarıyla uyumlu hâle getirilerek 224×224 piksel çözünürlüğe yeniden ölçeklendirilmiştir. Piksel yoğunlukları, 0–255 aralığından [0, 1] aralığına normalize edilmiş bu sayede modelin daha stabil ve hızlı öğrenmesi hedeflenmiştir.

Sınıflandırma problemi çok sınıflı bir yapı içerdiğinden, sınıf etiketleri one-hot kodlama yöntemiyle dönüştürülmüş ve modelin çıktısında categorical_crossentropy kayıp fonksiyonuyla uyumlu bir çıktı katmanı oluşturulmuştur. Görüntülerin yüklenmesi sürecinde, sınıflara göre dizinlenmiş klasör yapısı kullanılmış ve flow_from_directory() fonksiyonu tercih edilmiştir.

Modelleme sürecinde, ImageNet veri seti üzerinde önceden eğitilmiş dokuz farklı mimari kullanılarak transfer öğrenme gerçekleştirilmiştir: VGG16, ResNet50, InceptionV3, MobileNet, EfficientNetB0, DenseNet121, NASNetMobile, Xception ve InceptionResNetV2. Bu modeller, yalnızca üst sınıflandırıcı katman yeniden eğilecek biçimde yapılandırılmış; derin katmanlar dondurularak transfer edilen bilginin korunması ve aşırı öğrenmenin (overfitting) önlenmesi amaçlanmıştır.

Model başarımını güvenilir biçimde değerlendirmek amacıyla, eğitim–doğrulama süreci hem Hold-Out yöntemi ile sabit veri bölünmesi yapılarak hem de 10 katlı çapraz doğrulama (K-Fold) tekniği ile yinelenabilirlik esas alınarak yürütülmüştür.

3.1.1 Transfer Öğrenme Modellerinin Performans Karşılaştırması

3.1.1.1 Hold-Out Yöntemi ile Elde Edilen Sonuçlar

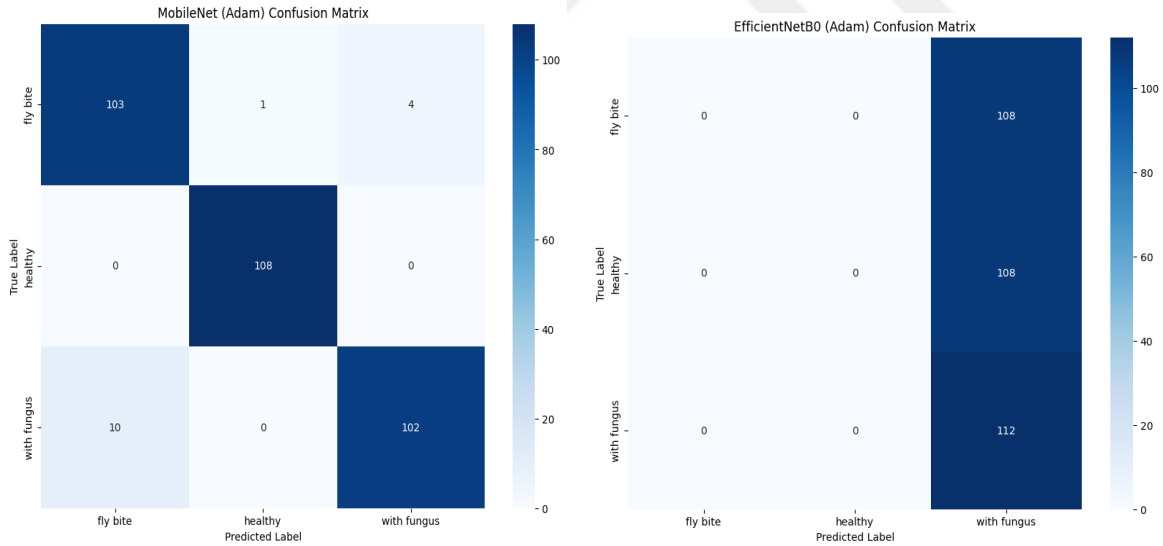
Bu bölümde, transfer öğrenme tabanlı modellerin Hold-Out yöntemiyle eğitildiği senaryoya ilişkin sınıflandırma başarımları sunulmaktadır. Söz konusu yöntemde, veri seti %80 eğitim ve %20 doğrulama olacak şekilde sabit oranla ikiye ayrılmıştır. Eğitim süreci boyunca doğrulama işlemleri sabit doğrulama kümesi üzerinden gerçekleştirilmiştir. Dokuz farklı modelin doğruluk (accuracy), F1 skoru, hassasiyet (precision), duyarlılık (recall) ve karışıklık matrisi (confusion matrix) gibi temel performans ölçütleri değerlendirilmiştir.

Modeller, yalnızca üst sınıflandırıcı katmanları eğitilecek şekilde yapılandırılmış ve önceden eğitilmiş gövde katmanları sabitlenerek transfer edilen bilginin korunması ve aşırı öğrenmenin önlenmesi amaçlanmıştır.

Modellerin nicel performans karşılaştırmalarına ilişkin bulgular Tablo 3.1’de, sınıflandırma doğruluğuna ilişkin karmaşıklık matrisleri ise Şekil 3-1’de sırasıyla en iyi ve en kötü performans sonuçları şeklinde sunulmaktadır. İlgili tablo ve şekiller, farklı mimarilerin sınıflar arası ayırt edicilik düzeylerini karşılaştırmalı olarak ortaya koymaktadır.

Tablo 3- 1: Modellerin Hold-Out yöntemiyle elde edilen doğrulama kümesindeki genel sınıflandırma performansları

Model	Doğruluk	Kesinlik	Duyarlılık	F1-Skoru
VGG16	0.8079	0.81	0.81	0.81
ResNet50	0.5976	0.73	0.60	0.56
InceptionV3	0.8994	0.90	0.90	0.90
MobileNet	0.9451	0.95	0.95	0.95
EfficientNetB0	0.3415	0.11	0.33	0.17
DenseNet121	0.9116	0.93	0.92	0.92
NASNetMobile	0.9238	0.93	0.93	0.93
Xception	0.9207	0.91	0.91	0.91
InceptionResNetV2	0.9177	0.92	0.92	0.91



Şekil 3- 1: Modellerin Hold-Out yöntemiyle elde edilen karmaşıklık matrisleri

3.1.1.2 K-Fold Çapraz Doğrulama Sonuçları

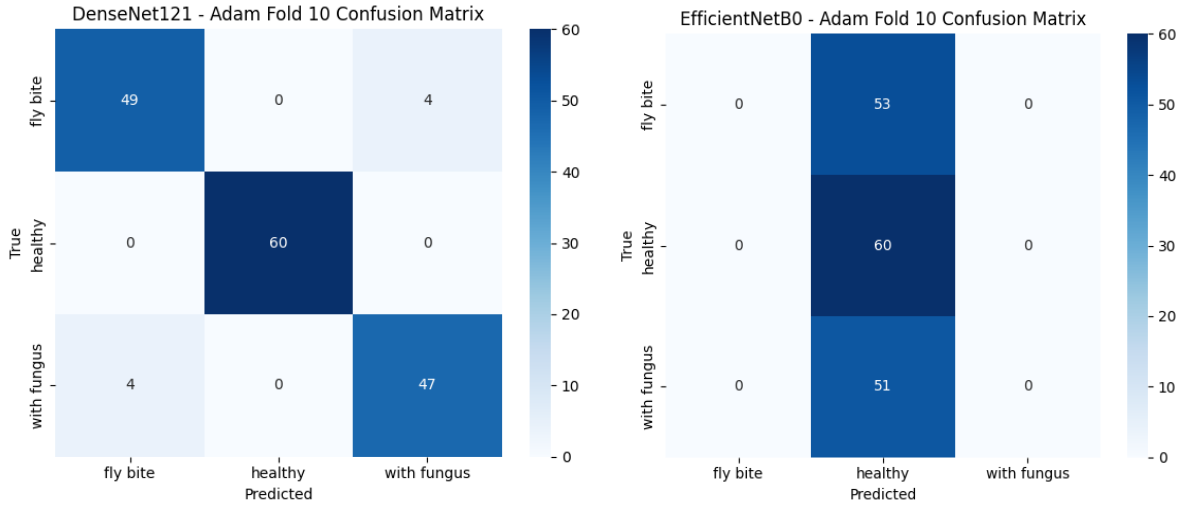
Bu bölümde, transfer öğrenme tabanlı modellerin 10 katlı çapraz doğrulama (K-Fold) yöntemi ile eğitildiği senaryo kapsamında elde edilen sınıflandırma performansları sunulmaktadır. Bu yöntemde, veri seti rastgele ve eşit büyüklükte 10 alt kümeye ayrılmış; her kat, bir kez doğrulama kümesi olarak kullanılırken kalan dokuz kat eğitim için değerlendirilmiştir. Bu sayede tüm veri kümesi hem eğitim hem doğrulama sürecinde yer almakta böylece modelin genelleme kabiliyeti daha güvenilir biçimde test edilmektedir.

Modellerin yapısal özellikleri, Hold-Out yönteminde olduğu gibi yalnızca üst sınıflandırıcı katmanların yeniden eğitileceği şekilde yapılandırılmış ve derin katmanlar sabitlenerek transfer edilen bilginin korunması amaçlanmıştır.

On katlı çapraz doğrulama sonunda elde edilen ortalama doğruluk değerleri, Tablo 3-2’de özetlenmekte ve sınıf tahminleme başarımlarına ilişkin karmaşıklık matrisleri ise Şekil 3-2’de sırasıyla en iyi ve en kötü performans sonuçları olarak sunulmaktadır. Tablo ve şekiller, farklı transfer öğrenme mimarilerinin çok sınıflı görüntü sınıflandırma görevindeki genel başarımlarını kapsamlı biçimde karşılaştırmalı olarak ortaya koymaktadır.

Tablo 3- 2: Modellerin 10-Fold çapraz doğrulama elde edilen genel sınıflandırma performansları

Model	Doğruluk	Kesinlik (Makro)	Duyarlılık (Makro)	F1 Skoru (Makro)
DenseNet121	0.9817	0.98	0.98	0.98
Xception	0.9390	0.94	0.94	0.94
MobileNet	0.9207	0.92	0.92	0.92
InceptionResNetV2	0.9207	0.92	0.92	0.92
NASNetMobile	0.9146	0.91	0.91	0.91
InceptionV3	0.8902	0.89	0.89	0.89
VGG16	0.8659	0.87	0.86	0.86
ResNet50	0.6646	0.64	0.65	0.61
EfficientNetB0	0.3659	0.12	0.33	0.18



Şekil 3- 2: Modellerin 10-Fold çapraz doğrulama elde edilen karmaşıklık matrisleri

3.1.2 Modellerin Karşılaştırmalı Değerlendirilmesi

Bu bölümde, transfer öğrenme temelli dokuz farklı derin öğrenme mimarisinin sınıflandırma görevlerindeki başarımı, kullanılan doğrulama stratejilerine bağlı olarak karşılaştırmalı biçimde analiz edilmektedir. Model başarısının yalnızca mimari karmaşıklık ya da önceden eğitilmiş ağırlıklarla değil aynı zamanda eğitim verisinin yapısı ve doğrulama yöntemiyle kurduğu etkileşimle şekillendiği dikkate alınarak, tüm modeller hem sabit veri bölmeli Hold-Out yöntemi hem de daha genelleyici bir yapı sunan 10 katlı çapraz doğrulama (K-Fold) yaklaşımı çerçevesinde test edilmiştir. Elde edilen sonuçlar, modellerin başarı düzeyleri, doğrulama stratejilerine gösterdiği duyarlılık ve sınıflar arası ayırt edicilik yetenekleri açısından belirgin farklılıklar taşıdığını ortaya koymaktadır.

Her iki doğrulama yönteminde de öne çıkan modellerin başında DenseNet121, Xception, MobileNet, NASNetMobile ve InceptionResNetV2 gelmektedir. Bu yapılar, genel doğruluk oranlarının yanı sıra kesinlik, duyarlılık ve F1 skoru gibi ölçütlerde de dengeli ve yüksek performans sergilemiştir. Özellikle DenseNet121, K-Fold yöntemi ile %98 üzerinde doğruluk ve 0.98 F1 skoru ile en üst başarıya ulaşmış aynı zamanda sınıflar arasında tutarlı bir ayırım gücü göstermiştir. Xception ve InceptionResNetV2 modelleri de üç sınıfta dengeli performans ortaya koymuş, hem Hold-Out hem de K-Fold sonuçları arasında tutarlılık sergilemiştir. MobileNet, Hold-Out yöntemiyle %94,51 gibi çok yüksek bir doğruluk değeri yakalarken, K-Fold ile bu oran %92'ye gerilemiş olsa da başarı düzeyi korunmuştur. NASNetMobile ise düşük parametre yüküne

rağmen yüksek duyarlılık ve doğruluk değerleri ile özellikle mobil ve gömülü sistemler açısından dikkat çeken bir mimari olmuştur.

InceptionV3 ve VGG16 modelleri, başarı sıralamasında orta düzeyde kalmakla birlikte bazı sınıflarda güçlü sonuçlar üretmiştir. InceptionV3, özellikle ‘sağlıklı’ sınıfında yüksek doğruluk sağlamış ancak ‘mantar hastalıklı’ ve ‘sinek ısırıklı’ sınıfları arasında zaman zaman karışıklıklar göstermiştir. VGG16 ise mimari olarak daha eski ve derin olmayan bir yapıya sahip olmasına rağmen, %86 civarındaki F1 skoru ile beklenenin üzerinde bir başarı sergilemiştir. Bu iki modelin performansları, sınıf bazlı duyarlılıkta hafif dengesizlikler barındırsa da genel görev başarımı açısından uygulanabilir düzeyde kalmıştır.

Buna karşın, ResNet50 ve EfficientNetB0 modelleri her iki doğrulama yönteminde de zayıf performans göstermiştir. ResNet50, özellikle ‘sinek ısırıklı’ sınıfında belirgin hata örüntüleri üretmiş; Hold-Out yöntemiyle %59 doğruluk ve yalnızca 0.56 F1 skoru elde edebilmiştir. K-Fold yöntemiyle bu oran %66’ya yükselmişse de sınıflar arası ayrım başarısızlığı devam etmiştir. EfficientNetB0 ise her iki senaryoda da ciddi biçimde düşük F1 skorları (Hold-Out: 0.17, K-Fold: 0.18) üretmiş; bazı sınıflarda tüm örnekleri tek bir etiket altında toplama eğilimi göstermiştir. Bu durum, modelin küçük ve dengesiz veri setlerinde öğrenme başarısı gösteremediğini ortaya koymuştur.

Genel olarak değerlendirildiğinde, K-Fold doğrulama yöntemi modellerin genelleme kapasitesini ölçmede daha sağlıklı bir çerçeve sunmuştur. Özellikle sınıf dengesizliği içeren veri kümelerinde, Hold-Out yöntemi bazı modellerin performansını olduğundan yüksek gösterebilmiş oysa K-Fold yöntemi tüm veriyi hem eğitim hem de doğrulama aşamasına dahil ederek daha dengeli bir değerlendirme ortamı sağlamıştır. Bu nedenle örneklem boyutu sınırlı olan çalışmalarda K-Fold yaklaşımının tercih edilmesi daha güvenilir sonuçlara ulaşılmasını mümkün kılmaktadır.

Sonuç olarak, DenseNet121, Xception ve NASNetMobile modelleri hem genel başarı hem de sınıf ayrımı açısından öne çıkan yapılar olmuştur. Bu mimariler transfer öğrenme temelli görüntü sınıflandırma görevleri için güvenilir seçenekler olarak değerlendirilebilir. Buna karşın, ResNet50 ve EfficientNetB0 gibi modellerin, küçük örneklem setleri karşısında sınırlı genelleme kabiliyeti nedeniyle dikkatli biçimde ele alınması gerektiği anlaşılmaktadır. Başarıyı belirleyen unsur yalnızca mimari karmaşıklık değil aynı zamanda modelin öğrenme stratejileri ile veri seti özellikleri arasındaki uyumdur. Bu bağlamda model seçimi veri yapısına özgü olarak şekillendirilmeli ve doğrulama yöntemi, değerlendirme güvenilirliğini artıracak biçimde yapılandırılmalıdır.

3.2 Transformer Tabanlı Görüntü Sınıflandırma Modelleri ve Optimizasyon Yöntemlerinin Karşılaştırmalı Analizi

Bu bölümde, zeytin meyvesine ait üç sınıflı görüntü verisi üzerinde gerçekleştirilen sınıflandırma görevine yönelik altı farklı Transformer temelli mimarinin (ViT, BEiT, DeiT, MobileViT, MaxViT, LeViT) performans sonuçları sunulmaktadır. Söz konusu modeller, hem global bağlam yakalama kapasitesi hem de çok ölçekli öznitelik işleme yetenekleri bakımından çeşitlilik arz etmektedir. Bununla birlikte, her bir model, farklı öğrenme dinamiklerine sahip 10 ayrı optimizasyon algoritması (SGD, Adam, AdamW, RMSprop, Adagrad, Adadelata, Adamax, NAdam, RAdam, Adafactor) ile eğitilerek, model-eğitim stratejisi etkileşimi ayrıntılı biçimde analiz edilmiştir. Performans değerlendirmesinde doğruluk (%Accuracy), makro F1 skoru, kayıp değeri (Loss), makro duyarlılık (Recall) ve makro kesinlik (Precision) gibi temel metrikler kullanılmıştır. Bulgular, model mimarisi ve optimizasyon yöntemi arasındaki ilişkinin sınıflar arası ayırım başarımını önemli ölçüde etkileyebileceğini ortaya koymaktadır. Aşağıda, her bir Transformer mimarisi için optimizasyon bazında elde edilen performans çıktıları karşılaştırmalı olarak sunulmaktadır.

3.2.1 BEiT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi

BEiT modeli için uygulanan farklı optimizasyon algoritmaları, modelin sınıflandırma başarımında bazı düzeylerde farklılıklar ortaya koymuştur. Tablo 3.3’de sunulan sınıflandırma sonuçlarına göre, en düşük kayıp değeri (0.0125) ve en yüksek doğruluk (%98,38) ile SGD algoritması öne çıkmaktadır. Ayrıca, RAdam algoritması da %97,98 doğruluk ve 0.9798 F1 skoru ile oldukça başarılı sonuçlar üretmiştir. Bu durum, momentuma dayalı ancak adaptif olmayan algoritmaların BEiT mimarisiyle daha tutarlı çalışabileceğini göstermektedir.

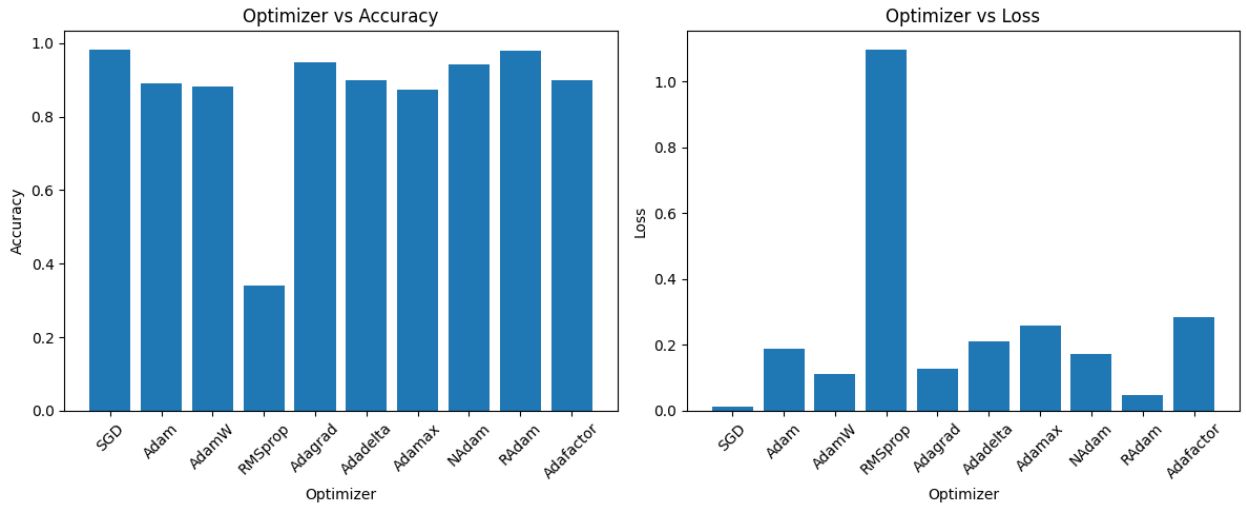
Tablo 3- 3: BEiT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları

Optimizer	Best Loss	Accuracy	Macro F1-score	Macro Precision	Macro Recall
SGD	0.0125	0.9838	0.9839	0.9845	0.9838
Adam	0.1889	0.8907	0.8899	0.9134	0.8895
AdamW	0.1102	0.8826	0.8778	0.8939	0.8822
RMSprop	1.0988	0.3401	0.1692	0.1134	0.3333
Adagrad	0.1291	0.9474	0.9477	0.9509	0.9472
Adadelata	0.2095	0.8988	0.8982	0.9141	0.8977
Adamax	0.2599	0.8745	0.8728	0.8835	0.8742

NAdam	0.1731	0.9433	0.9436	0.9463	0.9431
RAdam	0.0462	0.9798	0.9798	0.9798	0.9800
Adafactor	0.2837	0.8988	0.8988	0.9097	0.8979

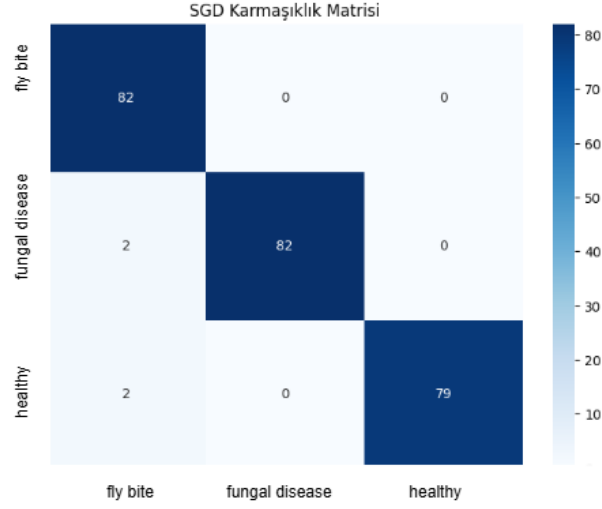
Buna karşın, yaygın olarak tercih edilen Adam ve AdamW algoritmaları sırasıyla %89,07 ve %88,26 doğruluk oranlarıyla daha düşük performans sergilemiştir. NAdam ve Adagrad, %94'ün üzerinde doğruluk ile daha dengeli sonuçlar verirken RMSprop, %34.01 doğruluk ve 0.1692 F1 skoru ile açık ara en başarısız algoritma olarak dikkat çekmiştir. Adafactor ve Adadelta gibi bellek dostu yöntemler ise ortalama düzeyde kalmıştır.

Ayrıca, doğruluk ve kayıp değerlerinin grafiksel karşılaştırmaları Şekil 3.3'de sunulmuştur. Grafikler, SGD ve RAdam'ın düşük kayıpla yüksek doğruluk sağladığını, buna karşın RMSprop'un tüm metriklerde ayrıştığını net biçimde göstermektedir.



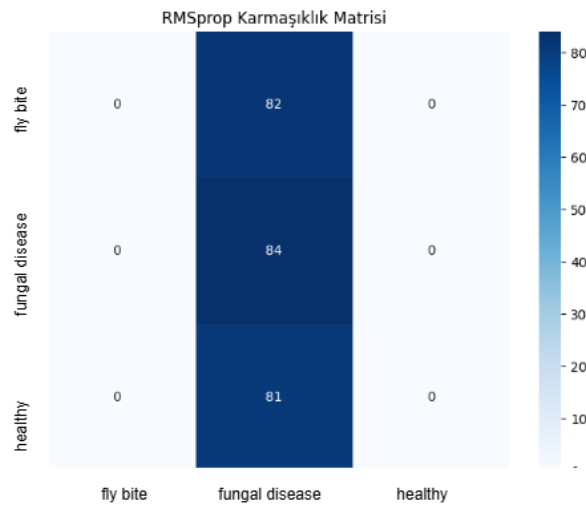
Şekil 3- 3: BEiT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri

Karmaşıklık matrisleri incelendiğinde Şekil 3-4'de BEiT modeli ve SGD optimizasyon yöntemiyle elde edilen karmaşıklık matrisi, modelin sınıflar arası ayırım performansının oldukça yüksek olduğunu göstermektedir. Model, her üç sınıfa ait örnekleri büyük oranda doğru şekilde sınıflandırmıştır. Özellikle “sinek ısırıklı” sınıfında %100'e yakın doğruluk sağlanmış, bu sınıfa ait tüm örnekler doğru sınıflandırılmıştır. “Mantar hastalıklı” ve “sağlıklı” sınıflarında ise yalnızca ikişer örnek farklı sınıfa karışmış olup, bu durum modelin genel başarısını anlamlı düzeyde etkilememektedir. Elde edilen bu sonuçlar, SGD algoritmasının BEiT mimarisiyle birlikte etkili bir optimizasyon sunduğunu göstermektedir.



Şekil 3- 4: BEiT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

Şekil 3-5’de ise BEiT modeli ile RMSprop optimizasyon yöntemi kullanılarak elde edilen karmaşıklık matrisi, modelin sınıflandırma başarısında ciddi bozulmalar yaşadığını göstermektedir. Tüm örneklerin yalnızca “fungal disease” sınıfına atanmış olması, modelin bu sınıf dışındaki kategorileri ayırt etme yetisini neredeyse tamamen kaybettiğini göstermektedir. “Fly bite” ve “healthy” sınıflarına ait hiçbir örnek doğru şekilde sınıflandırılamamıştır. Bu durum, RMSprop algoritmasının BEiT mimarisiyle birlikte uyumsuz çalıştığını ve öğrenme sürecinde yeterli ayrıştırıcı temsilleri öğrenemediğini göstermektedir. Özellikle hassas sınıflandırma gerektiren görevlerde bu tür hatalar sistem performansını olumsuz etkileyebilir.



Şekil 3- 5: BEiT Modelinde En Düşük Sınıflandırma Başarısını Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

3.2.2 DeiT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi

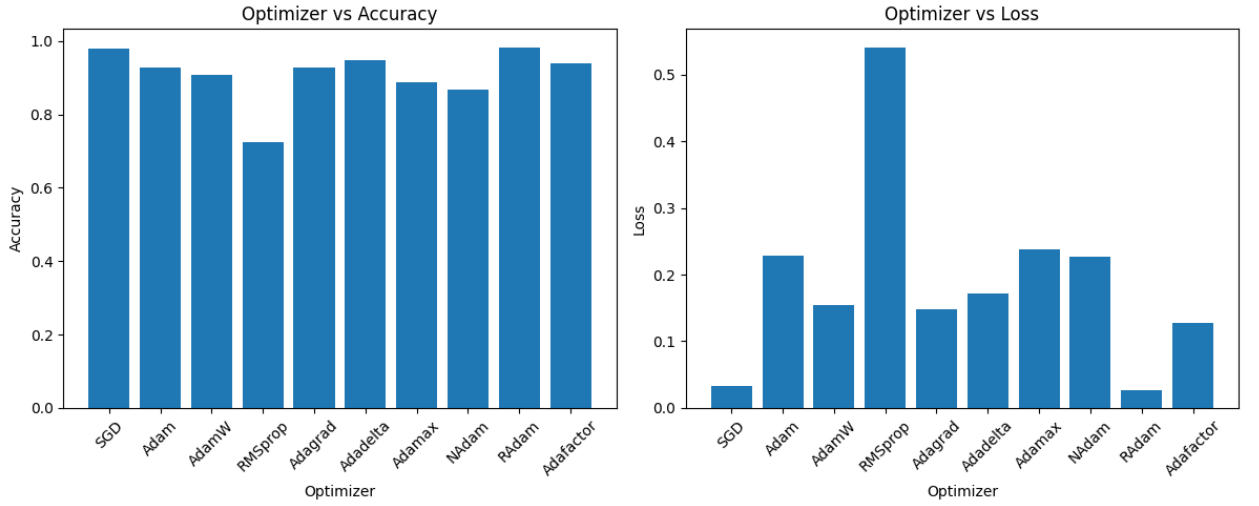
DeiT modeli için farklı optimizasyon algoritmalarıyla elde edilen sınıflandırma performansları, modelin öğrenme başarımının kullanılan optimizasyon stratejisine duyarlı olduğunu ortaya koymaktadır. Tablo 3-4’de sunulan sonuçlara göre, en düşük kayıp değeri (0.0273) ve en yüksek doğruluk oranı (%98,38) ile RAdam algoritması, DeiT modeli için en yüksek başarıyı göstermiştir. Bu performans, aynı zamanda 0.9838 makro F1 skoru ve dengeli precision-recall değerleri ile desteklenmiştir. RAdam’ın bu başarısı, DeiT mimarisinde ağırlık güncellemelerinin istikrarlı bir şekilde optimize edilebildiğini göstermektedir.

Tablo 3- 4: DEiT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları

Optimizer	Best Loss	Accuracy	Macro F1-score	Macro Precision	Macro Recall
SGD	0.0335	0.9798	0.9799	0.9800	0.9798
Adam	0.2290	0.9271	0.9277	0.9352	0.9268
AdamW	0.1543	0.9069	0.9069	0.9082	0.9065
RMSprop	0.5414	0.7247	0.7138	0.7213	0.7224
Adagrad	0.1477	0.9271	0.9275	0.9315	0.9267
Adadelat	0.1714	0.9474	0.9477	0.9497	0.9472
Adamax	0.2379	0.8866	0.8859	0.9073	0.8855
NAdam	0.2273	0.8664	0.8660	0.8788	0.8651
RAdam	0.0273	0.9838	0.9838	0.9842	0.9841
Adafactor	0.1282	0.9393	0.9396	0.9417	0.9390

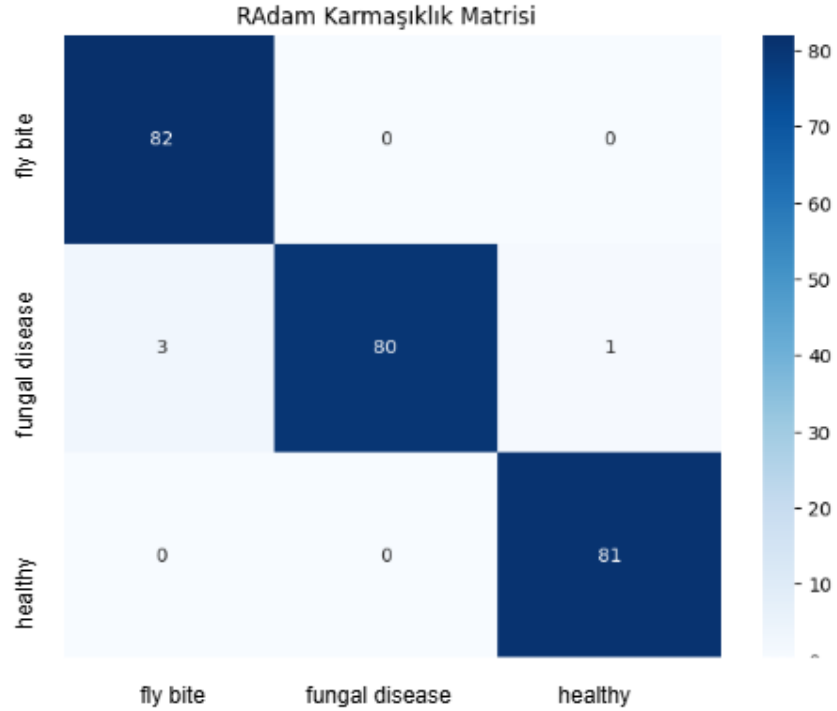
Benzer şekilde, SGD algoritması %97,98 doğruluk ve 0.0335 kayıp değeri ile en yüksek başarıyı gösterirken; Adadelat (%94,74) ve Adafactor (%93,93) gibi alternatif yöntemler de düşük kayıp ve yüksek makro F1 skorlarıyla dikkat çekmiş, özellikle Adafactor sınıflar arası denge açısından etkili bir performans sergilemiştir. Öte yandan Adamax, NAdam ve AdamW algoritmaları, sırasıyla %88,66, %86,64 ve %90,69 doğruluk seviyelerinde kalmış; RMSprop algoritması ise diğer tüm algoritmalarından ayrılarak yalnızca %72,47 doğruluk ve 0.7138 F1 skoru ile en düşük başarıyı göstermiştir.

Bu farklılıkları daha net biçimde göstermek amacıyla oluşturulan Şekil 3-6 'daki doğruluk ve kayıp grafikleri, RAdam ve SGD’nin yüksek doğruluk ve düşük kayıp değerleri ile öne çıktığını; RMSprop’un ise hem yüksek kayıp hem de düşük doğruluk değerleriyle başarısız olduğunu açık biçimde ortaya koymaktadır.



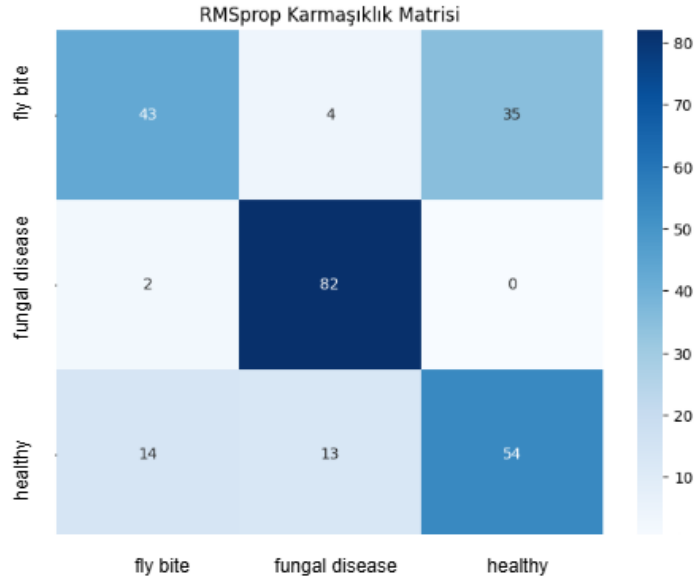
Şekil 3- 6: DEiT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri

Karmaşıklık matrisleri üzerinden yapılan incelemeler de sınıf bazlı performans farklarını ortaya koymaktadır. Şekil 3-7'te sunulan RAdam optimizasyon yöntemi ile eğitilmiş modelin karmaşıklık matrisi, modelin her üç sınıfı da yüksek doğruluk oranlarıyla ayırt edebildiğini göstermektedir. Sinek ısırıklı, mantar hastalıklı ve sağlıklı sınıflarında sırasıyla 82, 80 ve 81 örnek doğru sınıflandırılmıştır. Özellikle mantar hastalıklı sınıfta yalnızca 4 örnek yanlış sınıflanmış, bu da modelin sınıflar arası ayırma gücünü göstermektedir.



Şekil 3- 7: DEiT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

Buna karşılık Şekil 3-8’ deki RMSprop matrisinde ciddi bir sınıflandırma karışıklığı dikkat çekmektedir. Sinek ısırıklı sınıfının örneklerinin önemli bir kısmı sınıfa hatalı olarak atanmış, sağlıklı sınıfta da benzer bir dağılıma yaşanmıştır. Bu durum RMSprop ile eğitilen modelin sınıflar arası ayrımı başaramadığını göstermektedir.



Şekil 3- 8: DEiT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

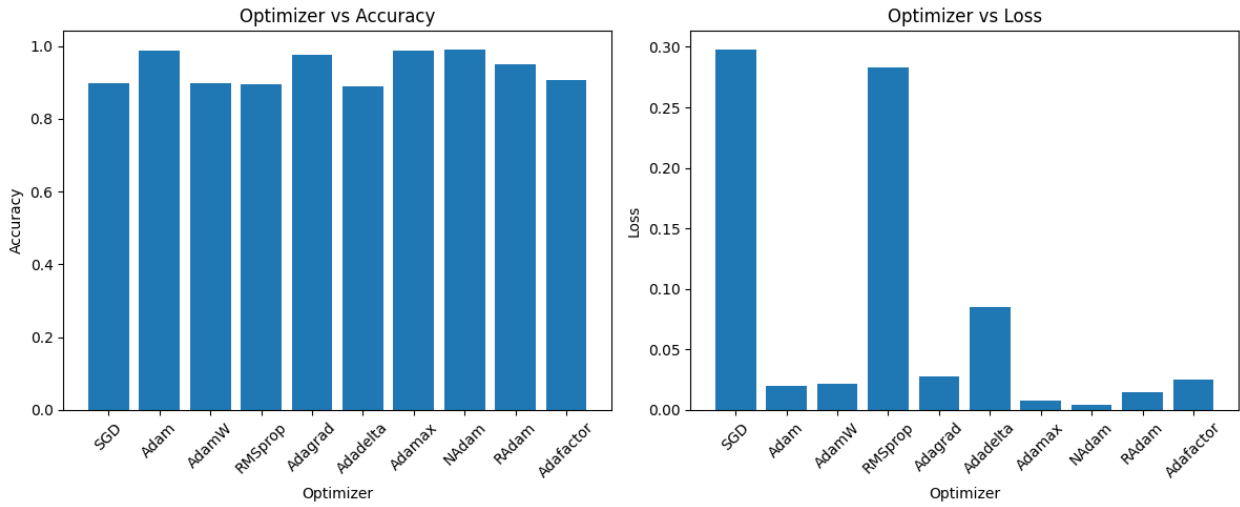
3.2.3 LeViT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi

LeViT modeli için uygulanan farklı optimizasyon algoritmaları, genel olarak yüksek sınıflandırma başarıları üretmiştir. Tüm algoritmalar %89’un üzerinde doğruluk sağlamış olmakla birlikte, aralarındaki görece başarı farklılıkları, özellikle sınıf ayrımı başarımı ve çıktı güven düzeyi açısından dikkat çekicidir. Tablo 3.5’te sunulan performans verilerine göre, NAdam algoritması, en düşük kayıp değeri (0.0041) ve en yüksek doğruluk oranı (%99,19) ile LeViT modeli için en başarılı sonuçları üretmiştir. Aynı zamanda 0.9919 makro F1 skoru ile sınıflar arası dengeli sınıflama kabiliyeti göstermiştir.

Tablo 3- 5: LeViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları

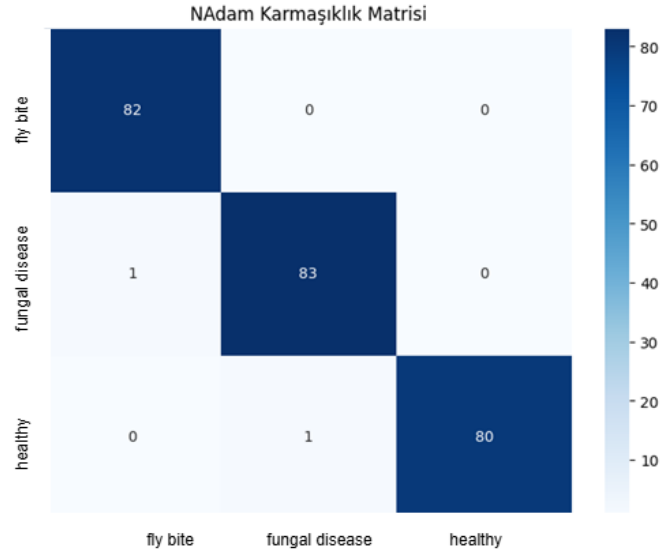
Optimizer	Best Loss	Accuracy	Macro F1-score	Macro Precision	Macro Recall
SGD	0.2983	0.8988	0.8995	0.9036	0.8990
Adam	0.0199	0.9879	0.9879	0.9882	0.9879
AdamW	0.0220	0.8988	0.9007	0.9173	0.8994
RMSprop	0.2833	0.8947	0.8949	0.9024	0.8957
Adagrad	0.0280	0.9757	0.9758	0.9767	0.9759
Adadelat	0.0852	0.8907	0.8869	0.9084	0.8928
Adamax	0.0081	0.9879	0.9880	0.9882	0.9881
NAdam	0.0041	0.9919	0.9919	0.9920	0.9919
RAdam	0.0149	0.9514	0.9514	0.9533	0.9519
Adafactor	0.0250	0.9069	0.9077	0.9257	0.9063

Bu farklılıkları daha açık biçimde ortaya koyan Şekil 3-9, optimizasyon algoritmalarına göre doğruluk ve kayıp değerlerinin grafiksel karşılaştırmasını sunmaktadır. Grafik, NAdam, Adamax ve Adam algoritmalarının yüksek doğruluk–düşük kayıpla öne çıktığını buna karşın SGD ve RMSprop gibi algoritmaların hem doğruluk hem de kayıp değerleri açısından görece olarak daha düşük performans sergilediğini ortaya koymaktadır.

**Şekil 3- 9 :** LeViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri

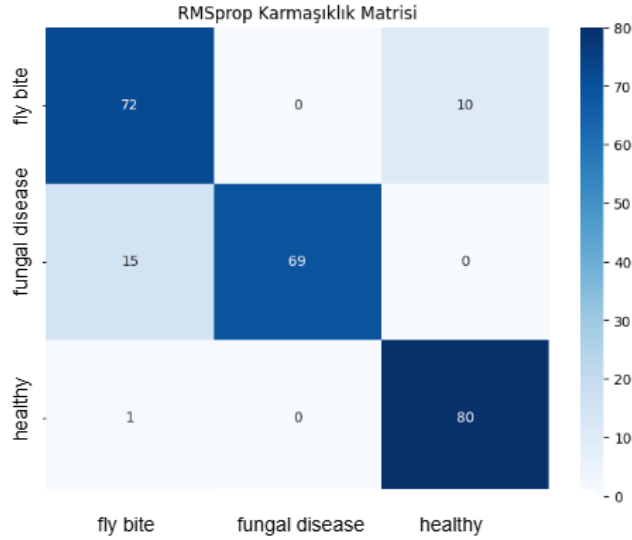
Sınıf bazlı başarımlar, karmaşıklık matrisleri üzerinden ayrıntılı şekilde değerlendirilmiştir. Şekil 3-10’da sunulan NAdam optimizasyon algoritmasıyla eğitilen LeViT modeline ait karmaşıklık matrisi, modelin üç sınıfı da yüksek doğruluk oranlarıyla ayırt edebildiğini göstermektedir. Sinek ısırıklı, mantar hastalıklı ve sağlıklı sınıflarına ait örneklerde yalnızca iki yanlış sınıflandırma tespit edilmiş; doğru sınıflandırılan örnek sayıları sırasıyla 82, 83 ve 80

olmuştur. Bu sonuçlar, LeViT mimarisinin NAdam algoritmasıyla yüksek düzeyde uyum içinde çalıştığını ve tüm sınıflarda %98'in üzerinde bir başarı sağladığını göstermektedir.



Şekil 3- 10: LeViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

Buna karşılık, Şekil 3-11’de yer alan RMSprop algoritmasına ait karmaşıklık matrisi, görece olarak daha fazla sınıf karışıklığına işaret etmektedir. Özellikle fungal disease sınıfına ait 15 örnek fly bite sınıfına; fly bite sınıfına ait 10 örnek ise hatalı şekilde healthy sınıfına atanmıştır. Bu tür hata dağılımı, RMSprop ile eğitilen modelin bazı sınıflar arasındaki ayrımı yapmakta zorlandığını ve karar sınırlarının net şekilde çizilemediğini düşündürmektedir. Yine de modelin genel doğruluk oranının %89’un üzerinde kalması, temel sınıflandırma kabiliyetinin korunduğunu göstermektedir.



Şekil 3- 11: LeViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

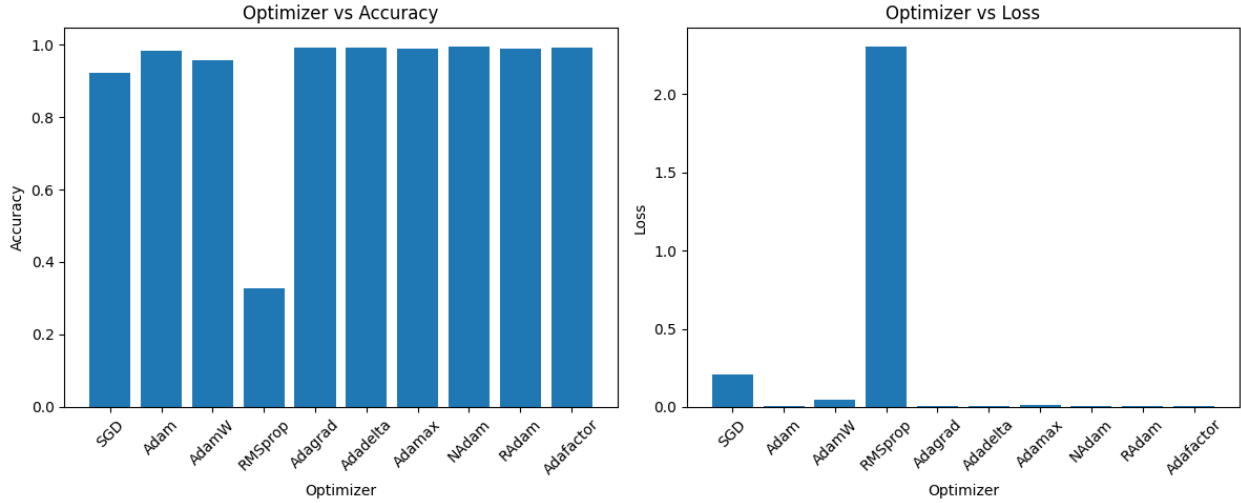
3.2.4 MaxViT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi

MaxViT (Multi-Axis Vision Transformer) modeli üzerinde uygulanan çeşitli optimizasyon algoritmaları, genel olarak yüksek düzeyde sınıflandırma başarısı sağlamış ve modelin güçlü yapısal kapasitesini desteklemiştir. Tablo 3.4'te sunulan karşılaştırmalı sonuçlara göre, en başarılı performans NAdam algoritmasıyla elde edilmiştir. NAdam, 0.0092 gibi oldukça düşük bir kayıp değeriyle birlikte %99,60 doğruluk oranına ulaşarak tüm metriklerde en iyi sonucu vermiştir. Bu algoritmayı sırasıyla Adagrad, Adadelata, Adafactor, Adamax ve RAdam izlemiştir; bu algoritmalar da %98 ile %99 arasında değişen doğruluk oranlarıyla istikrarlı ve güçlü bir performans sergilemiştir.

Tablo 3- 6: MaxViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları

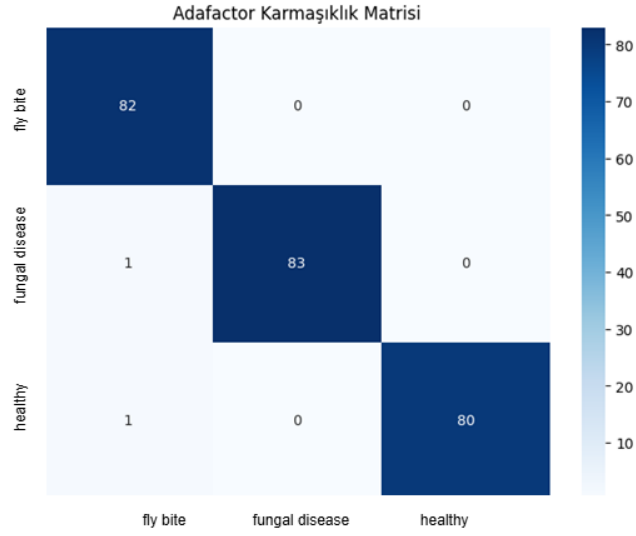
Optimizer	Best Loss	Accuracy	Macro F1-score	Macro Precision	Macro Recall
SGD	0.2107	0.9231	0.9244	0.9327	0.9232
Adam	0.0094	0.9838	0.9839	0.9845	0.9838
AdamW	0.0471	0.9555	0.9550	0.9581	0.9563
RMSprop	2.3083	0.3279	0.1646	0.1093	0.3333
Adagrad	0.0071	0.9919	0.9919	0.9921	0.9919
Adadelata	0.0078	0.9919	0.9919	0.9921	0.9919
Adamax	0.0128	0.9879	0.9879	0.9882	0.9879
NAdam	0.0092	0.9960	0.9960	0.9960	0.9960
RAdam	0.0058	0.9879	0.9879	0.9882	0.9879
Adafactor	0.0048	0.9919	0.9919	0.9921	0.9919

Bu farklılıkların daha net anlaşılması için oluşturulan Şekil 3.2, optimizier'ların doğruluk ve kayıp değerlerini karşılaştırmalı olarak sunmaktadır. Grafik, NAdam ve Adagrad gibi adaptif algoritmaların hem yüksek doğruluk hem de düşük kayıp değerleriyle öne çıktığını buna karşılık RMSprop algoritmasının tüm metriklerde açık biçimde ayrıştığını göstermektedir. RMSprop'un doğruluk oranı %32,79'da kalırken, kayıp değeri 2.3083 ile en yüksektir. Bu uçurumsal fark, RMSprop'un MaxViT mimarisiyle neredeyse hiç öğrenme sağlayamadığını göstermektedir.



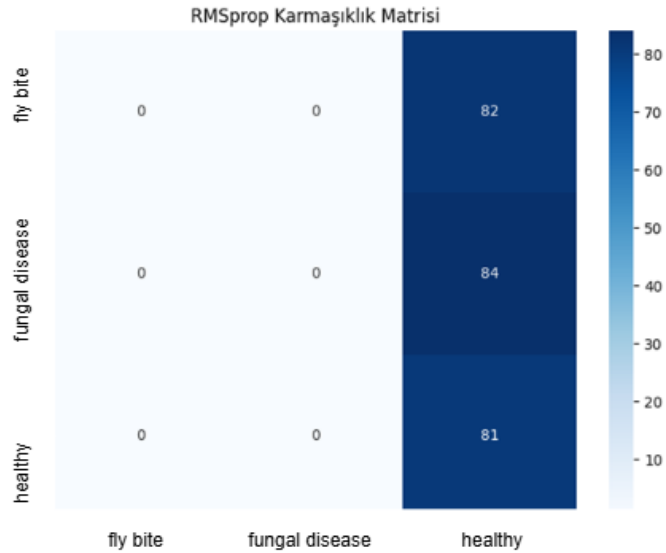
Şekil 3- 12: MaxViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri

Şekil 3-13'te gösterilen Adafactor algoritmasıyla eğitilen MaxViT modeline ait karmaşıklık matrisi, sınıf bazlı doğrulukların dengeli ve yüksek seviyede olduğunu ortaya koymaktadır. Sinek ısırıklı, mantar hastalıklı ve sağlıklı sınıflarında sırasıyla 82, 83 ve 80 örnek doğru şekilde sınıflandırılmıştır. Yalnızca iki örnekte sınıf karışıklığı gözlenmiş, bu da modelin genel başarımının oldukça yüksek olduğunu göstermektedir. Bu sonuçlar, Adafactor algoritmasının düşük bellek gereksinimine karşın güçlü bir sınıflandırma kapasitesine sahip olduğunu kanıtlamaktadır. Özellikle mobil cihazlar veya kaynak kısıtlı sistemlerde, Adafactor gibi optimizasyon algoritmaları, MaxViT gibi karmaşık derin öğrenme mimarileri için etkili bir alternatif sunmaktadır.



Şekil 3- 13: MaxViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

Buna karşılık, Şekil 3.14’te sunulan RMSprop algoritmasına ait karmaşıklık matrisi, modelin ciddi derecede dengesiz sınıflandırma gerçekleştirdiğini göstermektedir. Tüm test verisi yalnızca sağlıklı sınıfına atanmış; sinek ısırıklı ve mantar hastalıklı sınıflarına ait örneklerin hiçbiri doğru sınıflandırılmamıştır. Bu durum modelin eğitim sürecinde sınıflar arası ayırım yapma yeteneğini kaybettiğini ve ciddi bir öğrenme bozukluğu yaşadığını göstermektedir. RMSprop algoritması ile elde edilen bu sonuçlar, MaxViT mimarisi açısından uyumsuz ve yetersiz bir öğrenme davranışına işaret etmektedir.



Şekil 3- 14: MaxViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

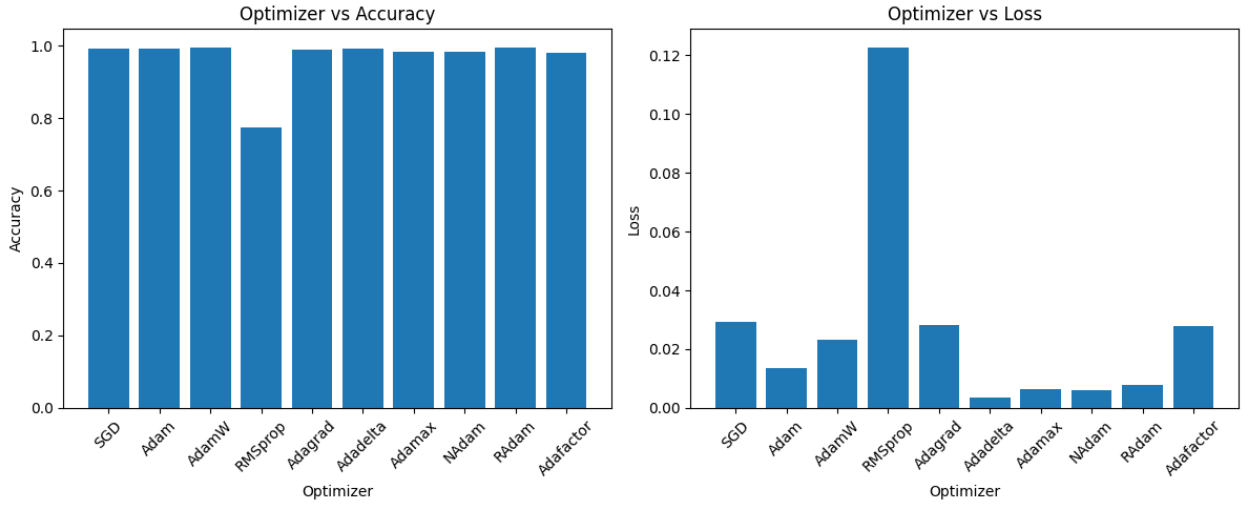
3.2.5 MobileViT: Optimizasyon Algoritmalarına Göre Performans Analizi

MobileViT modeli üzerinde gerçekleştirilen optimizasyon algoritması karşılaştırmaları, modelin genel olarak oldukça yüksek doğruluk seviyelerine ulaşabildiğini göstermiştir. Tablo 3-7’de yer alan sonuçlara göre, en başarılı performans AdamW ve RAdam algoritmaları tarafından sağlanmıştır. Bu iki algoritma da %99,60 doğruluk, 0.9960 makro F1 skoru ve dengeli precision–recall değerleriyle öne çıkmıştır. Takip eden algoritmalar (Adadelata, Adam, SGD, Adagrad, NAdam) da %98–99 doğruluk aralığında başarılı sonuçlar vermiştir. Bu durum, MobileViT’in farklı güncelleyicilere karşı toleranslı ve kararlı bir yapıya sahip olduğunu göstermektedir.

Tablo 3- 7: MobileViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları

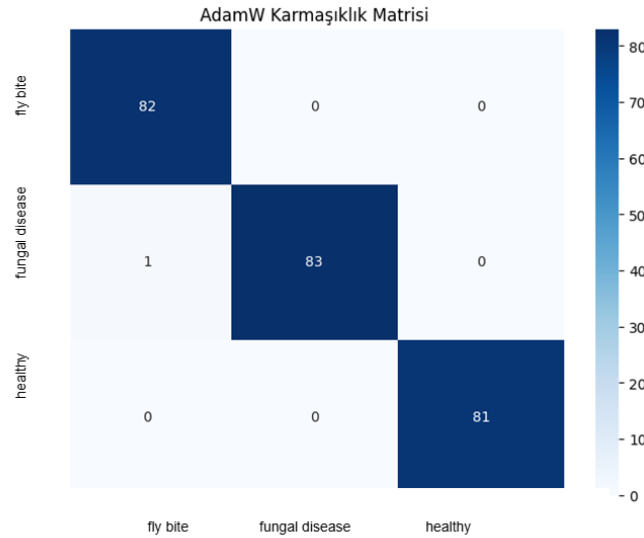
Optimizer	Best Loss	Accuracy	Macro F1-score	Macro Precision	Macro Recall
SGD	0.0294	0.9919	0.9920	0.9921	0.9921
Adam	0.0136	0.9919	0.9919	0.9921	0.9919
AdamW	0.0233	0.9960	0.9960	0.9960	0.9960
RMSprop	0.1228	0.7733	0.7599	0.8180	0.7765
Adagrad	0.0283	0.9879	0.9879	0.9882	0.9879
Adadelata	0.0034	0.9919	0.9919	0.9919	0.9921
Adamax	0.0066	0.9838	0.9839	0.9841	0.9839
NAdam	0.0059	0.9838	0.9838	0.9838	0.9839
RAdam	0.0078	0.9960	0.9960	0.9960	0.9960
Adafactor	0.0279	0.9798	0.9798	0.9798	0.9798

Optimizasyon algoritmalarının başarı düzeyleri, Şekil 3-15’te sunulan doğruluk ve kayıp grafiklerinde daha açık biçimde görselleştirilmiştir. Neredeyse tüm algoritmalar doğrulukta %98 üzerindeyken, RMSprop açık şekilde ayrılmış ve yalnızca %77,33 doğruluk ile görece olarak en düşük başarıyı göstermiştir. Aynı zamanda kayıp değeri (0.1228) ile ikinci en yüksek değeri alarak modelin öğrenme istikrarını bozduğu izlenimini vermektedir.



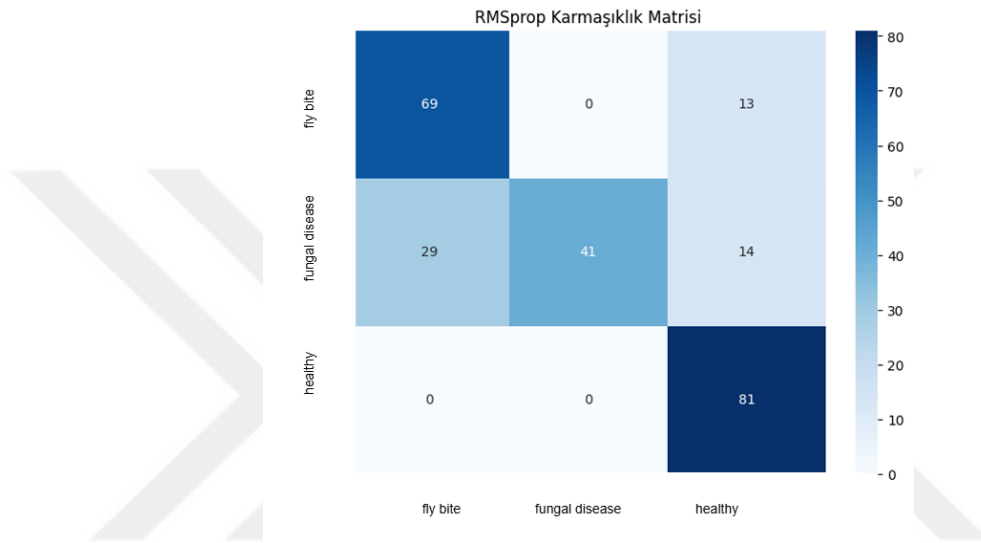
Şekil 3- 15: MobileViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri

Şekil 3-16'da gösterilen AdamW algoritmasıyla eğitilen MobileViT modeline ait karmaşıklık matrisi, modelin sınıflar arasında oldukça yüksek doğrulukla sınıflandırma yaptığını göstermektedir. Sinek ısırıklı, mantar hastalıklı ve sağlıklı sınıflarındaki doğru sınıflandırma sayıları sırasıyla 82, 83 ve 81 olup, yalnızca iki örnek yanlış sınıflandırılmıştır. Bu yüksek başarı düzeyi, AdamW algoritmasının MobileViT mimarisiyle son derece uyumlu çalıştığını ve sınıf ayırımında etkili olduğunu göstermektedir.



Şekil 3- 16: MobileViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

Buna karşılık, Şekil 3.17’de sunulan RMSprop algoritmasına ait karmaşıklık matrisi, sınıflar arası ayırım performansının belirgin şekilde düştüğünü ortaya koymaktadır. Fungal disease sınıfına ait örneklerin 29’u sinek ısırıklı, 14’ü sağlıklı sınıfa yanlış atanmış; ayrıca sinek ısırıklı sınıfa ait 13 örnek de hatalı olarak sağlıklı sınıfa sınıflandırılmıştır. Bu dağılım, modelin özellikle mantar hastalıklı sınıfını ayırt etmede zorlandığını ve RMSprop algoritmasının sınıf sınırlarını net biçimde öğrenemediğini göstermektedir. Sadece healthy sınıftaki 81 örnek doğru sınıflandırılmış ve diğer iki sınıfta ciddi hata oranları oluşmuştur.



Şekil 3- 17: MobileViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

3.2.6 ViT Modeli: Optimizasyon Algoritmalarına Göre Performans Analizi

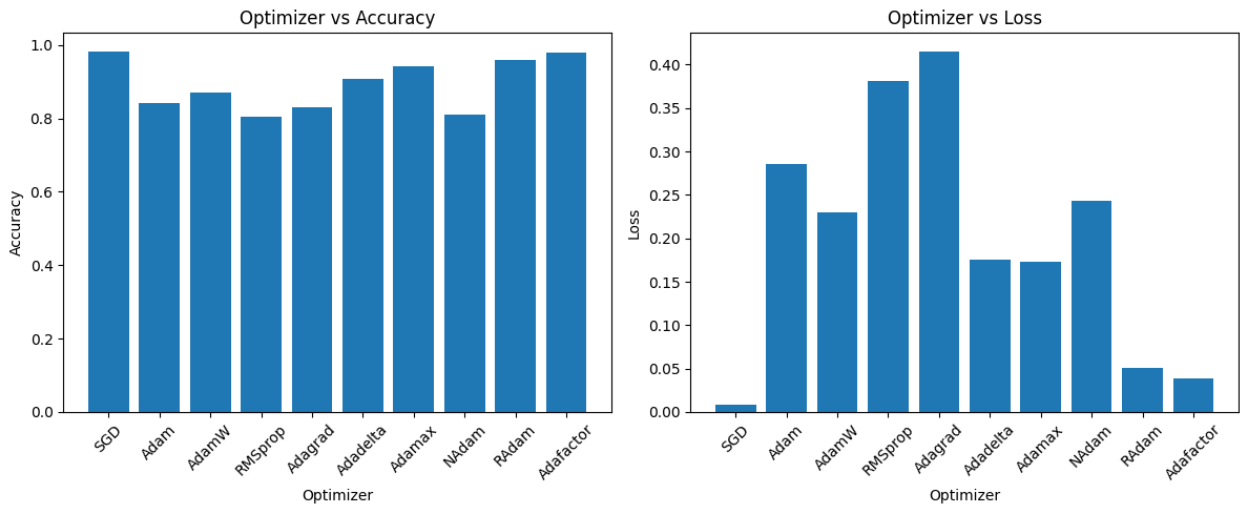
ViT modeli üzerinde uygulanan 10 farklı optimizasyon algoritmasının karşılaştırmalı değerlendirmesi, model başarımının optimizier seçimine yüksek derecede duyarlı olduğunu göstermektedir. Tablo 3-8’de sunulan sonuçlara göre, Adafactor algoritması, tüm metriklerde en başarılı sonucu vermiştir. En yüksek doğruluk oranı (%97,98), en yüksek makro F1 skoru (0.9799) ve en düşük kayıp değeri (0.0388) ile ViT modelinde en etkili optimizasyonu sağlamıştır. Özellikle düşük bellek tüketimiyle bilinen Adafactor’un bu yüksek başarımı, modelin parametre güncellemelerinde verimli bilgi akışı sunduğunu göstermektedir.

Tablo 3- 8: ViT Modeli İçin Farklı Optimizasyon Algoritmalarına Ait Sınıflandırma Performans Sonuçları

Optimizer	Best Loss	Accuracy	Macro F1-score	Macro Precision	Macro Recall
SGD	0.0087	0.9838	0.9839	0.9841	0.9839
Adam	0.2859	0.8421	0.8397	0.8421	0.8408
AdamW	0.2298	0.8704	0.8697	0.8700	0.8696
RMSprop	0.3815	0.8057	0.8064	0.8125	0.8041
Adagrad	0.4157	0.8300	0.8308	0.8336	0.8288
Adadelata	0.1749	0.9069	0.9077	0.9238	0.9063
Adamax	0.1729	0.9433	0.9434	0.9442	0.9433
NAdam	0.2428	0.8097	0.8085	0.8095	0.8084
RAdam	0.0507	0.9595	0.9597	0.9615	0.9595
Adafactor	0.0388	0.9798	0.9799	0.9808	0.9800

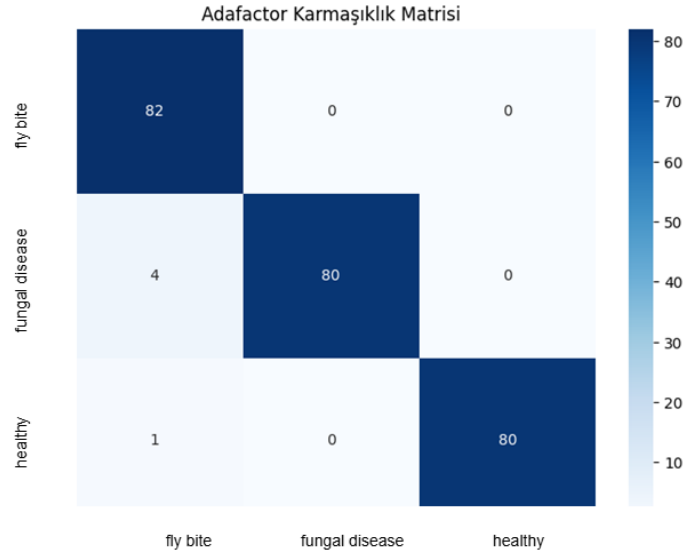
Adafactor'un ardından gelen RAdam (%95,95 doğruluk) ve Adamax (%94,33) algoritmaları da oldukça yüksek başarı göstermiştir. Bu algoritmalar hem doğruluk hem de precision–recall dengesinde istikrarlı sonuçlar üretmiş ve modelin sınıflar arası ayırım başarısını korumasına katkı sağlamıştır.

Bu karşılaştırmalar Şekil 3-18'de sunulan doğruluk ve kayıp grafiklerinde daha net biçimde görselleştirilmiştir. Grafikler, Adafactor'un hem doğruluk hem de loss bakımından diğer algoritmalarla açık farkla ayrıştığını ortaya koymaktadır. Buna karşılık RMSprop, Adam ve NAdam gibi algoritmalar daha yüksek kayıp ve daha düşük doğruluk değerleriyle görece olarak düşük performans göstermiştir.

**Şekil 3- 18:** ViT Modelinde Optimizasyon Algoritmalarına Göre Doğruluk (sol) ve Kayıp (sağ) Değerleri

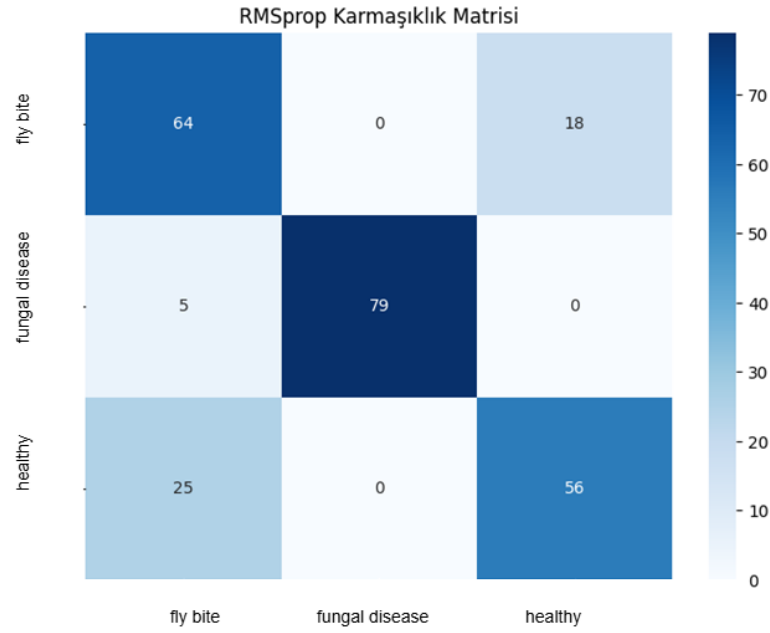
Şekil 3.19'da sunulan Adafactor algoritmasına ait karmaşıklık matrisi, ViT modelinin sınıflar arasında yüksek doğrulukla tahminler gerçekleştirdiğini göstermektedir. Fly bite, fungal

disease ve healthy sınıflarına ait doğru sınıflandırma sayıları sırasıyla 82, 80 ve 80'dir. Yanlış sınıflandırmalar oldukça sınırlıdır; fungal disease sınıfındaki 4 örnek fly bite olarak, healthy sınıfındaki 1 örnek ise fly bite olarak hatalı tahmin edilmiştir. Bu durum, Adafactor algoritmasının yalnızca genel doğruluk değil, sınıf içi dengeli ayırım başarımı açısından da etkili olduğunu göstermektedir.



Şekil 3- 19: ViT Modelinde En İyi Sonucu Veren Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

Buna karşın, Şekil 3.20’de yer alan RMSprop algoritmasına ait karmaşıklık matrisi, ciddi sınıf karışıklıkları barındırmaktadır. Healthy sınıfına ait 25 örnek fly bite sınıfına yanlış atanmış, fly bite sınıfına ait 18 örnek de hatalı şekilde healthy sınıfına sınıflandırılmıştır. Ayrıca fungal disease sınıfında da 5 örnek fly bite olarak yanlış tahmin edilmiştir. Bu hata dağılımı, RMSprop algoritmasıyla eğitilen ViT modelinin, özellikle fly bite ile healthy sınıfları arasındaki görsel benzerlikleri ayırt etmekte zorlandığını ve karar sınırlarını netleştiremediğini göstermektedir.



Şekil 3- 20: ViT Modelinde En Düşük Sınıflandırma Başarımı Sağlayan Optimizasyon Yöntemine Ait Karmaşıklık Matrisi

3.2.7 Genel Değerlendirme ve Sonuç

Bu bölümde gerçekleştirilen deneysel analizler, zeytin meyvesine ait üç sınıflı görüntü verisi üzerinde kullanılan farklı Transformer tabanlı mimarilerin ve çeşitli optimizasyon algoritmalarının sınıflandırma performansına etkisini kapsamlı bir şekilde ortaya koymuştur. Bulgular, model mimarisi ve optimizasyon algoritması seçiminin yalnızca genel doğruluk düzeyini değil aynı zamanda sınıf içi dengeli ayırım gücü ve öğrenme istikrarını da önemli ölçüde etkilediğini göstermektedir.

Deneysel sonuçlara göre NAdam, RAdam, Adafactor ve AdamW gibi optimizasyon algoritmaları genellikle daha yüksek doğruluk, daha düşük kayıp ve daha dengeli makro F1 skorları ile öne çıkmıştır. Özellikle NAdam algoritması, MaxViT ve LeViT modelleri üzerinde yüksek başarımlı sonuçlar üretmiş; Adafactor ise ViT modelinde hem genel doğruluk hem de sınıf içi denge açısından en başarılı performansı sergilemiştir. AdamW ve RAdam ise MobileViT ve BEiT gibi modellerde öne çıkan diğer optimizasyon algoritmaları olmuştur. Bu bulgular, transformer mimarilerinde ağırlık güncellemelerinde istikrarlı ve hızlı uyum sağlayan algoritmaların tercih edilmesinin model başarısını anlamlı derecede artırabildiğine işaret etmektedir.

Buna karşın RMSprop algoritmasının hemen her modelde belirgin biçimde başarısız olduğu hem genel doğruluk hem de sınıf içi ayırt edicilikte ciddi dengesizlikler ve yüksek kayıp değerleri ürettiği görülmüştür. Özellikle RMSprop ile eğitilen modellerde sınıflar arasında ciddi karışıklıklar yaşanmış, kimi modellerde ise sınıf örnekleri neredeyse tamamen tek bir sınıfa yönlendirilmiştir. Benzer şekilde klasik SGD ve bazı momentum tabanlı optimizierların, özellikle daha karmaşık veya derin mimarilerde, adaptif algoritmalarla kıyasla daha düşük başarı gösterdiği gözlemlenmiştir.

Her bir model-optimizer kombinasyonunda dikkat çeken bir diğer unsur ise yüksek genel doğruluk değerlerinin bazen sınıf içi dengesizlikleri maskeleyebilmesidir. Bu nedenle yalnızca doğruluk değil, makro F1 skoru, precision ve recall gibi sınıf bazlı ölçütlerin de dikkate alınması, modellerin pratik kullanım potansiyelini daha gerçekçi biçimde değerlendirmeye olanak sağlamaktadır.

Genel olarak transformer tabanlı mimarilerde en yüksek ve en dengeli başarılar, modern adaptif optimizasyon algoritmalarıyla elde edilmiştir. Model başarısı, yalnızca mimari karmaşıklık veya parametre sayısı ile değil uygun optimizer seçimiyle birlikte eğitim dinamiklerinin uyumuna da bağlıdır. Bu sonuçlar özellikle küçük ve dengesiz veri setlerinde, optimizer seçiminin model başarımı üzerinde kritik rol oynadığını; yüksek doğruluk için yalnızca gelişmiş mimarilerin değil uygun öğrenme stratejilerinin de tercih edilmesi gerektiğini göstermektedir.

Sonuç olarak çalışmada elde edilen bulgular, derin öğrenme tabanlı görüntü sınıflandırma problemlerinde, model mimarisi kadar optimizasyon algoritması seçiminin de titizlikle ele alınmasının, sınıflar arası ayırım gücünü ve genel başarı düzeyini artırmada belirleyici olduğunu ortaya koymuştur. Bu nedenle gelecekte yapılacak benzer çalışmalarda sadece model seçimi değil hiperparametre ve optimizer ayarlarının da deneysel olarak optimize edilmesi önerilmektedir.

3.3 MetaFormer Tabanlı Modellerin Karşılaştırmalı Performans Analizi

Bu bölümde MetaFormer mimarisine dayalı üç farklı model olan CAFormer-S18, ConvFormer-S18 ve PoolFormer-V2-S12 yapılarına ait sınıflandırma performansları karşılaştırmalı olarak analiz edilmiştir. Modeller, timm kütüphanesi aracılığıyla önceden eğitilmiş ağırlıklarla yüklenmiş ve çalışma kapsamında belirlenen veri kümesi üzerinde yeniden eğitilmiştir. Eğitim sürecinde erken durdurma stratejisi uygulanmış ve en iyi doğruluk değerinin elde edildiği

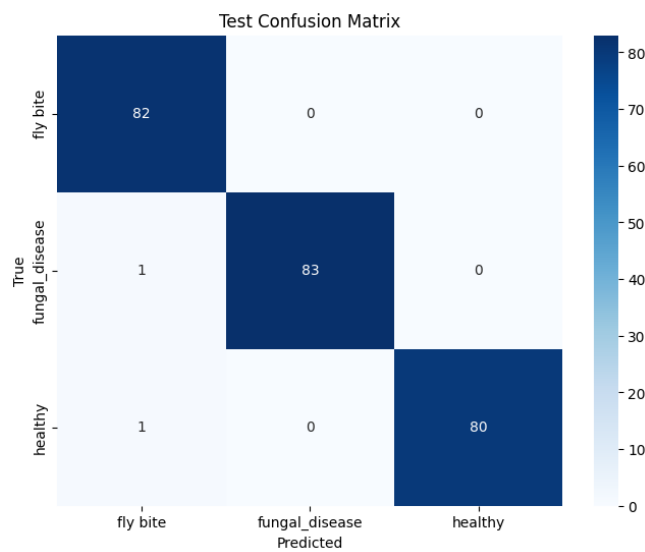
epoch'ta model değerlendirmeye alınmıştır. Elde edilen performans sonuçları doğruluk, kayıp (loss), precision, recall, F1 skoru ve karmaşıklık matrisleri üzerinden yorumlanmıştır.

Tablo 3-9'da sunulan performans sonuçlarına göre, CAFormer-S18 ve PoolFormer-V2-S12 modelleri test verisinde %99.19 doğruluk oranı ile en yüksek başarıyı göstermiştir. Precision, recall ve F1 değerleri açısından da bu iki modelin sonuçları eşit düzeyde yüksek bulunmuştur. Buna karşılık, ConvFormer-S18 modeli %98.38 doğruluk oranı ile daha düşük bir performans sergilemiştir. Aynı zamanda hatalı sınıflandırma sayısı da bu modelde diğerlerine göre daha fazladır. Kayıp değeri açısından değerlendirildiğinde, PoolFormer-V2-S12 en düşük test kaybını göstererek öğrenme sürecinde daha kararlı bir model profili çizmiştir.

Tablo 3- 9: MetaFormer Tabanlı Modellerin Performans Sonuçları

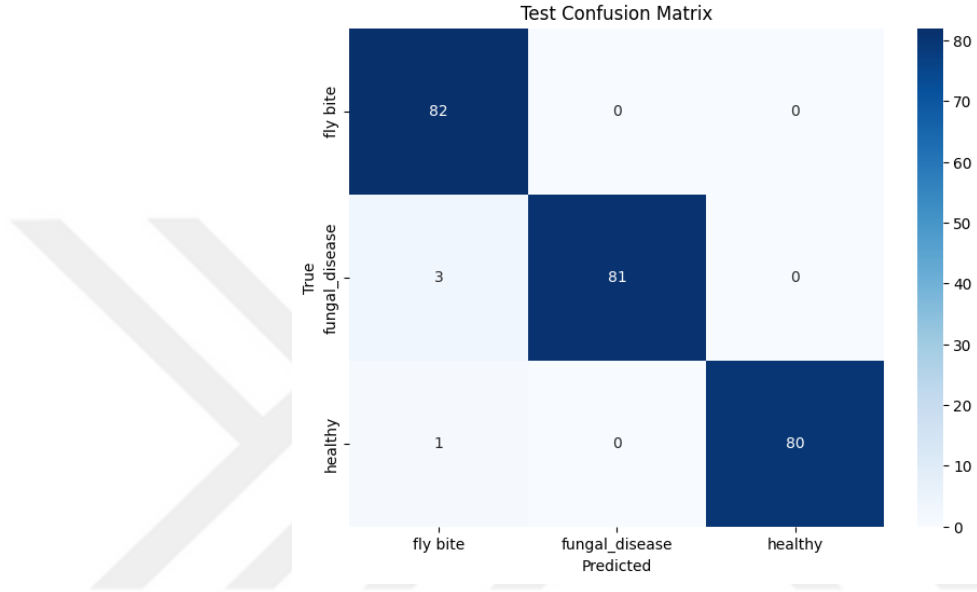
Model	Doğruluk (%)	Kayıp	Precision	Recall	F1-Score
CAFormer-S18	99.19	0.0475	0.9921	0.9919	0.9919
ConvFormer-S18	98.38	0.0548	0.9845	0.9840	0.9839
PoolFormer-V2-S12	99.19	0.0297	0.9921	0.9919	0.9919

Şekil 3-21'de sunulan CAFormer-S18 modeline ait karmaşıklık matrisi, modelin üç sınıfta da yüksek doğrulukla tahmin gerçekleştirdiğini göstermektedir. Sinek ısırıklı sınıfta tüm örnekler doğru sınıflandırılmıştır. Mantar hastalıklı ve sağlıklı sınıflarında ise yalnızca birer örnek yanlış tahmin edilmiştir. Bu sonuçlar, modelin sınıflar arası ayrımı güçlü şekilde gerçekleştirebildiğini göstermektedir.



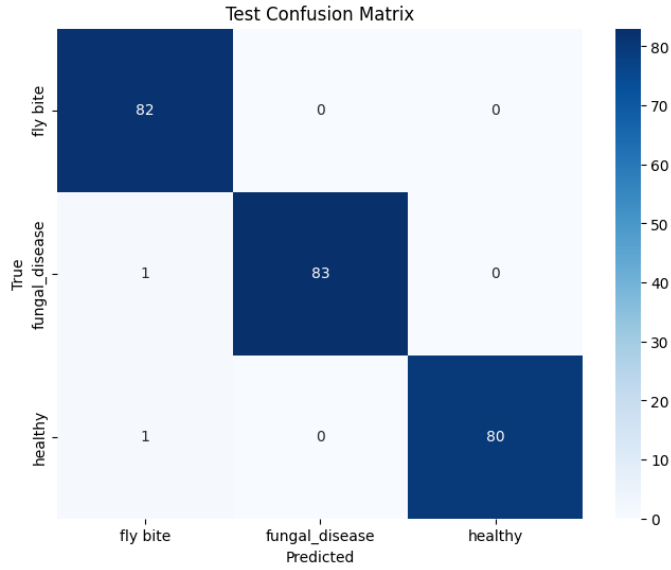
Şekil 3- 21: CAFormer-S18 Modeli Karmaşıklık Matrisi

Şekil 3.22’de sunulan ConvFormer-S18 modeline ait karmaşıklık matrisi incelendiğinde, modelin genel olarak başarılı bir sınıflandırma yaptığı görülmekle birlikte, özellikle fungal disease sınıfında 3 örneğin ve healthy sınıfında 1 örneğin hatalı olarak fly bite sınıfına atandığı tespit edilmiştir. Bu durum, modelin bazı sınıflar arasındaki benzerlikleri ayırtmakta zaman zaman zorlandığını göstermektedir.



Şekil 3- 22: ConvFormer-S18 Modeli Karmaşıklık Matrisi

Şekil 3.23’te gösterilen PoolFormer-V2-S12 modeline ait karmaşıklık matrisi, modelin her üç sınıfta da oldukça dengeli ve başarılı bir sınıflandırma gerçekleştirdiğini ortaya koymaktadır. Sinek ısırıklı sınıfında %100 doğruluk elde edilmiş, diğer iki sınıfta ise sadece birer örnek yanlış tahmin edilmiştir. Özellikle düşük test kaybı ile birlikte değerlendirildiğinde, PoolFormer-V2-S12 modeli istikrarlı bir öğrenme başarımını sergilemiş ve sınıflar arasında etkili bir ayırım kabiliyeti sunmuştur.



Şekil 3- 23: PoolFormer-V2-S12 Modeli Karmaşıklık Matrisi

3.4 MetaFormer Tabanlı Özgün Hibrit Blok Kombinasyonlarının Sınıflandırma Performansları

Bu bölümde sabit Metaformer iskeleti üzerinde farklı token mixer bloklarının sıra ve tür bakımından yeniden düzenlenmesiyle oluşturulan altı özgün hibrit modelin sınıflandırma performansları karşılaştırmalı olarak analiz edilmiştir. Tüm modeller, aynı hiperparametre ayarları ve eğitim protokolüyle, üç sınıflı zeytin hastalığı veri seti üzerinde değerlendirilmiştir.

Elde edilen bulgular, Tablo 3-10'da verilmiştir. Bu tablo her bir hibrit modelin doğruluk, kayıp, precision, recall ve F1 skoru gibi temel performans metriklerindeki başarı düzeylerini ayrıntılı şekilde sunmaktadır. Özellikle Model 1, Model 2 ve Model 4; %97'nin üzerinde doğruluk ve makro F1 skorlarıyla öne çıkmıştır. Bu modellerde yerel örüntü çıkarımı, rastlantısallık, kimlik aktarımı ve küresel bağlam modelleme gibi farklı bilgi işleme stratejileri ardışık olarak entegre edilmiştir. Model 5 ve Model 6 da yüksek doğruluk oranlarıyla dikkat çekmiş ancak özellikle Model 3, diğerlerinden belirgin biçimde ayrılarak %84 doğruluk ve daha düşük F1 skoru ile en zayıf performansı göstermiştir.

Tablo 3- 10: MetaFormer Tabanlı Özgün Hibrit Modellerin Performans Sonuçları

Model	Doğruluk (%)	Test Loss	Precision	Recall	F1-Score
Model 1	97.98	0.0620	0.9800	0.9800	0.9797
Model 2	97.98	0.0949	0.9800	0.9799	0.9798
Model 3	84.21	0.4552	0.8451	0.8412	0.8427
Model 4	97.98	0.0842	0.9800	0.9799	0.9798
Model 5	97.57	0.0813	0.9759	0.9759	0.9757
Model 6	97.17	0.0841	0.9716	0.9717	0.9717

Model 3'ün daha düşük başarısı, blok diziliminin bilgi akışı ve örüntü çıkarımında yetersiz kaldığını, rastgele veya uygun olmayan blok kombinasyonlarının öğrenme kapasitesini sınırlayabileceğini göstermektedir. Buna karşılık yüksek performans sunan modellerde farklı blok türlerinin ardışık ve uyumlu şekilde yerleştirilmesinin, modelin genelleme yeteneği ve sınıf ayırma gücünü anlamlı şekilde artırdığı gözlenmiştir.

Bir diğer dikkat çekici bulgu ise önerilen hibrit MetaFormer modellerinin, herhangi bir büyük ölçekli veri kümesinde (örneğin ImageNet) önceden eğitilmemiş olmalarına rağmen yüksek doğruluk ve sınıf ayırma başarısı elde edebilmesidir. Bu durum, modüler ve görev-uyarlanabilir blok dizilimine dayalı mimarilerin, yalnızca transfer öğrenmeye veya büyük veri setlerine bağımlı olmadan da etkili çözümler sunabileceğini göstermektedir.

Bu sonuçlar ileride gerçekleştirilecek çalışmalarda, az veriyle öğrenebilen, probleme özgü otomatik blok seçimi (AutoBlock Selection) gibi adaptif sistemlerin geliştirilmesi için önemli bir kuramsal ve pratik referans sunmaktadır. Modüler ve görev uyarlanabilir mimarilerin, sınırlı veriyle çalışan ya da dinamik olarak farklı görevleri çözmesi gereken yapay zeka sistemlerinde uygulanabilirliği oldukça yüksektir.

Sonuç olarak bu bölümde analiz edilen özgün hibrit blok kombinasyonları yalnızca klasik, sabit blok dizilimli modellere alternatif oluşturmakla kalmamış, aynı zamanda görev-adaptif esnek mimarilerin model başarısı üzerindeki olumlu etkisini de ortaya koymuştur. Tabloda özetlenen yüksek doğruluk, F1 skoru ve diğer metrikler, geliştirilen hibrit yapıların hem açıklanabilirlik hem de pratik uygulanabilirlik açısından önemli avantajlar sunduğunu göstermektedir.

3.5 Önerilen Özgün DeformationScoreFormer Mimarisi ile Zeytin Görüntülerinin Sınıflandırılması: Performans Analizi ve Değerlendirme

Bu bölümde, MetaFormer iskeletine dayalı olarak geliştirilen ve özgün bir token mixer bileşeni içeren DeformationScoreFormer mimarisinin sınıflandırma başarımı detaylı biçimde değerlendirilmektedir. Modelin üç sınıflı zeytin görüntüleri veri seti üzerindeki doğruluk düzeyi, kayıp değeri, sınıf bazlı başarı ölçütleri ve hata analizleri üzerinden kapsamlı bir inceleme sunulmuştur. Ayrıca, modelin karar mekanizmasının açıklanabilirliğini artırmak amacıyla GRAD-CAM gibi görsel dikkat haritalarına da yer verilmiştir.

3.5.1 Sınıflandırma Performansı: Doğruluk, Kayıp ve F1-Skoru

Önerilen DeformationScoreFormer mimarisinin sınıflandırma başarımı, doğruluk, kayıp ve makro ortalama sınıflandırma metrikleri üzerinden değerlendirilmiştir. Tablo 3-11’de sunulan test sonuçları, modelin genel doğruluk oranının %97,98 olduğunu ve test kümesi üzerinde 0.0992 düzeyinde bir kayıp değeri ile yüksek doğruluk sağladığını ortaya koymaktadır. Ayrıca precision, recall ve F1-score değerlerinin tamamının 0.98 düzeyine yakın gerçekleşmesi, modelin tüm sınıflarda dengeli ve tutarlı bir performans sergilediğini göstermektedir. Bu sonuçlar, modelin her sınıf için dengeli bir ayırt ediciliğe sahip olduğunu göstermektedir.

Tablo 3- 11: Önerilen metaformer modelinin performans sonucu

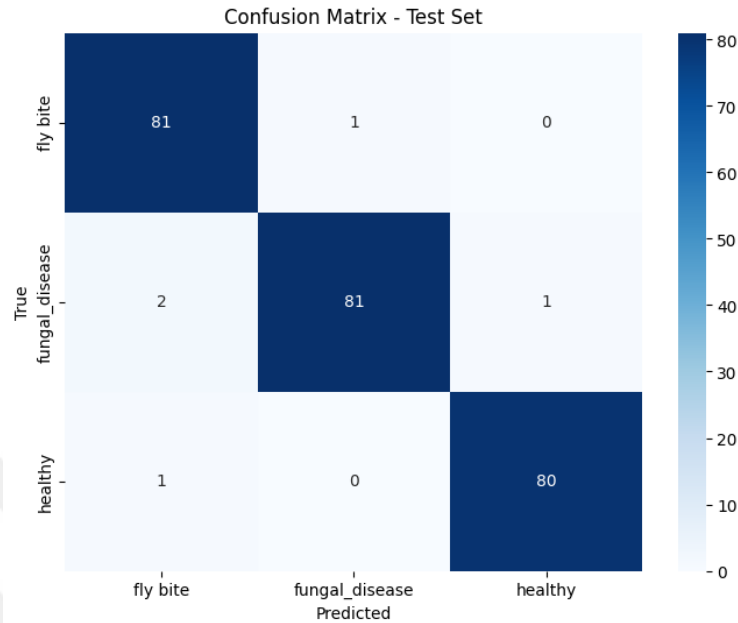
Model	Accuracy (%)	Loss	Precision	Recall	F1-Score
DeformationScoreFormer	97.98	0.0992	0.9799	0.9799	0.9798

Modelin sınıflandırma başarısı, karışıklık matrisi üzerinden de incelenmiştir. Şekil 3-24’te sunulan karışıklık matrisi, her sınıfa ait doğru ve yanlış sınıflandırmaları açık biçimde görselleştirmektedir. Buna göre:

- Sinek ısırıklı sınıfında 81 doğru tahmin, yalnızca 1 yanlış tahmin yapılmıştır. Mantar hastalığı sınıfında 81 örnek doğru sınıflandırılmış, 3 örnek yanlış etiketlenmiştir
- Sağlıklı sınıfında ise 80 örnek doğru tanımlanmış, sadece 1 örnek “fly bite” olarak hatalı sınıflandırılmıştır.

Bu veriler, modelin özellikle “sağlıklı” ve “sinek ısırıklı” sınıflarını yüksek doğrulukla ayırt edebildiğini, “mantar hastalıklı” sınıfında ise çok sınırlı sayıda karışıklık olduğunu göstermektedir. Bunun sebebinin ise zeytin meyvesinin ilk olarak sinek ısırığına maruz kalıp

ardından da çeşitli fugal etmenlere maruz kalarak mantar hastalığına sahip zeytin meyvesi olması sonucunda ortaya çıkan görüntüsü ile ilgili olduğu düşünülmektedir.



Şekil 3- 24: DeformationScoreFormer modeli için test veri kümesine ait karışıklık matrisi.

3.5.2 DeformationScore Haritaları ile Dikkat Bölgelerinin Yorumlanması

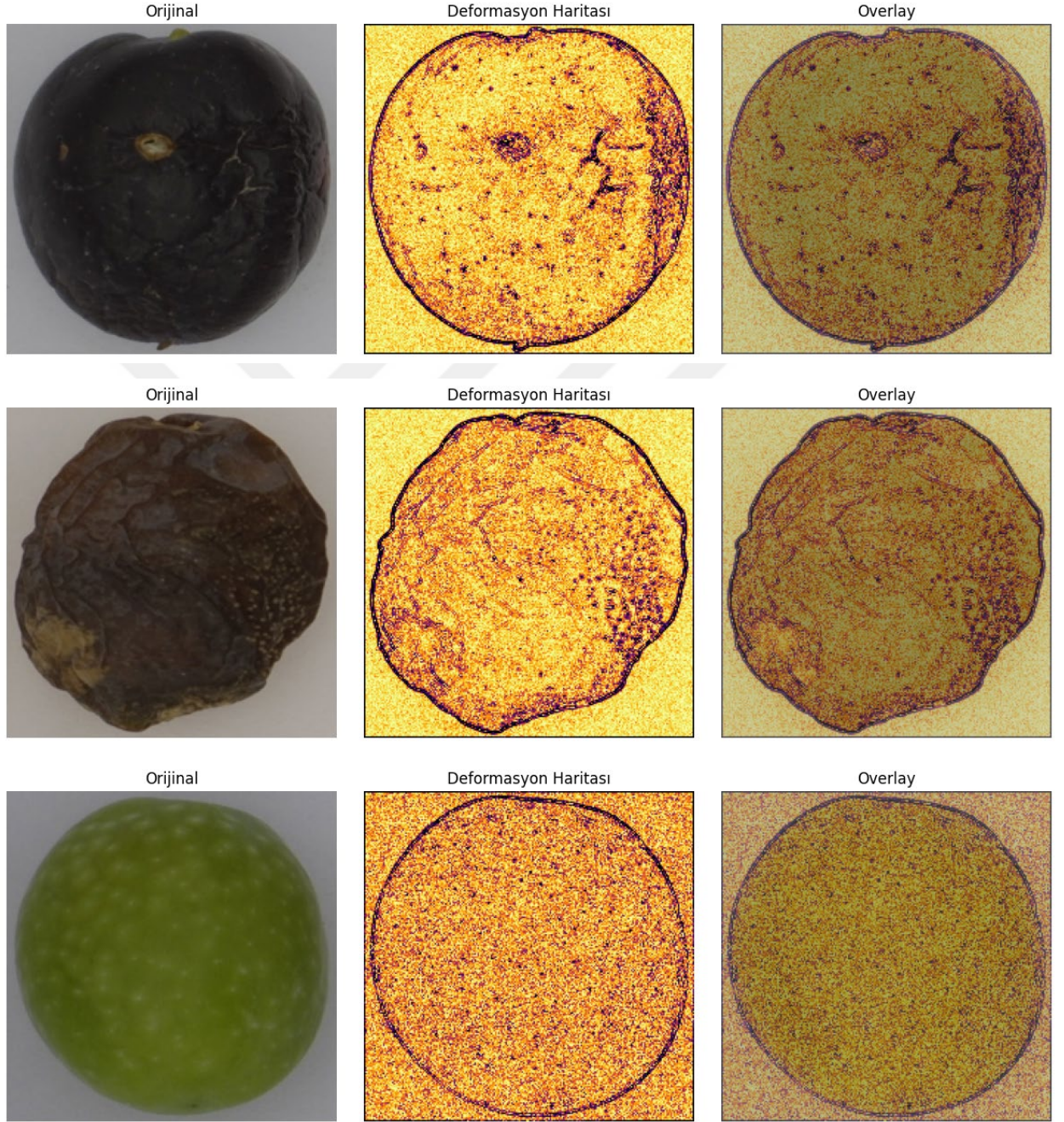
Modelin karar alma süreçlerini daha anlaşılır ve yorumlanabilir kılmak amacıyla, test görüntülerine ilişkin deformasyon tabanlı dikkat haritaları üretilmiştir. Bu haritalar klasik GRAD-CAM gibi geriye yayılım temelli yöntemlerden farklı olarak, önerilen mimaride doğrudan kullanılan DeformationScore modülünün ürettiği skor dağılımına dayanmaktadır. Modül her pikselin lokal çevresindeki yapısal bozulma düzeyini — özellikle kenar yoğunluğu ve pürüzlük bilgilerini — analiz ederek bir önemlilik skoru hesaplamaktadır.

Her bir örnek için üç aşamalı görselleştirme yapılmıştır:

- I. Orijinal Görüntü: Modelin sınıflandırma yürüttüğü ham giriş görüntüsü,
- II. Deformasyon Haritası: Yapısal bozulmaya dayalı olarak hesaplanan piksel skorları (ısı haritası),
- III. Overlay (Bindirme): Deformasyon haritasının orijinal görüntü ile üst üste getirilmiş görselleştirmesi.

Şekil 3-25'te gösterilen örneklerde, sırasıyla sinek ısırıklı, mantar enfeksiyonlu ve sağlıklı zeytin meyvesi örnekleri sunulmaktadır. İlk örnekte yüzey çöküntüleri yüksek skorlarla öne

çıkarken, ikinci örnekte mantar kaynaklı lezyonların bulunduğu alanlar yoğun biçimde vurgulanmıştır. Üçüncü örnek olan sağlıklı meyvede ise yapısal homojenliğe bağlı olarak düşük skor dağılımı gözlenmektedir.



Şekil 3- 25: Sırasıyla sinek ısırığı, mantar enfeksiyonu ve sağlıklı örnekler için DeformationScore haritaları ve bindirmeli görseller.

Overlay görseller, bu skorların orijinal görüntü ile birleştirilerek anlamlı bir bağlama oturtulmasını sağlar. Böylece modelin yalnızca hangi sınıfa karar verdiği değil, bu kararı verirken görüntünün hangi bölgelerini referans aldığı da açık biçimde izlenebilir hâle gelir. Özellikle tarımsal karar destek sistemlerinde açıklanabilirlik ihtiyacı göz önüne alındığında, bu yaklaşım hem güven artırıcı hem de kullanıcıya yönelik yorumlanabilirlik açısından yüksek katma değer taşımaktadır.

Teknik olarak DeformationScore modülü, önerilen mimarinin Metaformer iskeleti içinde yer alan Token Mixer bileşeni olarak yapılandırılmıştır. Model eğitimi sırasında bu modül, giriş tensörleri üzerinde lokal bozulma skorlarına dayalı dönüşümler gerçekleştirerek sınıflandırma sürecine katkı sağlamaktadır. Ancak burada sunulan deformasyon haritaları eğitim sonrası aşamada, aynı modülün giriş görüntüsüne bağımsız şekilde uygulanması ile elde edilmiştir. Bu bağlamda sunulan haritalar modelin iç katmanlarında oluşan tensörlerden değil Token Mixer'ın bağımsız olarak hesapladığı skorların görselleştirilmiş halinden oluşmaktadır.

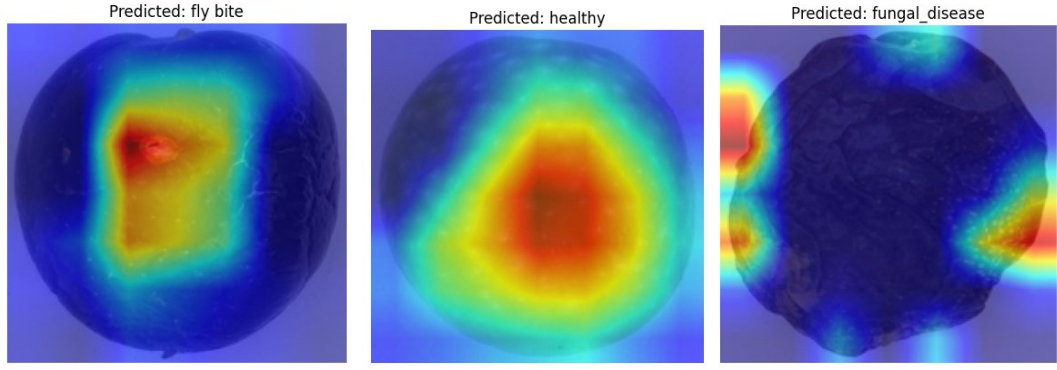
3.5.3 Grad-CAM Tabanlı Görsel Açıklanabilirlik

Modelin hangi bölgeleri dikkate alarak karar verdiğini analiz edebilmek amacıyla, Grad-CAM yöntemi uygulanmış ve her sınıf için örnek görüntü açıklamaları elde edilmiştir. Görselleştirme sürecinde modelin son aşamasındaki konvolüsyonel katman hedeflenmiştir.

Şekil 3-26'da üç farklı örneğin modelin sınıflandırma kararlarında hangi bölgelere odaklandığı görselleştirilmiştir:

- Sinek ısırganı meyvesi örneğinde, modelin zararlının doğrudan ısırık bölgesine yoğunlaştığı gözlemlenmiştir.
- Sağlıklı zeytin meyvesi sınıfında, meyvenin merkez bölgesine odaklanan yaygın bir dikkat deseni izlenmiştir.
- Mantar hastalıklı zeytin meyvesi örneğinde ise modelin kenar bölgelerdeki renk bozulmalarına ve yüzeysel doku anormalliklerine yüksek önem atfettiği görülmüştür.

Bu sonuçlar, modelin sınıflandırma kararlarını biyolojik olarak anlamlı bölgelere dayandığını göstermekte; ayrıca geliştirilen yapının açıklanabilirlik açısından da güven verici çıktılar sunduğunu ortaya koymaktadır.



Şekil 3- 26: Sinek ısırıklı, sağlıklı ve mantar hastalıklı sınıflarına ait Grad-CAM çıktıları.

3.6 Tartışma ve Sonuç

Bu çalışma, zeytin meyvesinde yaygın olarak karşılaşılan iki temel biyotik hastalık semptomunun –*Bactrocera oleae* zararlısı kaynaklı sinek ısırıkları ve çeşitli mantar etmeniyle gelişen fungal enfeksiyonların– derin öğrenme tabanlı sınıflandırma sistemleri aracılığıyla otomatik olarak teşhis edilmesini amaçlamış bu kapsamda hem klasik CNN tabanlı transfer öğrenme modelleri hem transformer temelli dikkat mekanizmaları hem de MetaFormer mimarisi üzerine inşa edilen özgün hibrit modeller kullanılmıştır. Çalışma, sınırlı hacimli ve etiketli tarımsal görüntü verilerinin yorumlanmasında derin öğrenme tabanlı yapıların avantajları, sınırlılıkları ve entegrasyon potansiyelleri açısından çok katmanlı bir tartışma alanı sunmaktadır.

- i. Çalışmanın ilk aşamasında, önceden ImageNet üzerinde eğitilmiş dokuz farklı CNN modeli (VGG16, ResNet50, InceptionV3, InceptionResNetV2, DenseNet121, Xception, MobileNetV2, NASNetMobile ve EfficientNetB0) zeytin meyvesi görüntüleri üzerinde yeniden eğitilmiştir. Bu modeller, hem Hold-Out hem de K-Fold çapraz doğrulama stratejileriyle test edilmiş ve performansları doğruluk, F1 skoru, karışıklık matrisi ve kayıp eğrileri gibi temel metriklerle değerlendirilmiştir.

DenseNet121, Xception ve NASNetMobile modelleri %97 üzeri doğruluk oranlarıyla öne çıkarken, ResNet50 ve InceptionV3 gibi daha karmaşık mimarilerin sınıflar arası ayrımı sağlamakta zaman zaman yetersiz kaldığı gözlemlenmiştir. Bu sonuç, transfer öğrenmenin başarısının yalnızca model derinliğine değil, aynı zamanda veri setinin doğasına, görsel varyasyon düzeyine ve önceden öğrenilen filtrelerin transfer edilebilirliğine bağlı olduğunu göstermektedir.

Transformer modellerinin (ViT, BEiT, LeViT, DeiT III, MaxViT, MobileViT v2) uygulanması, klasik CNN'lere göre özellikle sınıf içi varyasyonun yüksek olduğu durumlarda daha kararlı sınıflandırmalar sunduğunu ortaya koymuştur. Bu modellerin başarısı, görüntüdeki her yamanın tüm diğer yamalarla eş zamanlı etkileşim kurarak küresel bağlamı dikkate alabilmesiyle ilişkilidir. Özellikle ViT ve MaxViT, %98'in üzerinde F1 skoru ile yüksek performans göstermiştir. Buna karşın, bazı modellerde eğitim sürecinde optimizasyon seçimlerinin (ör. SGD yerine Adam veya RMSprop) model başarımında belirleyici olduğu gözlemlenmiştir.

Transformer'ların başarısı, zeytin gibi yapısal karmaşıklığa ve mikrodetaylara sahip tarımsal ürünlerde, klasik CNN'lerin lokal filtreleme yaklaşımının yetersiz kalabileceğini göstermektedir. Bu sonuç, uzun menzilli bağıntıların öğrenilmesini mümkün kılan dikkat mekanizmalarının, tarımsal görsellerde açıklanabilirlik ve doğruluk açısından önemli bir avantaj sunduğunu teyit etmektedir.

- ii. MetaFormer mimarisine dayalı hazır modellerden özellikle CAFormer-S18 ve PoolFormer-V2-S12 mimarileri, %99'a yakın doğruluk oranlarıyla çalışmanın en yüksek başarı gösteren sistemleri arasında yer almıştır. Bu mimariler, klasik self-attention içermemelerine rağmen, backbone'larındaki yapıların düzenleniş biçimi sayesinde yüksek verimlilik ve düşük hesaplama gereksinimiyle öne çıkmaktadır.
- iii. Özgün olarak geliştirilen altı hibrit MetaFormer modeli ise, Conv, Rand, Identity ve Transformer bloklarının farklı kombinasyonlarından oluşturulmuştur. Özellikle Model 1, Model 2 ve Model 4, %97,98'e varan doğruluk oranları ve dengeli F1 skorları ile dikkat çekmiştir. Model 3'ün düşük performansı, yalnızca bileşenlerin varlığı değil, bu bileşenlerin sıralaması ve geçiş düzeninin de model başarımı üzerinde doğrudan etkili olduğunu göstermektedir. Bu durum, blok dizilimi ve yapısal topoloji tasarımının özgün mimari geliştirme süreçlerinde ne denli kritik olduğunu ortaya koymaktadır.
- iv. Tezin en önemli özgün katkılarından biri olan DeformationScoreFormer adlı yeni model, yalnızca sınıflandırma başarısı (%97,98 doğruluk) ile değil, aynı zamanda açıklanabilirlik ve biyolojik anlamlılık açısından da literatüre yeni bir perspektif sunmuştur. Bu modelde

önerilen yeni token mixer modülü, lokal kenar yoğunluğu ve deformasyon paternlerini dikkate alarak daha odaklı skor üretimi sağlamaktadır.

Grad-CAM görselleştirmeleri, modelin biyolojik olarak anlamlı bölgelere (sinek ısırığı boşlukları, fungal çöküntüler vb.) odaklandığını ortaya koymuş, bu da geliştirilen sistemin karar mekanizmalarının yalnızca istatistiksel değil, aynı zamanda tarımsal uzmanlıkla uyumlu olduğunu kanıtlamıştır.

Bu tez, yalnızca model karşılaştırmaları sunmakla kalmamış aynı zamanda kendi içsel mimarisini, yeni yapı bloklarını ve açıklanabilirlik mekanizmalarını geliştirmiştir. Tarımsal görüntü işleme alanında derin öğrenme modellerinin uygulanabilirliği, veri kısıtları altında model dayanıklılığı, optimizasyon stratejilerinin etkisi ve görsel açıklanabilirliğin pratik anlamı gibi çok katmanlı konular bütüncül biçimde ele alınmıştır.

Sonuçlar göstermektedir ki geleneksel CNN modelleri sınırlı veri setlerinde yeterli performans gösterebilirken, özellikle çok sınıflı ve biyolojik varyasyonu yüksek problem uzaylarında transformer ve MetaFormer tabanlı sistemler daha kararlı ve açıklanabilir sonuçlar sunmaktadır. Özgün model önerileriyle bu tez, tarımda derin öğrenmenin daha verimli, modüler, açıklanabilir ve doğrudan saha uygulamasına uygun hale getirilmesine katkı sağlamaktadır.

3.7 Öneriler

Bu tez kapsamında geliştirilen derin öğrenme tabanlı sınıflandırma modelleri, zeytin meyvesinde yaygın olarak görülen biyotik hastalık semptomlarını yüksek doğrulukla tespit edebilmiştir. Özellikle MetaFormer iskeletine dayalı hibrit mimariler içerisinde, lokal kenar yoğunluğu ve şekil bozulmalarına duyarlı skor üretimi gerçekleştiren özgün DeformationScore modülünün entegre edildiği DeformationScoreFormer yapısı hem yüksek sınıflandırma başarımı hem de modelin açıklanabilirliği açısından öne çıkmıştır. Bu yaklaşım, klasik attention mekanizmalarına kıyasla daha hesaplamasal verimli bir alternatif sunarken sınıflandırma kararlarının biyolojik olarak anlamlı bölgeler üzerinden alınmasına da olanak sağlamıştır. Benzer modüllerin farklı bitki türleri ve hastalık semptomlarına uyarlanması, gelecekteki alan-adaptif tarımsal modelleme çalışmalarına katkı sağlayabilir.

Çalışmanın diğer önemli katkısı, sabit mimari kalıplar yerine konvolüsyonel, rastgele, kimliksel ve Transformer tabanlı blokların farklı dizilimlerle bir araya getirildiği adaptif MetaFormer yapılarının denenmesidir. Bu mimarilerle oluşturulan bazı modeller (özellikle Model 1, 2 ve 4), yüksek doğruluk ve F1 skoru ile dikkat çekmiş; ağ içi bileşenlerin veriye duyarlı biçimde yapılandırılmasının sınıflandırma performansını artırabileceği ortaya konmuştur. Gelecek çalışmalarda bu adaptif mimarilerin daha büyük veri kümeleri üzerinde dinamik biçimde optimize edilmesi önerilmektedir.

Bunun yanında, çalışmada CNN ve Transformer tabanlı modellerin farklı optimizasyon algoritmaları (SGD, Adafactor, Adadelat vb.) ile eğitilerek performanslarının karşılaştırılması, optimizasyon stratejilerinin sınıflandırma başarımı üzerindeki kritik etkisini göstermiştir. Özellikle Adafactor'un düşük kayıp ve dengeli F1 skorları, sınıflar arası denge sağlamada avantaj sunmuştur. Bu doğrultuda, ilerleyen çalışmalarda adaptif öğrenme oranı planlamaları, optimizasyon stratejisi seçimi için meta-öğrenme yöntemleri ya da veri yapısına özgü dinamik optimizasyon tercihi gibi ileri düzey tekniklerin değerlendirilmesi önerilmektedir.

Çalışmada kullanılan görüntülerin yalnızca belirli bir bölgeden (Bandırma-Edincik) ve sınırlı bir zaman diliminden elde edilmiş olması, modelin genellenebilirliği açısından sınırlayıcıdır. Bu nedenle, farklı bölgelerden, çeşitli fenolojik evrelerden ve farklı çevresel koşullardan elde edilecek örneklerle genişletilmiş veri setlerinin oluşturulması gerekmektedir. Ayrıca hastalıkların zaman içindeki gelişimini izlemeye olanak sağlayan zaman serisi tabanlı veri toplama yaklaşımları, yalnızca tanılamaya değil aynı zamanda erken uyarı ve öngörü sistemlerinin geliştirilmesine de katkı sağlayabilir.

Görsel verilerin ötesinde, nem, sıcaklık, ışık gibi çevresel faktörlerin entegre edilmesiyle oluşturulacak multimodal derin öğrenme sistemleri, sınıflandırma kararlarının daha sağlam temellere dayandırılmasını mümkün kılacaktır. Ayrıca, GAN veya varyasyonel otoenkoder gibi yöntemlerle sentetik veri üretimi yapılması, özellikle azınlık sınıfların temsilini artırarak modelin sınıf dengesi açısından daha adil hale getirilmesine katkı sağlayabilir.

Son olarak, geliştirilen mimarilerin gerçek sahada kullanılabilirliğini artırmak adına, modellerin mobil cihazlara veya gömülü sistemlere entegre edilmesine yönelik çalışmalar önerilmektedir. Özellikle DeformationScoreFormer gibi özgün yapılar, TensorRT veya TFLite

gibi mobil uyumlu framework'ler aracılığıyla optimize edilerek saha koşullarında gerçek zamanlı teşhis sistemlerinin geliştirilmesine olanak sağlayabilir.

Bu bütünsel öneriler, çalışmanın sunduğu adaptif, açıklanabilir ve uygulanabilir modelleme yaklaşımını sürdürerek derin öğrenmenin tarımsal karar destek sistemlerine entegrasyonunu güçlendirmeyi amaçlamaktadır.



4. KAYNAKLAR

- [1] A. Y. Kocadağlı, “TÜRKİYE’DE ZEYTİNCİLİK FAALİYETLERİNDE EDREMİT KÖRFEZİ KİYİLLERİNİN ÖNEMİ”.
- [2] R. Ilgar, “Çanakkale İlinde Zeytin Yetiştiriciliği ve Yaşanan Sorunlar”, 2016.
- [3] Özkan, Z. (2024). Zeytinyağı ve sofralık zeytin: 2023 ürün raporu (TEPGE Yayın No: 394). Tarımsal Ekonomi ve Politika Geliştirme Enstitüsü Müdürlüğü.
- [4] Tarımsal Ekonomi ve Politika Geliştirme Enstitüsü Müdürlüğü. (2023). Zeytinyağı ve sofralık zeytin: 2022 ürün raporu (Yayın No: 371). T.C. Tarım ve Orman Bakanlığı.
- [5] T.C. Gıda Tarım ve Hayvancılık Bakanlığı Gıda ve Kontrol Genel Müdürlüğü. (2016). Zeytin Hastalık ve Zararlıları İle Mücadele.
- [6] H. Basmacıoğlu-Malayoğlu ve B. Aktaş, “Zeytin Yağı İşleme Yan Ürünlerinden Zeytin Yaprığı ile Zeytin Karasuyunun Antimikrobiyal ve Antioksidan Etkileri”, 2011.
- [7] Elif Dikmen Diriöz, E., & Alkan, A. K. B. (2023). Zeytin ağacı: Tarihten gelen önemi ışığında Türkiye ve dünyada korunmasının uluslararası çevresel sürdürülebilirlik kapsamında önemi. *Uluslararası Çevre Araştırmaları ve Teknolojileri Dergisi*, 7(2), 134–145.
- [8] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis”, *Comput. Electron. Agric.*, c. 145, ss. 311-318, Şub. 2018, doi: 10.1016/j.compag.2018.01.009.
- [9] S. Özözen, “TÜRKİYE’NİN ZEYTİN VE ZEYTİNYAĞI SEKTÖRÜNDE KÜRESEL REKABET GÜCÜNÜN DEĞERLENDİRİLMESİ”, *Önetim Bilim. Derg.*, c. 22, sy 53, ss. 1084-1117, Tem. 2024, doi: 10.35408/comuybd.1433166.
- [10] Aktepe Tangu, N. (t.y.). Zeytin yetiştiriciliği [Broşür]. T.C. Tarım ve Orman Bakanlığı, Tarımsal Araştırmalar ve Politikalar Genel Müdürlüğü, Atatürk Bahçe Kültürleri Merkez Araştırma Enstitüsü Müdürlüğü., “ZEYTİN YETİŞTİRİCİLİĞİ”.
- [11] Dıraman, H. (2006). Zeytinyağı Kalitesini Nasıl İyileştirebiliriz ?. *Akademik Gıda*, 4(4), 41-42.
- [12] S. Özpınar, A. Özpınar, A. K. Şahin, B. Polat, ve B. Büyükcan, “Çanakkale İlinde Toprak İşlemenin Zeytin Sineği (*Bactrocera oleae* Gmelin, 1790, Diptera:Tephritidae)’nin Popülasyon Yoğunluğun”.
- [13] T.C. Gıda, Tarım ve Hayvancılık Bakanlığı Tarımsal Araştırmalar ve Politikalar Genel Müdürlüğü. (2011). Zeytin Entegre Mücadele Teknik Talimatı. [Çevrimiçi]. Erişim adresi: https://www.tarimorman.gov.tr/TAGEM/Belgeler/yayin/005_zeytin.pdf
- [14] P. Karahocagil vd., “TÜRKİYE’DE ZEYTİNYAĞI VE SOFRALIK ZEYTİN”.
- [15] Karakoyun, N. S., & Akça Uçkun, A. (2023). Zeytin Çeşitlerinin, Zeytin Sineği (*Bactrocera Oleae*)’ne Karşı Duyarlılıkları ve Zararın Kalite Üzerindeki Etkileri. Zeytincilik Araştırma Enstitüsü, İzmir.

- [16] L. Grandi *vd.*, “Innovations towards sustainable olive crop management: a new dawn by precision agriculture including endo-therapy”, *Front. Plant Sci.*, c. 14, s. 1180632, Haz. 2023, doi: 10.3389/fpls.2023.1180632.
- [17] Gözel, H. (2023). Zeytinde Hasat Zamanı ve Yöntemleri. Antepfıstığı Araştırma Dergisi, Antepfıstığı Araştırma İstasyonu Müdürlüğü.
- [18] R. Efe, *Dünyada, Türkiye’de, Edremit Körfezi çevresinde zeytin ve zeytinyağı*. Balıkesir: Edremit Belediyesi, 2013.
- [19] K. Yousfi, C. M. Weiland, ve J. M. García, “Effect of Harvesting System and Fruit Cold Storage on Virgin Olive Oil Chemical Composition and Quality of Superintensive Cultivated ‘Arbequina’ Olives”, *J. Agric. Food Chem.*, c. 60, sy 18, ss. 4743-4750, May. 2012, doi: 10.1021/jf300331q.
- [20] H. Altekin ve N. Demirbaş, “Kalite ve Miktar Kayıpları Açısından Dip Zeytin Hasadıyla İlgili Üretici Görüşlerinin Belirlenmesi : Foça İlçesi Örneği”.
- [21] A. M. Chacón-Maldonado, L. Melgar-García, G. Asencio-Cortés, ve A. Troncoso, “A novel method based on hybrid deep learning with explainability for olive fruit pest forecasting”, *Neural Comput. Appl.*, c. 37, sy 5, ss. 3245-3264, Şub. 2025, doi: 10.1007/s00521-024-10731-z.
- [22] H. Miho *vd.*, “The dynamic changes in olive fruit phenolic metabolism and its contribution to the activation of quiescent Colletotrichum infection”, *Food Chem.*, c. 450, s. 139299, Ağu. 2024, doi: 10.1016/j.foodchem.2024.139299.
- [23] M. A. Mahmood ve K. Alsalem, “Olive Leaf Disease Detection via Wavelet Transform and Feature Fusion of Pre-Trained Deep Learning Models”, *Comput. Mater. Contin.*, c. 78, sy 3, ss. 3431-3448, 2024, doi: 10.32604/cmc.2024.047604.
- [24] N. M. H. Hassan, A. A. Donkol, M. M. Mabrook, ve A. M. Mabrouk, “Highly Efficient Machine Learning Approach for Automatic Disease and Color Classification of Olive Fruits”, *IEEE Access*, c. 12, ss. 35683-35699, 2024, doi: 10.1109/ACCESS.2024.3362294.
- [25] Diker, A., Elen, A., Közkurt, C., Kılıçarslan, S., Dönmez, E., Arslan, K., & Kuran, E. C. (2024). An effective feature extraction method for olive peacock eye leaf disease classification. *European Food Research and Technology*, 250(1), 287–299. <https://doi.org/10.1007/s00217-023-04386-8>
- [26] M. Macías-Macías, H. Sánchez-Santamaria, C. J. García Orellana, H. M. González-Velasco, R. Gallardo-Caballero, ve A. García-Manso, “Mask R-CNN for quality control of table olives”, *Multimed. Tools Appl.*, c. 82, sy 14, ss. 21657-21671, Haz. 2023, doi: 10.1007/s11042-023-14668-8.
- [27] H. El Akhal, A. Ben Yahya, N. Moussa, ve A. El Belrhiti El Alaoui, “A novel approach for image-based olive leaf diseases classification using a deep hybrid model”, *Ecol. Inform.*, c. 77, s. 102276, Kas. 2023, doi: 10.1016/j.ecoinf.2023.102276.

- [28] H. H. Alshammari, A. I. Taloba, ve O. R. Shahin, "Identification of olive leaf disease through optimized deep learning approach", *Alex. Eng. J.*, c. 72, ss. 213-224, Haz. 2023, doi: 10.1016/j.aej.2023.03.081.
- [29] A. Ksibi, M. Ayadi, B. O. Soufiene, M. M. Jamjoom, ve Z. Ullah, "MobiRes-Net: A Hybrid Deep Learning Model for Detecting and Classifying Olive Leaf Diseases", *Appl. Sci.*, c. 12, sy 20, s. 10278, Eki. 2022, doi: 10.3390/app122010278.
- [30] E. M. Raouhi, M. Lachgar, H. Hrimech, ve A. Kartit, "Optimization techniques in deep convolutional neuronal networks applied to olive diseases classification", *Artif. Intell. Agric.*, c. 6, ss. 77-89, 2022, doi: 10.1016/j.aiia.2022.06.001.
- [31] B. Dikici, M. F. Bekçioğullari, H. Açıkgöz, ve D. Korkmaz, "ZEYTİN YAPRAĞINDAKİ HASTALIKLARIN SINIFLANDIRILMASINDA ÖN EĞİTİMLİ EVRİŞİMLİ SINIR AĞLARININ PERFORMANSLARININ İNCELENMESİ", *Konya J. Eng. Sci.*, c. 10, sy 3, ss. 535-547, Eyl. 2022, doi: 10.36306/konjes.1078358.
- [32] A. Sinha ve R. S. Shekhawat, "Olive Spot Disease Detection and Classification using Analysis of Leaf Image Textures", *Procedia Comput. Sci.*, c. 167, ss. 2328-2336, 2020, doi: 10.1016/j.procs.2020.03.285.
- [33] M. Alruwaili, S. Alanazi, S. Abd, ve A. Shehab, "An Efficient Deep Learning Model for Olive Diseases Detection", *Int. J. Adv. Comput. Sci. Appl.*, c. 10, sy 8, 2019, doi: 10.14569/IJACSA.2019.0100863.
- [34] R. Haff, E. Jackson, R. Moschetti, ve R. Massantini, "Detection of Fruit-fly Infestation in Olives using X-Ray Imaging: Algorithm Development and Prospects", *Am. J. Agric. Sci. Technol.*, 2016, doi: 10.7726/ajast.2016.1001.
- [35] M. Kuncan, H. M. Ertunç, G. Küçükyıldız, B. Hızarcı, H. Ocak, ve S. Öztürk, "Görüntü İşleme Tabanlı Zeytin Ayıklama Makinesi", 2013.
- [36] M. T. Riquelme, P. Barreiro, M. Ruiz-Altisent, ve C. Valero, "Olive classification according to external damage using image analysis", *J. Food Eng.*, c. 87, sy 3, ss. 371-379, Ağu. 2008, doi: 10.1016/j.jfoodeng.2007.12.018.
- [37] M. J. Ayora-Cañada, B. Muik, J. A. García-Mesa, D. Ortega-Calderón, ve A. Molina-Díaz, "Fourier-Transform Near-Infrared Spectroscopy as a Tool for Olive Fruit Classification and Quantitative Analysis", *Spectrosc. Lett.*, c. 38, sy 6, ss. 769-785, Kas. 2005, doi: 10.1080/00387010500316106.
- [38] S. Kiliçarslan, "Kuru Üzüm Tanelerinin Sınıflandırılması için Hibrit Bir Yaklaşım A Hybrid Approach for Raisin Grains Classification".
- [39] E. Sert, "A deep learning based approach for the detection of diseases in pepper and potato leaves", *ANADOLU J. Agric. Sci.*, ss. 167-178, Haz. 2021, doi: 10.7161/omuanajas.805152.
- [40] M. Göksu, K. M. Sünnetci, ve A. Alkan, "Derin öğrenme ağları kullanılarak mısır yapraklarında hastalık tespiti", *Comput. Sci.*, Eyl. 2021, doi: 10.53070/bbd.989305.

- [41] M. Türkoğlu, K. Hanbay, I. Saraç SiVriKaya, ve D. Hanbay, “Derin Evrişimsel Sinir Ağı Kullanılarak Kayısı Hastalıklarının Sınıflandırılması”, *Bitlis Eren Üniversitesi Fen Bilim. Derg.*, c. 9, sy 1, ss. 334-345, Mar. 2020, doi: 10.17798/bitlisfen.562101.
- [42] B. Duman ve K. Kayaalp, “Detection of Harvest Status of Oil Rose (*Rosa damascena* Mill.) with Machine Learning and Deep Learning Methods”, *El-Cezeri Fen Ve Mühendis. Derg.*, Eyl. 2022, doi: 10.31202/ecjse.1134822.
- [43] S. Ertem ve E. Özbay, “Sınıflandırma Probleminde Derin Özellik Birleştirme Yaklaşımıyla Domates Yaprığı Görüntülerinde Hastalık Tespiti”, *Eur. J. Sci. Technol.*, Ara. 2022, doi: 10.31590/ejosat.1216380.
- [44] T. Ananthanarayana, R. Ptucha, ve S. C. Kelly, “Deep Learning based Fruit Freshness Classification and Detection with CMOS Image sensors and Edge processors”, *Electron. Imaging*, c. 32, sy 12, ss. 172-1-172-7, Oca. 2020, doi: 10.2352/ISSN.2470-1173.2020.12.FAIS-172.
- [45] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms”, *IEEE Trans. Syst. Man Cybern.*, c. 9, sy 1, ss. 62-66, Oca. 1979, doi: 10.1109/TSMC.1979.4310076.
- [46] Shapiro, L. G., & Stockman, G. C. (2001). *Computer Vision*. Prentice Hall.
- [47] R. C. Gonzalez ve R. E. Woods, *Digital image processing*, Fourth edition, Global edition. New York: Pearson, 2017.
- [48] S. Suzuki, “Topological Structural Analysis of Digitized Binary Images by Border Following”.
- [49] L. Sun, S. Zhao, G. Li, ve B. Liu, “High accuracy object detection via bounding box regression network”, *Front. Optoelectron.*, c. 12, sy 3, ss. 324-331, Eyl. 2019, doi: 10.1007/s12200-019-0853-1.
- [50] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- [51] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- [52] S. Albawi, T. A. Mohammed, ve S. Al-Zawi, “Understanding of a convolutional neural network”, içinde *2017 International Conference on Engineering and Technology (ICET)*, Antalya: IEEE, Ağu. 2017, ss. 1-6. doi: 10.1109/ICEngTechnol.2017.8308186.
- [53] V. Dumoulin ve F. Visin, “A guide to convolution arithmetic for deep learning”, 11 Ocak 2018, *arXiv*: arXiv:1603.07285. doi: 10.48550/arXiv.1603.07285.
- [54] V. Nair ve G. E. Hinton, “Rectified Linear Units Improve Restricted Boltzmann Machines”.
- [55] D. Hutchison vd., “Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition”, içinde *Artificial Neural Networks – ICANN 2010*, c. 6354, K. Diamantaras, W. Duch, ve L. S. Iliadis, Ed., içinde *Lecture Notes in Computer Science*, vol.

6354. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, ss. 92-101. doi: 10.1007/978-3-642-15825-4_10.
- [56] M. D. Zeiler ve R. Fergus, “Visualizing and Understanding Convolutional Networks”, 28 Kasım 2013, *arXiv*: arXiv:1311.2901. doi: 10.48550/arXiv.1311.2901.
- [57] G. Litjens *vd.*, “A Survey on Deep Learning in Medical Image Analysis”, *Med. Image Anal.*, c. 42, ss. 60-88, Ara. 2017, doi: 10.1016/j.media.2017.07.005.
- [58] A. Kamilaris ve F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey”, *Comput. Electron. Agric.*, c. 147, ss. 70-90, Nis. 2018, doi: 10.1016/j.compag.2018.02.016.
- [59] S. J. Pan ve Q. Yang, “A Survey on Transfer Learning”, *IEEE Trans. Knowl. Data Eng.*, c. 22, sy 10, ss. 1345-1359, Eki. 2010, doi: 10.1109/TKDE.2009.191.
- [60] H.-C. Shin *vd.*, “Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning”, *IEEE Trans. Med. Imaging*, c. 35, sy 5, ss. 1285-1298, May. 2016, doi: 10.1109/TMI.2016.2528162.
- [61] J. Yosinski, J. Clune, Y. Bengio, ve H. Lipson, “How transferable are features in deep neural networks?”, 06 Kasım 2014, *arXiv*: arXiv:1411.1792. doi: 10.48550/arXiv.1411.1792.
- [62] S. P. Mohanty, D. P. Hughes, ve M. Salathé, “Using Deep Learning for Image-Based Plant Disease Detection”, *Front. Plant Sci.*, c. 7, s. 1419, Eyl. 2016, doi: 10.3389/fpls.2016.01419.
- [63] K. Simonyan ve A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, 10 Nisan 2015, *arXiv*: arXiv:1409.1556. doi: 10.48550/arXiv.1409.1556.
- [64] A. Krizhevsky, I. Sutskever, ve G. E. Hinton, “ImageNet classification with deep convolutional neural networks”, *Commun. ACM*, c. 60, sy 6, ss. 84-90, May. 2017, doi: 10.1145/3065386.
- [65] W. Rawat ve Z. Wang, “Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review”, *Neural Comput.*, c. 29, sy 9, ss. 2352-2449, Eyl. 2017, doi: 10.1162/neco_a_00990.
- [66] M. Lin, Q. Chen, ve S. Yan, “Network In Network”, 04 Mart 2014, *arXiv*: arXiv:1312.4400. doi: 10.48550/arXiv.1312.4400.
- [67] M. Tan ve Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”, 11 Eylül 2020, *arXiv*: arXiv:1905.11946. doi: 10.48550/arXiv.1905.11946.
- [68] K. He, X. Zhang, S. Ren, ve J. Sun, “Deep Residual Learning for Image Recognition”, içinde *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, Haz. 2016, ss. 770-778. doi: 10.1109/CVPR.2016.90.
- [69] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, ve C. Liu, “A Survey on Deep Transfer Learning”, 06 Ağustos 2018, *arXiv*: arXiv:1808.01974. doi: 10.48550/arXiv.1808.01974.
- [70] C. Szegedy, S. Ioffe, V. Vanhoucke, ve A. Alemi, “Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning”, *Proc. AAAI Conf. Artif. Intell.*, c. 31, sy 1, Şub.

- 2017, doi: 10.1609/aaai.v31i1.11231.
- [71] X. Zhang, J. Zou, K. He, ve J. Sun, “Accelerating Very Deep Convolutional Networks for Classification and Detection”, 18 Kasım 2015, *arXiv*: arXiv:1505.06798. doi: 10.48550/arXiv.1505.06798.
- [72] S. Ioffe ve C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”, 02 Mart 2015, *arXiv*: arXiv:1502.03167. doi: 10.48550/arXiv.1502.03167.
- [73] C. Szegedy *vd.*, “Going Deeper with Convolutions”, 17 Eylül 2014, *arXiv*: arXiv:1409.4842. doi: 10.48550/arXiv.1409.4842.
- [74] M. Z. Alom *vd.*, “A State-of-the-Art Survey on Deep Learning Theory and Architectures”, *Electronics*, c. 8, sy 3, s. 292, Mar. 2019, doi: 10.3390/electronics8030292.
- [75] R. Müller, S. Kornblith, ve G. Hinton, “When Does Label Smoothing Help?”, 10 Haziran 2020, *arXiv*: arXiv:1906.02629. doi: 10.48550/arXiv.1906.02629.
- [76] M. Q. Kheder ve A. A. Mohammed, “Transfer Learning Based Traffic Light Detection and Recognition Using CNN Inception-V3 Model”, *Iraqi J. Sci.*, ss. 6258-6275, Eki. 2023, doi: 10.24996/ijs.2023.64.10.40.
- [77] S. Bianco, R. Cadene, L. Celona, ve P. Napoletano, “Benchmark Analysis of Representative Deep Neural Network Architectures”, *IEEE Access*, c. 6, ss. 64270-64277, 2018, doi: 10.1109/ACCESS.2018.2877890.
- [78] G. Huang, Z. Liu, L. Van Der Maaten, ve K. Q. Weinberger, “Densely Connected Convolutional Networks”, içinde *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI: IEEE, Tem. 2017, ss. 2261-2269. doi: 10.1109/CVPR.2017.243.
- [79] X. Chen, S. Xie, ve K. He, “An Empirical Study of Training Self-Supervised Vision Transformers”, 16 Ağustos 2021, *arXiv*: arXiv:2104.02057. doi: 10.48550/arXiv.2104.02057.
- [80] J. Gu *vd.*, “Recent Advances in Convolutional Neural Networks”, 19 Ekim 2017, *arXiv*: arXiv:1512.07108. doi: 10.48550/arXiv.1512.07108.
- [81] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, ve K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size”, 04 Kasım 2016, *arXiv*: arXiv:1602.07360. doi: 10.48550/arXiv.1602.07360.
- [82] S. Jegou, M. Drozdal, D. Vazquez, A. Romero, ve Y. Bengio, “The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation”, içinde *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Honolulu, HI, USA: IEEE, Tem. 2017, ss. 1175-1183. doi: 10.1109/CVPRW.2017.156.
- [83] M. Tan ve Q. V. Le, “EfficientNetV2: Smaller Models and Faster Training”, 23 Haziran 2021, *arXiv*: arXiv:2104.00298. doi: 10.48550/arXiv.2104.00298.

- [84] N. Tajbakhsh *vd.*, “Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?”, *IEEE Trans. Med. Imaging*, c. 35, sy 5, ss. 1299-1312, May. 2016, doi: 10.1109/TMI.2016.2535302.
- [85] D. Xiao, Y. Huang, C. Qin, Z. Liu, Y. Li, ve C. Liu, “Transfer learning with convolutional neural networks for small sample size problem in machinery fault diagnosis”, *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.*, c. 233, sy 14, ss. 5131-5143, Tem. 2019, doi: 10.1177/0954406219840381.
- [86] B. Zoph, V. Vasudevan, J. Shlens, ve Q. V. Le, “Learning Transferable Architectures for Scalable Image Recognition”, 11 Nisan 2018, *arXiv*: arXiv:1707.07012. doi: 10.48550/arXiv.1707.07012.
- [87] T. Elsken, J. H. Metzen, ve F. Hutter, “Neural Architecture Search: A Survey”, 26 Nisan 2019, *arXiv*: arXiv:1808.05377. doi: 10.48550/arXiv.1808.05377.
- [88] M. Tan *vd.*, “MnasNet: Platform-Aware Neural Architecture Search for Mobile”, içinde *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: IEEE, Haz. 2019, ss. 2815-2823. doi: 10.1109/CVPR.2019.00293.
- [89] H. Pham, M. Y. Guan, B. Zoph, Q. V. Le, ve J. Dean, “Efficient Neural Architecture Search via Parameter Sharing”, 12 Şubat 2018, *arXiv*: arXiv:1802.03268. doi: 10.48550/arXiv.1802.03268.
- [90] H. Cai, L. Zhu, ve S. Han, “ProxylessNAS: Direct Neural Architecture Search on Target Task and Hardware”, 23 Şubat 2019, *arXiv*: arXiv:1812.00332. doi: 10.48550/arXiv.1812.00332.
- [91] A. G. Howard *vd.*, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”, 17 Nisan 2017, *arXiv*: arXiv:1704.04861. doi: 10.48550/arXiv.1704.04861.
- [92] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, ve L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks”, içinde *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT: IEEE, Haz. 2018, ss. 4510-4520. doi: 10.1109/CVPR.2018.00474.
- [93] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions”, içinde *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI: IEEE, Tem. 2017, ss. 1800-1807. doi: 10.1109/CVPR.2017.195.
- [94] Sifre, L., & Mallat, S. (2014). Rigid-motion scattering for image classification. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 373(2046), 20140187.
- [95] R. A. Hazarika, D. Kandar, ve A. K. Maji, “An experimental analysis of different Deep Learning based Models for Alzheimer’s Disease classification using Brain Magnetic Resonance Images”, *J. King Saud Univ. - Comput. Inf. Sci.*, c. 34, sy 10, ss. 8576-8598, Kas. 2022, doi: 10.1016/j.jksuci.2021.09.003.

- [96] Khan, S., Rahmani, H., Shah, S. A. A., & Bennamoun, M. (2022). A guide to convolutional neural networks for computer vision. Springer Nature.
- [97] F. Zhuang *vd.*, “A Comprehensive Survey on Transfer Learning”, *Proc. IEEE*, c. 109, sy 1, ss. 43-76, Oca. 2021, doi: 10.1109/JPROC.2020.3004555.
- [98] J. Hu, L. Shen, ve G. Sun, “Squeeze-and-Excitation Networks”.
- [99] Q. Xie, M.-T. Luong, E. Hovy, ve Q. V. Le, “Self-training with Noisy Student improves ImageNet classification”, 19 Haziran 2020, *arXiv*: arXiv:1911.04252. doi: 10.48550/arXiv.1911.04252.
- [100] Y. LeCun, Y. Bengio, ve G. Hinton, “Deep learning”, *Nature*, c. 521, sy 7553, ss. 436-444, May. 2015, doi: 10.1038/nature14539.
- [101] Z. Liu *vd.*, “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows”, içinde *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE, Eki. 2021, ss. 9992-10002. doi: 10.1109/ICCV48922.2021.00986.
- [102] A. Dosovitskiy *vd.*, “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”, 03 Haziran 2021, *arXiv*: arXiv:2010.11929. doi: 10.48550/arXiv.2010.11929.
- [103] A. Vaswani *vd.*, “Attention Is All You Need”, 02 Ağustos 2023, *arXiv*: arXiv:1706.03762. doi: 10.48550/arXiv.1706.03762.
- [104] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, ve S. Zagoruyko, “End-to-End Object Detection with Transformers”, 28 Mayıs 2020, *arXiv*: arXiv:2005.12872. doi: 10.48550/arXiv.2005.12872.
- [105] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, ve M. Shah, “Transformers in Vision: A Survey”, *ACM Comput. Surv.*, c. 54, sy 10s, ss. 1-41, Oca. 2022, doi: 10.1145/3505244.
- [106] S. Bhojanapalli, A. Chakrabarti, D. Glasner, D. Li, T. Unterthiner, ve A. Veit, “Understanding Robustness of Transformers for Image Classification”, içinde *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE, Eki. 2021, ss. 10211-10221. doi: 10.1109/ICCV48922.2021.01007.
- [107] J. Devlin, M.-W. Chang, K. Lee, ve K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, 24 Mayıs 2019, *arXiv*: arXiv:1810.04805. doi: 10.48550/arXiv.1810.04805.
- [108] H. Bao, L. Dong, S. Piao, ve F. Wei, “BEiT: BERT Pre-Training of Image Transformers”, 03 Eylül 2022, *arXiv*: arXiv:2106.08254. doi: 10.48550/arXiv.2106.08254.
- [109] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, ve R. Girshick, “Masked Autoencoders Are Scalable Vision Learners”, 19 Aralık 2021, *arXiv*: arXiv:2111.06377. doi: 10.48550/arXiv.2111.06377.

- [110] Z. Xie *vd.*, “SimMIM: a Simple Framework for Masked Image Modeling”, içinde *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA: IEEE, Haz. 2022, ss. 9643-9653. doi: 10.1109/CVPR52688.2022.00943.
- [111] Y. Fang *vd.*, “EVA: Exploring the Limits of Masked Visual Representation Learning at Scale”, içinde *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, BC, Canada: IEEE, Haz. 2023, ss. 19358-19369. doi: 10.1109/CVPR52729.2023.01855.
- [112] H. Touvron, M. Cord, ve H. Jégou, “DeiT III: Revenge of the ViT”, içinde *Computer Vision – ECCV 2022*, c. 13684, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, ve T. Hassner, Ed., içinde *Lecture Notes in Computer Science*, vol. 13684. , Cham: Springer Nature Switzerland, 2022, ss. 516-533. doi: 10.1007/978-3-031-20053-3_30.
- [113] L. Beyer, X. Zhai, A. Royer, L. Markeeva, R. Anil, ve A. Kolesnikov, “Knowledge distillation: A good teacher is patient and consistent”, 21 Haziran 2022, *arXiv*: arXiv:2106.05237. doi: 10.48550/arXiv.2106.05237.
- [114] S. Mehta ve M. Rastegari, “Separable Self-attention for Mobile Vision Transformers”, 06 Haziran 2022, *arXiv*: arXiv:2206.02680. doi: 10.48550/arXiv.2206.02680.
- [115] A. Howard *vd.*, “Searching for MobileNetV3”, içinde *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South): IEEE, Eki. 2019, ss. 1314-1324. doi: 10.1109/ICCV.2019.00140.
- [116] Z. Tu *vd.*, “MaxViT: Multi-Axis Vision Transformer”, 09 Eylül 2022, *arXiv*: arXiv:2204.01697. doi: 10.48550/arXiv.2204.01697.
- [117] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, ve S. Xie, “A ConvNet for the 2020s”.
- [118] K. Li *vd.*, “UNIFORMER: UNIFIED TRANSFORMER FOR EFFICIENT SPATIOTEMPORAL REPRESENTATION LEARNING”, 2022.
- [119] B. Graham *vd.*, “LeViT: a Vision Transformer in ConvNet’s Clothing for Faster Inference”, 06 Mayıs 2021, *arXiv*: arXiv:2104.01136. doi: 10.48550/arXiv.2104.01136.
- [120] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, ve H. Jégou, “Training data-efficient image transformers & distillation through attention”, 15 Ocak 2021, *arXiv*: arXiv:2012.12877. doi: 10.48550/arXiv.2012.12877.
- [121] M. F. Özdemir, “Çoklu Nesne Takibi FairMOT Algoritması İçin Optimizasyon Algoritmalarının Karşılaştırılması”, *Comput. Sci.*, Eyl. 2021, doi: 10.53070/bbd.990086.
- [122] E. Seyyarer, A. Karci, ve A. Ateş, “Stokastik ve deterministik hareketlerin optimizasyon süreçlerindeki etkileri”, *Gazi Üniversitesi Mühendis. Mimar. Fakültesi Derg.*, c. 37, sy 2, ss. 949-966, Şub. 2022, doi: 10.17341/gazimmfd.887976.
- [123] R. Zaheer ve H. Shaziya, “A Study of the Optimization Algorithms in Deep Learning”, içinde *2019 Third International Conference on Inventive Systems and Control (ICISC)*,

- Coimbatore, India: IEEE, Oca. 2019, ss. 536-539. doi: 10.1109/ICISC44355.2019.9036442.
- [124] R. Hadipour-Rokni, E. Askari Asli-Ardeh, A. Jahanbakhshi, I. Esmaili paeen-Afrakoti, ve S. Sabzi, “Intelligent detection of citrus fruit pests using machine vision system and convolutional neural network through transfer learning technique”, *Comput. Biol. Med.*, c. 155, s. 106611, Mar. 2023, doi: 10.1016/j.compbiomed.2023.106611.
- [125] R. S. Kale ve S. Shitole, “Deep learning optimizer performance analysis for pomegranate fruit quality gradation”, içinde *2022 IEEE Bombay Section Signature Conference (IBSSC)*, Mumbai, India: IEEE, Ara. 2022, ss. 1-5. doi: 10.1109/IBSSC56953.2022.10037429.
- [126] A. M. Taqi, A. Awad, F. Al-Azzo, ve M. Milanova, “The Impact of Multi-Optimizers and Data Augmentation on TensorFlow Convolutional Neural Network Performance”, içinde *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, Miami, FL: IEEE, Nis. 2018, ss. 140-145. doi: 10.1109/MIPR.2018.00032.
- [127] S. Y. Sen ve N. Ozkurt, “Convolutional Neural Network Hyperparameter Tuning with Adam Optimizer for ECG Classification”, içinde *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, Istanbul, Turkey: IEEE, Eki. 2020, ss. 1-6. doi: 10.1109/ASYU50717.2020.9259896.
- [128] B. Karadağ, A. Ari, ve M. Karadağ, “Derin Öğrenme Modellerinin Sinirsel Stil Aktarımı Performanslarının Karşılaştırılması”, *Politek. Derg.*, c. 24, sy 4, ss. 1611-1622, Ara. 2021, doi: 10.2339/politeknik.885838.
- [129] B. Xiao, Y. Liu, ve B. Xiao, “Accurate State-of-Charge Estimation Approach for Lithium-Ion Batteries by Gated Recurrent Unit With Ensemble Optimizer”, *IEEE Access*, c. 7, ss. 54192-54202, 2019, doi: 10.1109/ACCESS.2019.2913078.
- [130] J. Sun, P. Li, ve X. Wu, “Handwritten Ancient Chinese Character Recognition Algorithm Based on Improved Inception-ResNet and Attention Mechanism”, içinde *2022 IEEE 2nd International Conference on Software Engineering and Artificial Intelligence (SEAI)*, Xiamen, China: IEEE, Haz. 2022, ss. 31-35. doi: 10.1109/SEAI55746.2022.9832356.
- [131] S. Panigrahy, S. Karmakar, ve R. Sahoo, “Condition Assessment of High Voltage Insulator using Convolutional Neural Network”, içinde *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, Bangalore, India: IEEE, Tem. 2021, ss. 1-6. doi: 10.1109/CONECCT52877.2021.9622361.
- [132] N. Shazeer ve M. Stern, “Adafactor: Adaptive Learning Rates with Sublinear Memory Cost”.
- [133] L. Liu vd., “On the Variance of the Adaptive Learning Rate and Beyond”, 26 Ekim 2021, *arXiv*: arXiv:1908.03265. doi: 10.48550/arXiv.1908.03265.
- [134] I. Loshchilov ve F. Hutter, “Decoupled Weight Decay Regularization”, 04 Ocak 2019, *arXiv*: arXiv:1711.05101. doi: 10.48550/arXiv.1711.05101.

- [135] W. Yu *vd.*, “MetaFormer Is Actually What You Need for Vision”, 04 Temmuz 2022, *arXiv*: arXiv:2111.11418. doi: 10.48550/arXiv.2111.11418.
- [136] W. Yu *vd.*, “MetaFormer Baselines for Vision”, *IEEE Trans. Pattern Anal. Mach. Intell.*, c. 46, sy 2, ss. 896-912, Şub. 2024, doi: 10.1109/TPAMI.2023.3329173.
- [137] D. Li, Y. Bai, K. Wang, J. Jiang, X. Liu, ve W. Gao, “GroupedMixer: An Entropy Model with Group-wise Token-Mixers for Learned Image Compression”, *IEEE Trans. Circuits Syst. Video Technol.*, c. 34, sy 10, ss. 9606-9619, Eki. 2024, doi: 10.1109/TCSVT.2024.3395481.
- [138] W. Peebles ve S. Xie, “Scalable Diffusion Models with Transformers”, 02 Mart 2023, *arXiv*: arXiv:2212.09748. doi: 10.48550/arXiv.2212.09748.
- [139] H. Wang, S. Zhang, ve B. Leng, “HGFormer: Topology-Aware Vision Transformer with HyperGraph Learning”, 09 Nisan 2025, *arXiv*: arXiv:2504.02440. doi: 10.48550/arXiv.2504.02440.
- [140] H. Wu *vd.*, “CvT: Introducing Convolutions to Vision Transformers”, 29 Mart 2021, *arXiv*: arXiv:2103.15808. doi: 10.48550/arXiv.2103.15808.
- [141] X. Pan *vd.*, “On the Integration of Self-Attention and Convolution”.
- [142] M. Ding, B. Xiao, N. Codella, P. Luo, J. Wang, ve L. Yuan, “DaViT: Dual Attention Vision Transformers”, içinde *Computer Vision – ECCV 2022*, c. 13684, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, ve T. Hassner, Ed., içinde *Lecture Notes in Computer Science*, vol. 13684. , Cham: Springer Nature Switzerland, 2022, ss. 74-92. doi: 10.1007/978-3-031-20053-3_5.
- [143] I. Tolstikhin *vd.*, “MLP-Mixer: An all-MLP Architecture for Vision”, 11 Haziran 2021, *arXiv*: arXiv:2105.01601. doi: 10.48550/arXiv.2105.01601.
- [144] S. Kılıçarslan, “A novel nonlinear hybrid HardSReLU activation function in transfer learning architectures for hemorrhage classification”, *Multimed. Tools Appl.*, c. 82, sy 4, ss. 6345-6365, Şub. 2023, doi: 10.1007/s11042-022-14313-w.
- [145] S. Kılıçarslan ve S. Kılıçarslan, “A comparative study of bread wheat varieties identification on feature extraction, feature selection and machine learning algorithms”, *Eur. Food Res. Technol.*, c. 250, sy 1, ss. 135-149, Oca. 2024, doi: 10.1007/s00217-023-04372-0.
- [146] L. E. Sanchez, S. Moisan, ve J.-P. Rigault, “Metrics on feature models to optimize configuration adaptation at run time”, içinde *2013 1st International Workshop on Combining Modelling and Search-Based Software Engineering (CMSBSE)*, San Francisco, CA, USA: IEEE, May. 2013, ss. 39-44. doi: 10.1109/CMSBSE.2013.6604435.