



YAPRAK GÖRÜNTÜLERİNİ KULLANARAK DERİN ÖĞRENME İLE CEVİZ TÜRLERİNİN BELİRLENMESİ

Alper Talha KARADENİZ

**2023
DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ**

**Tez Danışmanı
Doç. Dr. Yüksel ÇELİK**

**YAPRAK GÖRÜNTÜLERİNİ KULLANARAK DERİN ÖĞRENME İLE
CEVİZ TÜRLERİNİN BELİRLENMESİ**

Alper Talha KARADENİZ

Tez Danışmanı

Doç. Dr. Yüksel ÇELİK

T.C.

Karabük Üniversitesi

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalında

Doktora Tezi

Olarak Hazırlanmıştır

KARABÜK

Şubat 2023

Alper Talha KARADENİZ tarafından hazırlanan “YAPRAK GÖRÜNTÜLERİNİ KULLANARAK DERİN ÖĞRENME İLE CEVİZ TÜRLERİNİN BELİRLENMESİ” başlıklı bu tezin Doktora Tezi olarak uygun olduğunu onaylarım.

Doç. Dr. Yüksel ÇELİK
Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Dr. Öğr. Üyesi Erdal BAŞARAN
İkinci Tez Danışmanı, Bilgisayar Programcılığı

Bu çalışma, jürimiz tarafından Oy Birliği ile Bilgisayar Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir. 24/02/2023

Ünvanı, Adı SOYADI (Kurumu) İmzası

Başkan : Prof. Dr. Mustafa Servet KIRAN (KTUN)

Üye : Prof. Dr. Oğuz FINDIK (KBÜ)

Üye : Doç. Dr. Yüksel ÇELİK (KBÜ)

Üye : Dr. Öğr. Üyesi Erdal BAŞARAN (AİÇÜ)

Üye : Dr. Öğr. Üyesi Kürşat Mustafa KARAOĞLAN (KBÜ)

Üye : Dr. Öğr. Üyesi Eyüp Burak CEYHAN (BÜ)

KBÜ Lisansüstü Eğitim Enstitüsü Yönetim Kurulu, bu tez ile, Doktora derecesini onamıştır.

Prof. Dr. Müslüm KUZU
Lisansüstü Eğitim Enstitüsü Müdürü



“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Alper Talha KARADENİZ

ÖZET

Doktora Tezi

YAPRAK GÖRÜNTÜLERİNİ KULLANARAK DERİN ÖĞRENME İLE CEVİZ TÜRLERİNİN BELİRLENMESİ

Alper Talha KARADENİZ

Karabük Üniversitesi

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:

Doç. Dr. Yüksel ÇELİK

Dr. Öğr. Üyesi Erdal BAŞARAN

Şubat 2023, 96 sayfa

Ceviz meyvesi cins, kalite olarak birbirinden farklılık göstermektedir ve geniş bir kullanım alanına sahiptir. İnsan sağlığı üzerine olumlu etkileri olan ceviz meyvesinin üretiminin devam etmesi için uygun ekolojik özelliğe sahip doğru ceviz çeşidinin seçilmesi oldukça önemlidir. Çünkü ceviz bahçesinin kurulması maliyetlidir ve hasat alma süresi de uzundur. Ancak birbirine renk ve doku olarak birbirine çok benzeyen ceviz yapraklarından çeşit ayrımının yapılması zordur. Uzmanlar tarafından ceviz çeşit ayrımının yapılması ciddi zaman gerektirir ve morfolojik testler yapılmalıdır. Ceviz çeşit ayrımı için literatürde yapılmış farklı çalışmalar mevcuttur. Ancak bu çalışmalar az sayıda ceviz çeşidinin sınıflandırılması veya laboratuvar deneyleri ile yapılan çalışmalardır. Günümüzde teknolojinin gelişmesi ile birlikte bilgisayar temelli derin öğrenme metotları yaprak tanımada sıklıkla kullanılmaktadır. Bu teknolojiler sayesinde zaman ve maliyet açısından kazanç sağlanmakta ayrıca hata oranı da

oldukça azalmaktadır. Bu tez çalışmasında öncelikle literatüre ceviz yapraklarından ceviz türlerinin belirlenmesi için 18 farklı çeşit cevizden alınan toplam 1751 yaprak görüntüsü ile özgün bir ceviz veri seti kazandırılmıştır. Oluşturulan veri setini otomatik olarak sınıflandırabilmek için farklı derin öğrenme modelleri önerilmiştir. Sınıflandırma başarısının ölçülmesi için yapılan deneysel testler hem özgün veri seti ile hemde yapılan ön işleme ve veri artırma yöntemleri ile oluşturulan veri setleri ile yapılmıştır. İlk deneysel testler literatürde yer alan popüler evrimsel sinir ağları ile yapılmıştır. Sonrasında bu sonuçlar önerilen derin öğrenme modellerinin test sonuçları ile karşılaştırılmıştır.

Önerilen ilk modelde arttırılmış veri setinin Vgg16 ile Gradcam görüntüleri üretildikten sonra yine Vgg16 ESA algoritması ile sınıflandırılmıştır. Deneysel testler sonucunda %77.11 başarı oranı elde edilmiştir. Önerilen bir sonraki modelde MobilnetV2, SequizeNet ve GLCM ile özellik çıkarılarak bir öznitelik havuzu oluşturulmuştur. Bu özniteliklerden en iyileri Ki-Kare yöntemi ile seçilmiş ve DVM ile sınıflandırılmıştır. Deneysel testler sonucunda %84.75 başarı oranı elde edilmiştir. Önerilen bir sonraki modelde residual blok tabanlı ResNet modelleri kullanılarak yeni bir derin öğrenme mimarisi hazırlanmıştır. Bu modelde ResNet18, ResNet50 ve ResNet101 modellerinden öznitelikler çıkarılmıştır. Bu özniteliklerden en iyileri Atom Arama Optimizasyon algoritması (AAO) ile seçilmiştir. Seçilen öznitelikler DVM' nin lineer, quadratik ve cubic çekirdekleri ile sınıflandırılmıştır. Deneysel testlerin güvenliğinin sağlanması için 5- parçalı doğrulama yapılmış ve %87.42 doğruluk sonucu elde edilmiştir. Önerilen son modelde %92.59 başarı oranına ulaşarak, tüm ESA algoritmaları ve önerilen modeller içinden en yüksek doğruluk sonucu elde edilmiştir. Bu tez çalışmasında, özgün bir ceviz veri seti literatüre kazandırılarak, yeni önerilen bir derin öğrenme modeli ile %92.59 doğruluk oranı ile sınıflandırılması sağlanmıştır.

Anahtar Sözcükler : Ceviz veri seti, makine öğrenmesi, derin öğrenme, özellik çıkarma, özellik seçme, sınıflandırma

Bilim Kodu : 92432

ABSTRACT

Ph. D. Thesis

DETERMINATION OF WALNUT SPECIES BY DEEP LEARNING USING LEAF IMAGES

Alper Talha KARADENİZ

**Karabuk University
Institute of Graduate Programs
Department of Computer Engineering**

Thesis Advisor:

**Assoc. Prof. Dr. Yüksel ÇELİK
Assist. Prof. Dr. Erdal BAŞARAN**

February 2023, 96 pages

Walnut fruit differs from each other in terms of genus and quality and has a wide range of uses. It is very important to choose the right walnut variety with suitable ecological characteristics in order to continue the production of walnut fruit, which has positive effects on human health. Because the establishment of a walnut garden is costly and the harvest time is long. However, it is difficult to distinguish between walnut leaves, which are very similar in color and texture. Differentiation of walnut varieties by experts requires serious time and morphological tests should be done. There are different studies in the literature for walnut variety differentiation. However, these studies are studies conducted with the classification of a small number of walnut varieties or laboratory experiments. Today, with the development of technology, computer-based deep learning methods are frequently used in leaf recognition. Thanks to these technologies, time and cost savings are achieved and the error rate is

considerably reduced. In this thesis, first of all, a unique walnut dataset was brought to the literature with a total of 1751 leaf images taken from 18 different types of walnuts in order to determine the walnut types from walnut leaves. Different deep learning models have been proposed to automatically classify the generated dataset. Experimental tests to measure classification success were carried out both with the original data set and with the data sets created by the preprocessing and data augmentation methods. The first experimental tests were made with popular convolutional neural networks in the literature. Afterwards, these results were compared with the test results of the proposed deep learning models.

In the first proposed model, Gradcam images of the augmented data set are produced with Vgg16 and classified with Vgg16 CNN algorithm. As a result of the experimental tests, the %77.11 success rate was obtained. In the next proposed model, a feature pool is created by extracting features with MobilnetV2, SequizeNet and GLCM. The best of these features were selected by Chi-Square method and classified by SVM. %84.75 success rate was obtained as a result of experimental tests. In the next proposed model, a new deep learning architecture is prepared using residual block-based ResNet models. In this model, features are extracted from ResNet18, ResNet50 and ResNet101 models. The best of these attributes were selected with the Atom Search Optimization algorithm (ASO). Selected features are classified with linear, quadratic and cubic kernels of SVM. In order to ensure the safety of the experimental tests, 5-part verification was performed and an accuracy of 87.42% was obtained. In the last model proposed, a success rate of 92.59% was achieved, and the highest accuracy result was obtained among all CNN algorithms and proposed models. In this thesis, a unique walnut data set was brought to the literature and it was classified with a newly proposed deep learning model with an accuracy rate of 92.59%.

Key Word : Walnut dataset, machine learning, deep learning, feature extraction, feature selection, classification

Science Code : 92432

TEŞEKKÜR

Bu tez çalışmasının planlanmasında, araştırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteğini esirgemeyen, engin bilgi ve tecrübelerinden yararlandığım, yönlendirme ve bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren sayın hocam Doç. Dr. Yüksel ÇELİK ve Dr. Öğr. Üyesi Erdal BAŞARAN' a sonsuz teşekkürlerimi sunarım.

Ayrıca tüm destekleri ve yönlendirmeleri için tez izleme komitesi üyelerinden Prof. Dr. Oğuz FINDIK ve Prof. Dr. Mustafa Servet KIRAN hocalarıma teşekkür ederim.

Veri setindeki yaprak çeşitlerinin belirlenmesi ve görüntülerin alınmasında ki yardımları için, babam Prof. Dr. Turan KARADENİZ, abla bildiğim Dr. Öğr. Üyesi Tuba BAK, ceviz görüntülerini Yalova Atatürk Bahçe Kültürleri Merkez Araştırma Enstitüsü uygulama bahçesinden almamıza izin veren Enstitüsü Müdürü Dr. Yılmaz BOZ ve Enstitü personeline teşekkür ederiz.

Tez yazım sürecinde ne zaman istesem yanımda olan kardeşim Muhammed Berker KARADENİZ, İsa PEKDEMİR, Said SADIKOĞLU ve mesai arkadaşım Aytaç Burak DERELİ'YE çok teşekkür ederim.

Sevgili annem Zekiye KARADENİZ ve babam Turan KARADENİZ' e maddi ve manevi hiçbir yardımı esirgemedi yanımda oldukları için tüm kalbimle teşekkür ederim.

Bu süreçte manevi desteklerini her zaman hissettiğim, çok fazla ihmal etsem de bana katlanan ve hep yanımda olan sevgili eşim Esra, kızım Zeynep Eliz ve oğlum Turan Aras'a çok teşekkür ediyorum.

İÇİNDEKİLER

	<u>Sayfa</u>
KABUL	ii
ÖZET	iv
ABSTRACT	vi
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ	xii
ÇİZELGELER DİZİNİ	xiv
SİMGELER VE KISALTMALAR DİZİNİ	xv
BÖLÜM 1	1
GİRİŞ	1
BÖLÜM 2	4
LİTERATÜR ARAŞTIRMASI	4
BÖLÜM 3	10
3.1. MAKİNA ÖĞRENMESİ	10
3.2. YAPAY SİNİR AĞI	10
3.3. DERİN ÖĞRENME	11
3.4. EVRİŞİMSEL SİNİR AĞLARI	12
3.4.1. Giriş Katmanı	13
3.4.2. Evrişim Katmanı (Convolution)	13
3.4.3. Havuzlama Katmanı (Pooling)	15
3.4.4. Tam Bağlantılı Katman (Fully Connected Layer)	15
3.4.5. ESA'nın Eğitilmesinde Kullanılan Diğer Yöntemler Ve Parametreler..	16
3.4.5.1. Aktivasyon Fonksiyonu	16
3.4.5.2. Seyretilme Katmanı (Dropout)	19
3.4.5.3. Mini-Batch Değeri	19
3.4.5.4. Epoch ve Döngü Sayısı	20
3.4.5.5. Öğrenme Katsayısı	20

	<u>Sayfa</u>
3.4.5.6. Optimizasyon	20
3.4.6. ESA Algoritmaları	24
3.4.6.1. AlexNet	24
3.4.6.2. InceptionV3.....	25
3.4.6.3. Vgg16.....	25
3.4.6.4. Vgg19.....	26
3.4.6.5. EfficientNet.....	27
3.4.6.6. DarkNet19.....	27
3.4.6.7. GoogleNet	28
3.4.6.8. ResNet.....	29
3.4.6.9. MobileNetV2	30
3.4.6.10. SqueezeNet	31
3.5. VERİ SETİ	32
3.6. ÖN İŞLEME.....	33
3.7. VERİ ARTIRMA	35
3.8. GRADYAN AĞIRLIKLIL SINIF AKTİVASYON EŞLEMESİ (GRADIENT-WEIGHTED CLASS ACTIVATION MAPPING).....	35
3.9. GRİ SEVİYELİ EŞ-OLUŞUM MATRİSİ (GRAY LEVEL CO- OCCURRENCE MATRIX)	36
3.10. ATOM ARAMA OPTİMİZASYONU	38
3.11. BALİNA OPTİMİZASYON ALGORİTMASI	39
3.11.1. Avın Etrafını Sarma.....	40
3.11.2. Ava Hareket Edilmesi.....	40
3.11.3. Av Arama	41
3.12. Kİ- KARE TESTİ İLE ÖZELLİK SEÇME	42
3.13. DESTEK VEKTÖR MAKİNELERİ.....	43
3.14. EN YAKIN KOMŞU ALGORİTMASI.....	44
3.15. PERFORMANS ÖLÇÜTLERİ	45
3.16. APRAZ DOĞRULAMA.....	47
 BÖLÜM 4	 48
ÖNERİLEN DERİN ÖĞRENME MODELLERİ.....	48
4.1. ESA MİMARİLERİ İLE SINIFLANDIRILMA (ESA-9)	48

4.2. GRAD-CAM TABANLI GÖRÜNTÜLERİN VGG16 ESA MODELİ İLE GÖRSELLEŞTİRİLEREK SINIFLANDIRILMASI (GRADCAM-VGG16).....	49
4.3. GLCM, HAFİF AĞIRLIKLIL DERİN ÖĞRENME MODELLERİ ve Kİ-KARE TABANLI MODEL ÖNERİSİ (GLCM-KİKARE-ESA).....	50
4.4. CEVİZ ÇEŞİTLERİNİN ÖNERİLEN ARTIK BLOK TABANLI ESA MODELİ İLE SINIFLANDIRILMASI (RES-AAO)	52
4.5. BALİNA OPTİMİZASYON ALGORİTMASI İLE ESA ÖZELLİKLERİNİN SEÇİLEREK CEVİZ VERİ SETİNİN SINIFLANDIRILMASI (BOA-ESA-KNN).....	55
 BÖLÜM 5	 59
DENEYSEL ÇALIŞMALAR VE BULGULAR.....	59
5.1. ESA-9 MODELİNİN DENEYSEL TEST SONUÇLARI	59
5.2. GRADCAM-VGG16 MODELİNİN DENEYSEL TEST SONUÇLARI.....	67
5.3. GLCM-KİKARE-ESA MODELİNİN DENEYSEL TEST SONUÇLARI ...	69
5.4. RES_AAO MODELİNİN DENEYSEL TEST SONUÇLARI	72
5.5. BOA-ESA-KNN MODELİNİN DENEYSEL TEST SONUÇLARI.....	77
 BÖLÜM 6	 80
TARTIŞMA	80
 BÖLÜM 7	 83
SONUÇLAR VE ÖNERİLER.....	83
 KAYNAKLAR	 86
 ÖZGEÇMİŞ	 96

ŞEKİLLER DİZİNİ

Sayfa

Şekil 3.1. YSA ve biyolojik sinir hücresi yapısı	11
Şekil 3.2. YSA çalışma şekli.	11
Şekil 3.3. ESA genel mimarisi.	13
Şekil 3.4. Evrişim katmanı yapısı.....	14
Şekil 3.5. Havuzlama katmanı gerçekleştirme yapısı.	15
Şekil 3.6. Aktivasyon fonksiyonu yapısı.....	17
Şekil 3.7. Sigmoid grafiği.....	17
Şekil 3.8. ReLu grafiği.	18
Şekil 3.9. Softmax fonksiyonu grafiği.....	19
Şekil 3.10. Dropout işlemi	19
Şekil 3.11. Momentum kullanılmadan gerçekleşen salınım.	22
Şekil 3.12. Momentum kullanılarak gerçekleşen salınım.	22
Şekil 3.13. AlexNet genel mimarisi	25
Şekil 3.14. InceptionV3 genel mimarisi.....	25
Şekil 3.15. VGG16 genel mimarisi	26
Şekil 3.16. Vgg19 genel mimarisi.....	26
Şekil 3.17. EfficientNet genel mimarisi	27
Şekil 3.18. DarkNet19 genel mimarisi	28
Şekil 3.19. GoogleNet genel mimarisi	28
Şekil 3.20. ResNet genel mimarisi.....	30
Şekil 3.21. MobilnetV2 mimarisi.....	31
Şekil 3.22. Squeezenet mimarisi	31
Şekil 3.23. Toplanan veri setinde ki 18 çeşit ceviz yaprağı görüntü örnekleri.	32
Şekil 3.24. Verinin ön işleme aşamaları.....	34
Şekil 3.25. Grad-Cam çalışma mimarisi	36
Şekil 3.26. GLCM'nin görüntü üzerinde oluşturulma örneği	37
Şekil 3.27. Kambur balina avlanma şekli.....	39
Şekil 3.28. Spiral hareket.	41
Şekil 3.29. DVM ile sınıflandırma örneği.....	44
Şekil 3.30. KNN sınıflandırma örneği.	45

Şekil 3.31. 5-parçalı doğrulama çalışma örneği.....	47
Şekil 4.1. ESA-9'un akış şeması.	48
Şekil 4.2. Ceviz yapraklarının Grad-Cam görüntü örnekleri.	49
Şekil 4.3. GRADCAM-VGG16 modeli akış şeması.....	50
Şekil 4.4. GLCM-KİKARE-ESA modelinin şeması.....	51
Şekil 4.5. RES-AAO modelinin çalışma şeması.	53
Şekil 4.6. AAO sözde kodu.....	53
Şekil 4.7. BOA ile özellik seçiminin akış diyagramı.	56
Şekil 4.8. BOA'nın sözde kodu.....	57
Şekil 4.9. BOA-ESA-KNN modelin çalışma aşamaları.....	58
Şekil 5.1. ESA-9 çalışmasının test sonuçlarının doğruluk oranlarının grafikleri.....	60
Şekil 5.2. Vgg16 modeli ile sınıflandırma sonucu elde edilen karmaşıklık matrisi. 61	
Şekil 5.3. Vgg19 modeli ile sınıflandırma sonucu elde edilen karmaşıklık matrisi. 62	
Şekil 5.4. AlexNet modeli ile sınıflandırma sonucu elde edilen karmaşıklık matrisi.....	63
Şekil 5.5. Arttırılmış veri seti ile ESA-9 modelinin test sonuçlarının doğruluk oranlarının grafikleri.	64
Şekil 5.6. Arttırılmış veri setinin Vgg16 karışıklık matrisi.....	65
Şekil 5.7. Arttırılmış veri setinin Vgg19 karışıklık matrisi.....	66
Şekil 5.8. Arttırılmış veri setinin DarkNet karışıklık matrisi.	67
Şekil 5.9. GRADCAM-VGG16 modelinin test sonuçları.....	68
Şekil 5.10. GRADCAM-VGG16 modeli karışıklık matrisi.....	69
Şekil 5.11. GLCM-KİKARE-ESA karmaşıklık matrisi.....	72
Şekil 5.12. Resnet modellerinden AAO ile seçilen özellik sayısı.....	74
Şekil 5.13. RES_AAO modelinin karışıklık matrisi.	76
Şekil 5.14. RES_AAO modelinin ROC eğrisi.	76
Şekil 5.15. AAO'nun yakınsama grafiği.....	77
Şekil 5.16. BOA-ESA-KNN modeli karmaşıklık matrisi.	78

ÇİZELGELER DİZİNİ

Sayfa

Çizelge 3.1. Toplanan veri setindeki 18 Ceviz çeşidine karşılık gelen sayılar.	32
Çizelge 3.2. Arttırılmış veri seti görüntü sayıları.....	35
Çizelge 4.1. BOA özellik seçiminde kullanılan parametreler.....	55
Çizelge 5.1. ESA-9 modelinin deneysel testlerinde kullanılan hiper-parametreler.	59
Çizelge 5.2. ESA-9 çalışmasının deneysel test sonuçları.	60
Çizelge 5.3. Arttırılmış veri seti ile ESA-9 modelinin deneysel test sonuçları.....	63
Çizelge 5.4. GRADCAM-VGG16 test sonuçları.....	69
Çizelge 5.5. MobilNetV2 ve SqueezeNet' in deneysel test sonuçları.....	70
Çizelge 5.6. GLCM-KİKARE-ESA kullanılan hiper-parametreler.....	70
Çizelge 5.7. GLCM-KİKARE-ESA çıkarılan özelliklerin deneysel test sonuçları.	70
Çizelge 5.8. GLCM-KİKARE-ESA öznelik havuzu deneysel test sonuçları.....	71
Çizelge 5.9. GLCM-KİKARE-ESA deneysel test sonuçları.	71
Çizelge 5.10. ResNet modellerinin performans sonuçları.	73
Çizelge 5.11. ResNet metotlarının DVM ile birlikte sınıflandırma performansı.....	73
Çizelge 5.12. AAO kullanarak sınıflandırmanın performans sonuçları.....	74
Çizelge 5.13. Tüm ESA özelliklerine AAO uygulanarak oluşturulan niteliklerin sınıflandırılması.	75
Çizelge 5.14. RES_AAO modelinin deneysel test sonuçları.	75
Çizelge 5.15. BOA-ESA-KNN modeli KNN deneysel test sonuçları.	77
Çizelge 5.16. BOA-ESA-KNN modeli deneysel test sonuçları.....	78
Çizelge 6.1. Veri seti ile yapılan deneysel çalışmalar.....	81

SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

x	: giriş değeri
y	: çıkış değeri
W, R	: ağırlık
b	: bias
c	: hücre değeri
i	: giriş kapısı
z	: güncelleme değeri
σ	: sigmoid aktivasyon fonksiyonu
g, h	: aktivasyon fonksiyonu
t	: zaman
f	: giriş katmanından gelen değer
j	: satır
k	: sütun
o	: çıkış görüntü boyutu
β	: hiper parameter

KISALTMALAR

AAO	: Atom Arama Optimizasyonu (Atom Search Optimization)
ADAM	: Adaptive Moment (Uyarlanabilir Moment Tahmini)
BOA	: Balina Optimizasyon Algoritması (Whale Optimization Algorithm)
ESA	: Evrişimli Sinir Ağı (Convolutional Neural Network)
DSA	: Derin Sinir Ağları (Deep Neural Network)
DN	: Doğru Negatif (True Negative)
DP	: Doğru Pozitif (True Positive)
DVM	: Destek Vektör Makinaları (Support Vektor Machine)
GD	: Gradient Descent (Gradyan İniş)
GLCM	: Gray Level Co-Occurance Matrix (Gri Seviyeli Eş-Oluşum Matrisi)
GRAD-CAM	: Gradyan Ağırlıklı Sınıf Aktivasyon Eşlemesi (Gradient-Weighted Class Activation Mapping)
KNN	: K-Nearest Neighbors (K-En Yakın Komşu)
LDA	: Linear Discriminant Analysis (Doğrusal Ayırma Analizi)
LSTM	: Long Short-Term Memory (Uzun Kısa Süreli Bellek)
MAX	: Maksimum (En Yüksek)
MIN	: Minimum (En Düşük)
RELU	: Rectified Linear Unit (Doğrultulmuş Doğrusal Birim)
RGB	: Red Green Blue (Kırmızı Yeşil Mavi)
ROC	: Receiver Operating Characteristic Curve (Alıcı İşlem Karakteristiği)
YP	: Yanlış Pozitif (False Positive)
YN	: Yanlış Negatif (False Negative)
YSA	: Yapay Sinir Ağları (Artificial Neural Networks)

BÖLÜM 1

GİRİŞ

Bitkilerin tanınması insanoğlunun geleceği açısından oldukça önemlidir. Bitkiler, dünyamız için en önemli kaynaklardan birisidir ve bu kaynakların sağlıklı şekilde geleceğe taşınması gerekmektedir [1]. Artan küresel nüfus ve iklim değişikliğinin getirdiği zorluklar nedeniyle gıda mahsullerine yönelik artan taleple birlikte, minimum maliyetle ve minimum zaman harcayarak tarımsal besinlerin arttırılmasına yönelik acil bir ihtiyaç vardır. Kaynakların etkin kullanılabilmesi için, uygun genotipli bitkilerin seçilmesi, bitki üretkenliği ve verimliliğini arttırmaya yardımcı olacaktır.

Özellikle ceviz yetiştiriciliğinde üreticilerin doğru ceviz fidanı alması, ceviz yetiştirme açısından oldukça önemlidir. Çünkü üretici satın almış olduğu fidanlarla bahçe kurduktan sonra, fidanlarda meyvenin görülmesi 3-5 yıl gibi bir süre almaktadır. Bu süre zarfında üretici, arzu etmediği çeşit veya çöğür (yabani) ağaçlarına 3-5 yıl boyunca sulama, gübreleme, budama ve hastalık ile mücadele yapmaktadır. Bu durum maddi ve zaman açısından ciddi maliyet gerektirmektedir. Uygun olmayan bu çeşitleri çevirme aşıları ile ya çeşit değiştirme yoluna gitmekte ya da bu ağaçları sökmekte, bu işlemler için de oldukça parasal kaynak harcamaya mecbur kalmaktadır. Bununla birlikte, isimleri belli olmayan, karışık çeşitlerle bahçe tesis edilmiş ise, hasat zamanında meyveler karışık olarak toplanacağından, meyveler çeşitlere göre bir sınıflandırma yapılamayacağından, pazarda meyve ağırlığı, randımanı, kalibresi belli olan meyve satın almak isteyen tüketicinin bu isteği karşılanamayacaktır. Aynı şekilde bu tür yetiştiricilik endüstriyel işlemeye de uygun olmayacak, meyvenin pazar değeri oldukça düşecektir. Zira endüstriyel işlemede ceviz meyvesinin kalibresi standart olduğunda sert kabuğun makine ile kırılması ve iç meyvenin işlenmesi, iç meyvenin zayı olmasının önüne geçmekte ve bu durum endüstride arzu edilen ve istenen bir özelliktir. Dolayısıyla, en az 700 yıl ürün vererek yaşayan ceviz bahçesinin tesislerinde, hangi çeşidin dikileceği, fidanın ismine doğru olarak seçilmesi çok

önemlidir. Bunun için, ceviz fidanları genel olarak Ağustos sonu Eylül başlarında aşılanmakta, aşı gözü anaca kaynaşmakta fakat sürmemektedir. Takip eden Nisan sonu Mayıs başlarında süren aşı gözlerinden çıkan yapraklarda çeşit tanımlanması yapılarak çeşit karışıklığının ya da fidanların çöğür (yabani) olup olmadıkları konusunda bilgi sahibi olunması, önemli bir hukuki sürecin önceden önlenmesi anlamına gelmektedir [2]. Ceviz yetiştiriciliği tarihte çok eski yıllara dayanmaktadır, ceviz çeşit ayrımı problemlerinin çözülebilmesi ve standartlaştırma yapılabilmesi için literatürde yapılan birçok çalışma vardır [3].

Ceviz yapraklarından sınıflandırma işleminin gerçekleştirilebilmesi, her yaprağın ayrıntılı incelenmesi sonucunda yapılabilmektedir. Birbirine renk ve şekil olarak benzeyen türleri insan gözüyle ayırt etmek oldukça zordur. Çok fazla türe ait yaprak görüntülerinin olduğu bitki çeşitlerini sınıflandırılması için, her bir yaprak ayrıntılı olarak incelenmelidir. Bu süreç, zaman ve maliyet açısından uygun olmayan bir durumdur. Bunun için yaprak görüntülerinin alınarak bilgisayar ortamında otomatik olarak teşhis edilmesi ve sınıflandırmanın yapılması zaman ve maliyet açısından oldukça kolaylık sağlamaktadır [4]. Sınıflandırma işlemleri için, yapay zeka teknikleri kullanılarak insan gözünün yaptığı işlevlerden daha hızlı ve daha doğru sonuçlar vermeye yarayan birçok yazılımsal çalışma bulunmaktadır [5].

Her yıl fidancılarla fidan satın alan üreticiler arasında çok sayıda, ismine doğru olmadığı gerekçesiyle hukuki açıdan davalar açılmaktadır. Fidanın satın alınması, dikimi, meyveye yatıncaya kadar geçen süre, uygun iklime dikilmeyen fidanların kaybı ve mahkeme süreçleri gibi istenmeyen birçok husus bulunmaktadır.

Tezin bundan sonraki ikinci bölümde, geniş bir literatür özeti ile hem benzer veri setleri hem de yapılan benzer çalışmalardan bahsedilmiştir. Burada yer alan benzer çalışmalarda kullanılan teknolojiler ve başarı sonuçları da verilmiştir.

Üçüncü bölüm “Materyal Metot” bölümüdür. Bu bölümde oluşturulan özgün veri seti açıklanmaktadır. Ayrıca, veri setine uygulanan ön işleme ve veri artırma yöntemlerinden bahsedilerek bu işlemler sonucunda oluşan yeni veri setine değinilmiştir. Veri seti dışında, çalışmada kullanılan yapay zeka ve optimizasyon

teknikleri, görüntü işleme teknikleri, sınıflama metotları, derin öğrenmede kullanılan performans ölçütleri ayrıntılı olarak anlatılmıştır.

Dördüncü bölümde geliştirilen yeni derin öğrenme metotları ayrıntılı olarak açıklanmıştır. İlk olarak literatürde yaygın olarak kullanılan derin öğrenme metotları kullanılmıştır. Sonrasında derin öğrenme, optimizasyon, görüntü işleme, sınıflandırma teknikleri birleştirilerek dört farklı özgün model önerilmiştir.

Beşinci bölüm “Deneyisel Çalışmalar ve Bulgular” bölümüdür. Burada geliştirdiğimiz tüm modellerin özgün veri seti üzerine uygulanması sonucu elde edilen deneyisel testler ayrıntılı olarak açıklanmış ve deneyisel test sonuçları verilmiştir.

Altıncı bölüm olan “Tartışma” kısmında geliştirilen tüm modellerin başarıları, karşılaştırılmış ve yorumlanmıştır.

Son olarak Yedinci bölüm olan “Sonuç ve Öneriler” kısmında tez çalışmasının bilime katkısı, literatüre kazandırdıkları ve çalışmanın başarısı anlatılarak tez çalışması sonlandırılmıştır.

BÖLÜM 2

LİTERATÜR ARAŞTIRMASI

Merkezi İşlem Birimleri (MİB) ve Grafik İşlem Birimlerinin (GİB) artan kapasite ve hızları, ham verileri el yapımı özelliklere ihtiyaç duymadan işleyebilen yüksek performanslı yeni yöntemlerin geliştirilmesine yol açmış ve bu da derin öğrenme mimarilerinin ortaya çıkmasına neden olmuştur [6]. 2019 yılına ait bir derleme makalesinde, bitki yaprak sınıflandırma ve hastalık tespitine yönelik derin öğrenme temelli çalışmalar incelenmiş ve potansiyelinin yüksek olduğu görülmüştür [7]. Bu gelişmeler ışığında literatürde bitki yapraklarından sınıflandırma yapılan birçok derin öğrenme çalışması bulunmaktadır.

Evrişimsel Sinir Ağı (ESA) ve Transfer Öğrenmenin yaprak görüntüsünden sınıflandırma yapılması üzerindeki etkisini gösterebilmek için yeni bir yöntem geliştirmiştir. Oxford Flowers veri seti kullanılarak yapılan çalışmada ImageNet etki alanından öğrenmeyi aktarmak için ince ayar yaklaşımı kullanılmıştır. Transfer öğrenme için, ImageNet veri tabanı üzerinde eğitilmiş MobileNetV2 kullanılmış ve %98,97 doğruluk elde edilmiştir. Ayrıca mobil uygulaması yazılarak olumlu sonuçlar alınmıştır [8]. Beş farklı kuruyemişi sınıflandırabilmek için, 2868 görüntü içeren bir veri seti oluşturulmuştur. Oluşturulan modelde, ilk katman her biri Relu aktivasyon fonksiyonuna sahip 4 evrişim katmanından oluşur. Bunu Max Pooling katmanı izler. Düzleştirme katmanından sonraki ikinci kısım iki yoğun katman içerir, ilki 512 gizli katman içerir. Toplam ağ eğitilebilir parametre sayısı 2.603.205' tür ve son katman aktivasyon fonksiyonu olarak softmax kullanılmaktadır. Yapılan çalışmada %98 doğruluk oranı elde edilmiştir [9]. 21 çeşit üzüm yaprağını sınıflandırabilmek için, ön işleme ve ESA algoritmaları kullanılmıştır. Görüntü tamamlayıcı ön işlemenin sınıflandırma sonuçları üzerindeki etkisini analiz etmek için Gradyan Ağırlıklı Sınıf Etkinleştirme Eşlemesi (Grad-CAM) algoritması geliştirilmiştir. Googlenet modeli kullanılarak yapılan testler sonucunda %97.4 doğruluk oranı bulunmuştur [8]. 12 farklı

türde ki bitki fidelerinin sınıflandırılması için K- Yakın Komşu Algoritması (KNN), Destek Vektör Makinaları (DVM) ve özgün bir derin öğrenme modeli geliştirilmiştir. Yapılan testler sonucunda %92.6 doğruluk oranı elde edilmiştir [10]. Bitki sınıflandırılmasının doğal ortamda ve büyük ölçekli yapılabilmesi için 8 yapı taşından oluşan 26 katmanlı bir derin öğrenme modeli önerilmiştir. Önerilen model, BJFU100 veri setinde %91,78'lik bir tanıma oranı elde ederek, derin öğrenmenin akıllı ormancılık için umut verici bir teknoloji olduğunu göstermektedir [11]. Farklı bitki türlerinin tespiti için ESA ve DVM kullanılarak yeni bir yöntem önerilmiştir. Yaprak görüntüleri LifeCLEF2015 [12] veri setinden alınmıştır. Özellik çıkarmak için 16 evrişim katmanı ve 3 tam bağlı katmandan oluşan bir ESA modeli kullanılmıştır. Çıkarılan özellikler doğrusal DVM ile sınıflandırılmıştır. Yapılan deneysel testler sonucunda klasik ESA modellerinden daha iyi sonuçlar verdiği görülmektedir [13]. 5 türe ait 500 adet asma yaprağı görüntüsü özel ortamda toplanarak veri artırma yöntemiyle 2500'e çıkarılmış ve ESA tabanlı modeller ile sınıflandırılmıştır. Sınıflandırma işlemi için ilk olarak MobilNetv2 kullanılmış. İkinci aşamada MobilNetv2 ile özellikler çıkarılarak bu özellikler DVM ile sınıflandırılmıştır. Son aşamada ise çıkarılan özellikler Ki-Kareler yöntemiyle seçilerek 250'ye düşürülmüştür. Ardından seçilen öznelikler kullanılarak çeşitli DVM çekirdekleri ile sınıflandırma yapılmıştır. En başarılı yöntem, özneliğin Ki-Kareler yöntemiyle indirgenmesiyle elde edilmiştir. En başarılı DVM çekirdeği %97.60 doğruluk oranıyla Cubic DVM'dir [14]. Bitki yaprak görüntülerinden sınıflandırma yapılırken, botanikçinin ayrımını taklit eden ESA tabanlı yeni bir model önerilmiştir. Bu model üç aşamadan oluşmaktadır. İlk olarak yaprağın şeklinden ayrım yapabilen bir ESA modeli tasarlanarak kolay ayrım yapılabilen türlerin ayrımı bu şekilde yapılmıştır. İlk aşamada sınıflandırma yapılamıyorsa, ilk aşamadaki şekil özelliğinin yanı sıra renk, doku gibi özelliklerle sınıflandırma yapılmaktadır. Sınıflandırma işlemi gerçekleşmeyen türler için yaprağın damarlanma yapısının tespiti için daha ayrıntılı incelemede bulunan bir ESA modeli tasarlanmıştır. Üçüncü aşamada da sınıflandırma yapılamazsa en benzer olan sınıf listesinden birisi seçilmektedir. Önerilen metottun eğitilmesi ve deneysel testleri için MalayaKew ve Flavia veri setleri kullanılmıştır. Deneysel test sonuçlarına göre önerilen yöntem %99.81 ve %99.67 doğruluk elde etmiş ve diğer tüm yöntemlerden daha iyi performans göstermiştir [15]. Yaprak görüntülerinden bitki tanıma için hiyerarşik bir mimari tasarım önerilmiştir. Bu

tasarımda yaprakların farklı özellikleri için ayrı ayrı yöntemler kullanılmıştır. Yaprakların farklı özelliklerinin tespiti için veri seti toplanırken benzer şekilde ancak farklı renkte olan yaprak türleri de veri setine eklenmiştir. Sınıflandırma işleminde Faster RESA ve InceptionV2 tabanlı bir model tasarlanmıştır. Özellik haritalarını oluştururken bölge tespiti için RPN uygulanmıştır. Deneysel testler farklı şekiller ve farklı renkler olmak üzere de iki farklı aşamada değerlendirilmiştir. Yapılan testler sonucunda önerilen modelin geleneksel ESA modellerinden daha başarılı olduğu görülmüştür [16]. Yaprak görüntülerinden bitki sınıflandırma yapılırken özellik çıkarma ve özellik seçme işlemleri oldukça önemlidir. ESA metotları özellik çıkarma işlemini kendisi yapmaktadır ancak bazı çalışmalarda ESA metotlarının çıkardığı özellikler yerinde farklı güçlü özellikler kullanılabilmektedir. İngiltere Kraliyet Botanik Bahçelerinden toplanan 44 farklı çeşitteki bitki türleri üzerinde yapılan sınıflandırma çalışmasında yapraktaki damarlanma yapısının güçlü özellik olduğu bulunmuştur [17]. Farklı renk ve şekillerdeki 5 farklı tür yaprak çeşidinin sınıflandırılması için InceptionV2 tabanlı yeni bir derin öğrenme yöntemi önerilmiştir. Bölge önerilerini tahmin etmek ve özellik haritası oluşturmak için RPN yöntemi, özellik çıkarma için InceptionV2 yöntemi, Havuzlama katmanı için ROI yöntemi ve sınıflandırma için ise softmax fonksiyonu kullanılmıştır. Sınıflandırma işleminin başarı sonuçlarının değerlendirilebilmesi için deneysel test sonuçları Faster ESA ile karşılaştırılmış ve önerilen modelin daha iyi sonuç verdiği görülmüştür [18]. Elma, Armut, Narenciye gibi 10 farklı meyvenin sınıflandırması için yeni bir yöntem önerilmiştir. 6847 görüntü içeren bir veri kümesi, eğitim için 4793 görüntü, doğrulama için 1027 görüntü ve test için 1027 görüntü olarak bölünmüştür. Görüntü tanıma işlemi için yaygın olarak uygulanan bir derin öğrenme tekniği olan Vgg16 kullanılmıştır. Deneysel testler sonucunda, arttırılmış test setinde %100 doğruluk elde ettiği ve uygulanabilir bir yaklaşım olduğu görülmüştür [19]. 15 farklı ağaçtan alınan yaprak görüntülerinden oluşan İsveç veri seti üzerinde, otomatik yaprak tanıma için önerilmiş ESA modeli çalışması yapılmıştır. Yapılan çalışmada ön işleme, özellik çıkarma ve sınıflandırma adımları uygulanmıştır. Deneysel çalışmalar sonucunda %93.75 doğruluk sonucuna ulaşılmıştır [20]. Akasya, papaya, kiraz, mango ve rambutan dan oluşan beş tür yerel Malezya yaprağının sınıflandırılması için Resnet50 tabanlı bir derin öğrenme yaklaşımı önerilmiştir. Veri setine gerekli ön işleme aşamaları uygulandıktan sonra Resnet50 ile eğitim yapılmış ve sistemin kullanılabilmesi için bir

ara yüz geliştirilmiştir. Yapılan deneysel testler sonucunda %98 doğruluk oranına ulaşılmıştır [21].

Literatürde bitki tanımanın yanı sıra yaprak görüntülerinden hastalık teşhisi çalışmaları da son yıllarda oldukça popüler hale gelmiştir.

EfficientNet ve transfer öğrenmesi bitki yapraklarından hastalık teşhisi için kullanılmıştır. PlantVillage veri seti kullanarak EfficientNet derin öğrenme mimarisin önerilmiş ve bu modelin performansı diğer son teknoloji derin öğrenme modelleri ile karşılaştırılmıştır. Transfer öğrenmede modellerin tüm katmanları eğitilebilir olacak şekilde ayarlanmıştır. Yapılan testler sonucunda %99.9 doğruluk oranı elde edilmiştir [22]. PlantVillage veri setini kullanarak yapılan bir başka yaprak hastalık tespiti çalışmasında, dört farklı ESA modeli uygulanmıştır. EfficientNetB0 modelinde en iyi doğruluk oranı olan %99,56'ya ulaşılmıştır [23]. Hindistan alt kıtasında yetişen bir baklagil bitkisi olan Vigna mungo için otomatik derin öğrenmeye dayalı viral enfeksiyon tespit yöntemi önerilmiştir. ESA mimarisine dayalı olan VirLeafNet, çoklu dönemler için sağlıklı, hafif enfeksiyonlu ve ciddi şekilde enfekte olmuş yapraklardan oluşan farklı yaprak görüntüleri ile eğitilmiştir. Önerilen modellerin test sonuçları, VirLeafNet-1 %91.2, VirLeafNet-2 %96.4 ve VirLeafNet-3 %97.4 olarak bulunmuştur [24]. Kahve ağacındaki lezyonların tanınmasını otomatikleştirmek için, kahve ağacının bir kısmını içeren görüntüler, akıllı telefon aracılığıyla toplanmış ve farklı ESA modellerini kullanarak entegre bir çalışma önerilmiştir. İlk aşamada, örnek segmentasyonu için bir Maske Bölge Tabanlı ESA (Mask R-ESA) ağı kullanılmış, ikinci aşamada, anlamsal bölümler için UNet ve PSPNet ağı ve son olarak, sınıflandırma için ResNet uygulanmıştır. Yapılan test sonucu %94.2 olarak bulunmuştur [25]. Elma yaprağı hastalıklarından en yaygın görülen dört farklı türün otomatik tespiti için ESA tabanlı bir model önerilmiştir. Önerilen modelde hastalık tespiti için AlexNet tabanlı yeni bir mimari önerilmiştir. Veri seti için 13.689 adet hastalıklı elma yaprağı görüntüsü kullanılmıştır. Yapılan testler sonucunda %97,62'lik bir doğruluk oranı ile orijinal AlexNet'ten %10,83 daha iyi sonuç vermiştir [26]. PlantVillage veri setinden alınan 10 sınıf domates yaprak görüntüsünden hastalık tespiti yapılabilmesi için yeni bir ESA modeli önerilmiştir. Önerilen modelin başarısı VggNet, ShuffleNet ve SqueezeNet ile karşılaştırılmış ve rekabetçi sonuçlar alınmıştır.

Yapılan deneysel testler sonucunda önerilen modelin doğruluk oranı %97.06 olarak bulunmuştur [27]. PlantVillage veri setinden üretilen başka bir çalışmada, patates, domates ve biber görüntülerinden oluşan toplam 20639 görüntünün içinden hastalıklı olanların tespiti için otomatik bir sistem geliştirilmiştir. Veri setine ön işleme işlemleri uygulandıktan sonra Bayesian öğrenme süreci kullanılarak bir ESA modeli önerilmiş ve deneysel testleri yapılmıştır. Test sonucunda %98.9 doğruluk oranına ulaşılmıştır [28]. Çeşitli açık veri kütüphanelerinden oluşturulan yeni bir veri seti üzerinde yer alan yaprak görüntüleri üzerinde hastalık tespiti yapılabilmesi için 14 katmanlı yeni bir ESA modeli önerilmiştir. Veri seti üzerinde çeşitli veri büyütme teknikleri uygulanmış ve toplam görüntü sayısı 147500 olarak eğitim işlemleri yapılmıştır. Yapılan testler sonucunda %99.8 doğruluk oranına ulaşılmıştır [29].

Literatürde, ceviz bahçelerindeki hastalık tespiti ile ilgili çalışmalar da yer almaktadır. Ceviz hastalık tespitinin otomatikleştirilmesi için makine öğrenme yöntemleri kullanan çalışmalardan birisi Khan vd. tarafından önerilmiştir. Önerilen modelde yaprak görüntüsü belli ön işlemlere tabi tutulduktan sonra Otsu thresholding algoritması ile segmentlere ayrılır. GLCM yöntemi ile öznelilikleri ayarlandıktan sonra sinir ağı ile sınıflandırma işlemi yapılır. Yapılan testlerin başarı sonucu %95.3 olarak bulunmuştur [30]. Derin öğrenme yöntemlerini kullanarak hastalık bulaşmış yaprakları tespit eden bir diğer çalışma, gerçek zamanlı olarak ceviz bahçesindeki yapraklara direk uygulanmaktadır. Bu işlem için tek atış detektörü (SSD) algoritması kullanılarak nesne tespiti yapılmış ve derin öğrenme modeli eğitilmiştir [31]. Bir başka çalışma da ise, yaprak görüntüleri alınarak ağacın Antraknoz mantar hastalığına yakalanıp yakalanmadığını ESA modeli ile tespit etmeye çalışmışlardır. Gri tonlamalı, Kırmızı-Yeşil-Mavi (Red-Green-Blue, RGB) modunda bir dizi görüntü kullanılmış, özellik çıkarımı için hızlı bir Fourier dönüşümü uygulanmış ve performansına göre en iyi ESA mimarisi seçilmiştir. Önerilen model ile literatürde yer alan ESA modellerinin başarı performansları kıyaslanmıştır. Modellere uygulanan deneysel testler de başarı oranları %92.4 ile %98.7 arasında değişmektedir [32].

Bu tez çalışmasının motivasyonu olan ceviz çeşitlerinin sınıflandırılması ile ilgili literatürde kısıtlı sayıda çalışma bulunmaktadır. Ceviz sınıflandırılması için yapılan bir çalışmada Sangi ve Kaghazi adında iki farklı İran ceviz çeşidi kullanılmıştır. Bu

sınıflandırma işlemi için toplam 4000 farklı cevizden ses sinyalleri alınmış ve bu sinyaller Yapay Sinir Ağlarına (YSA) verilerek sınıflandırma işlemi yapılmıştır. Çalışmanın deneysel testleri sonucunda %96.56 doğruluk oranına ulaşılmıştır [33]. İran ceviz çeşitlerinin sınıflandırılması için yapılan bir başka çalışma da İran'ın altı bölgesinden 2013-2014 döneminde hasat edilen ceviz örnekleri toplanmıştır. Ceviz orijinini ayırt etmek için ceviz yağının kromatografik parmak izleri kullanılmıştır. Temel bileşen analizi ve Doğrusal diskriminant analizi ile deneysel çalışmalar sonuçları, yağ asidi parmak izlerine dayalı olarak coğrafi orijinli altı bölgenin tanımlanabileceğini göstermiştir. Sınıflandırma başarısı %98.3 olarak tespit edilmiştir [34]. Ceviz yaprak görüntülerini kullanarak sınıflandırma işleminin yapıldığı çalışmalar da literatürde oldukça fazladır. Yapılan çalışma da üç çeşit ceviz görüntüsünü sınıflandırmak için Gri Seviye Birlikte Oluşum Matrisi (GLCM) ile görüntülere ait özellikler elde edildikten sonra geri yayılım sinir ağı ve DVM kullanılmaktadır. Yapılan testler sonucunda %93.3 doğruluk oranı ile sınıflandırıldığı belirlenmiştir [35].

BÖLÜM 3

MATERYAL METOT

Bu bölümde, tez çalışması için oluşturulan özgün veri seti ve kullanılan tüm teknolojiler detaylı olarak açıklanmaktadır.

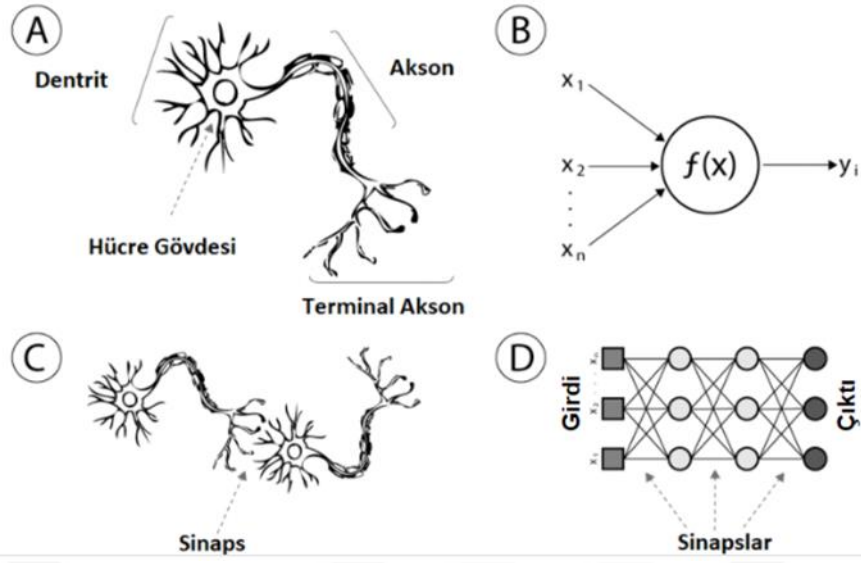
3.1. MAKİNA ÖĞRENMESİ

Makine öğrenmesi, probleme ait ortamdan elde edilen bilgiye dayanarak problemi çözmeyi amaçlayan bilgisayar tabanlı algoritmaların tümüdür [36]. Elle işlenmesi ve analiz edilmesi çok zor olan veriyi kullanarak, en uygun modeli üretmeyi amaçlamaktadır [37]. Literatürde birçok alanda başarılı bir şekilde kullanılır ancak özellikle belirsizliğin çok olduğu problemler için seçilmektedir [38]. Bu algoritmaların bir kısmı sınıflandırma, bir kısmı tahmin, bir kısmı ise kestirim yapmaktadır [36]. Farklı alanda yer alan problemlerin çözülebilmesi için öğrenme tabanlı geliştirilen, denetimli, denetimsiz, yarı denetimli, pekiştirmeli ve aktarım öğrenme gibi birçok farklı makine öğrenmesi yöntemi bulunmaktadır [39]. Bunun yanı sıra makine öğrenmesinin alt kümesi olarak günümüzde sıklıkla kullanılan ve geliştirilen popüler yöntemlerde bulunmaktadır.

3.2. YAPAY SİNİR AĞI

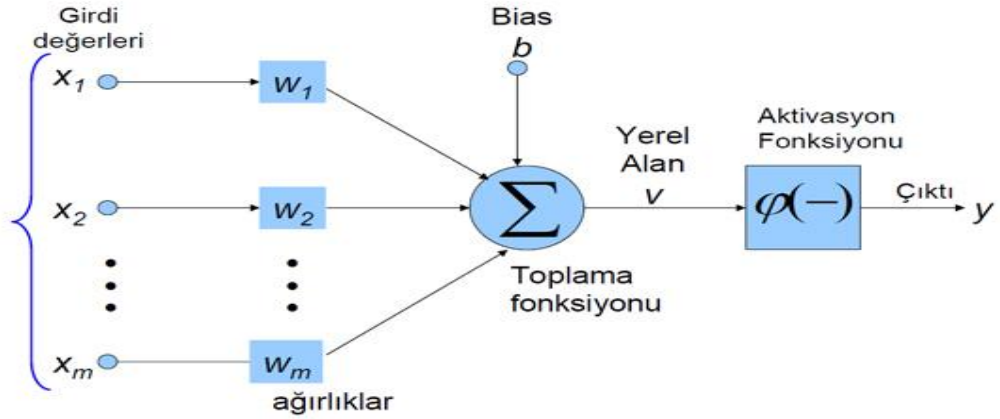
YSA, insan beynini taklit ederek, öğrenme yoluyla geliştirdiği yeni veri üretebilme işlevinin gerçekleştirildiği bilgisayar yazılımıdır. İnsan beyinde ki biyolojik sinir ağlarının yapısı matematiksel olarak modellenmiştir [44].

1943 yılında McCulloch ve Pitts tarafından yazılan bir makale ile ilk kez YSA'dan bahsedilmiştir [45]. YSA'nın biyolojik sinir ağı benzerliği Şekil 3.1' de verilmiştir.



Şekil 3.1. YSA ve biyolojik sinir hücresi yapısı [36].

Bir sinir ağında hücreye n adet veri girişi yapılır. Bu girişler belirlenen ağırlıklar ile çarpılarak toplanır ve toplam değerine bias eklenir ve sonuç bulunur. Sonuç değeri ilgili aktivasyon fonksiyonundan geçirilerek çıktı verisi elde edilir. Basit bir YSA'nın çalışma biçimi Şekil 3.2' de gösterilmiştir.



Şekil 3.2. YSA çalışma şekli.

3.3. DERİN ÖĞRENME

Derin öğrenme nesne tanıma, doğal dil işleme gibi birçok alanda kullanılan, çok katmanlı yapay sinir ağını temel alarak geliştirilmiş, makine öğrenmesi çeşitlerinden

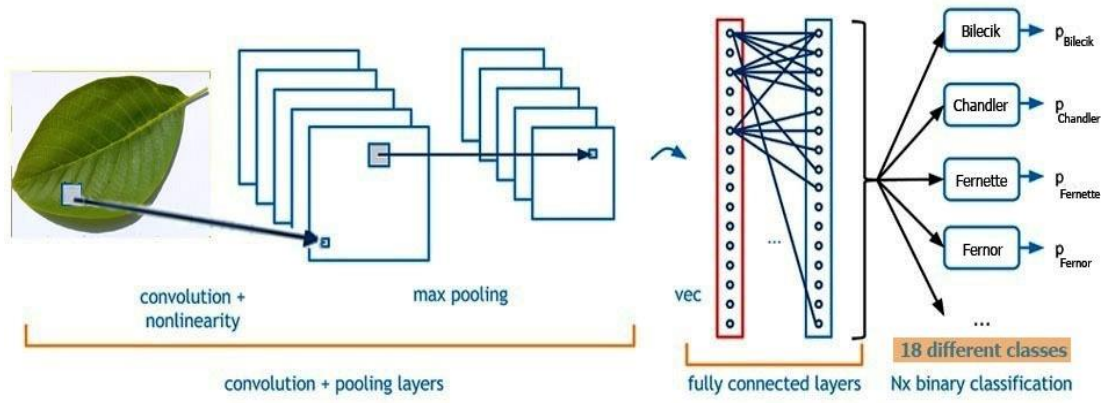
birisidir [40]. Derin öğrenme özellik çıkarma ve hesaplama işlemleri için, doğrusal olmayan birçok işlem birimi katmanını bir arada kullanarak hiyerarşik bir yapı oluşturmaktadır. Her bir katmanda ki çıktı bir sonra ki katmana girdi olarak verilmektedir [41]. MİB ve GİB gelişmeler ile birlikte hesaplama hızlarında ki artış ile birlikte, tek katmanlı YSA'lar çok katmanlı YSA'lara ve sonrasında derin öğrenme algoritmalarına teknolojik ilerleme göstermiştir [42]. Derin öğrenme ifadesi 2000 yıllarında tanıtılmış ve literatürde yer almaya başlamıştır [43].

3.4. EVRİŞİMSEL SİNİR AĞLARI

ESA temel olarak YSA'lara benzemektedir [46]. ESA mimarisi, ImageNet yarışmasında Alex Krizhevsky tarafından önerilen AlexNet ile birlikte literatürde popüler hale gelmiştir [42]. ESA'lar ard arda yerleştirilmiş eğitilebilir katmanlardan oluşmaktadır. ESA'ya giriş verileri gönderildikten sonra tüm katmanlar kendi görevlerini sırasıyla yaparlar ve bu şekilde eğitim süreci gerçekleştirilir. Son aşamada ise olması gereken sonuç ile üretilen sonucu karşılaştırabilmek için bir sonuç çıktısı üretilir. Üretilen sonuç ile olması gereken sonuç arasında ki fark kadar hata değeri oluşmaktadır. Bu hata geri yayılım algoritması ile tüm ağırlıklara aktarılmaktadır. Hatanın azaltılabilmesi için her bir iterasyonla ağırlıkların güncellenmesi yapılmaktadır [6].

ESA mimarisin de katmanlar üç boyut üzerinde çalışmaktadır ve her bir katmanın birbirinden farklı görevleri bulunmaktadır. Günümüzde ESA mimarileri, sınıflandırma [47], nesne tanıma [48], ses tanıma [49], görüntü [50], metin ve video işleme [51, 52], hastalık teşhisleri [53], yüz tanıma [54] gibi birçok alanda kullanılmaktadır. Temel olarak yapay zeka sistemlerinin başarısı, problemde yer alan örneklemin özelliklerinin tespitine bağlıdır. Bu özelliklere öznitelik denilmektedir. ESA mimarisinin en büyük avantajı bu öznitelikleri ham veriden kendisinin çıkarabilme yeteneğidir [55].

ESA mimarisi temel olarak 5 farklı katmandan oluşmaktadır. Bu katmanlar, giriş katmanı, evrişim katmanı (convolution), havuzlama katmanı (pooling), tam bağlı katman ve çıkış katmanıdır [56]. Şekil 3.3' de genel ESA mimarisi gösterilmiştir.



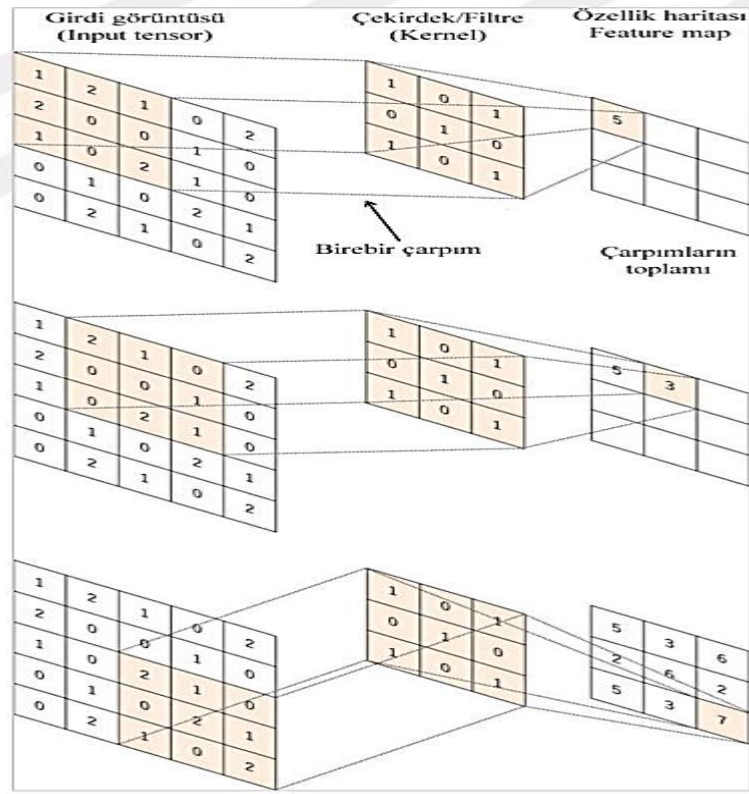
Şekil 3.3. ESA genel mimarisi.

3.4.1. Giriş Katmanı

ESA'ya verilerin ham olarak gönderildiği katmandır. RGB görüntüsü ESA'ya giriş olarak verilirken, Genişlik x Yükseklik x Kanal şeklinde 3 boyutlu bir matris oluşturulmaktadır. Genişlik ve Yükseklik için gereken piksel boyutu kullanılan ESA mimarisine göre değişebilmektedir. Giriş verilerinin boyutu arttıkça daha ayrıntılı bilgiler elde edilirken, hesaplama maliyetleri de artmaktadır.

3.4.2. Evrişim Katmanı (Convolution)

Evrişim katmanı temel olarak özellik çıkarma işleminin yapıldığı katmandır. Çekirdek (kernel) veya filtre olarak adlandırılan bir matris, giriş olarak verilen görüntü matrisi üzerinde soldan sağa ve yukarıdan aşağıya doğru adım adım gezdirilerek, giriş matrisi ile çekirdek matrisinin kesiştiği değerler çarpılır ve başka bir matrise bu çarpımın sonuç değeri yazılır. Bu sayede ESA mimarisinde evrişim katmanı görevini bitirmiştir. ESA mimarisinde ki evrişim işlemi Şekil 3.4' de gösterilmiştir.



Şekil 3.4. Evrişim katmanı yapısı [37].

Evrişim işlemi sonunda ortaya bir özellik haritası çıkmaktadır. Bu harita giriş görüntüsünün boyutu, filtre boyutu, adım büyüklüğü gibi parametrelere göre değişmektedir. Evrişim işleminin formülü denklem 3.1' de verilmiştir [38].

$$(f * g)(t) = \int_{-\infty}^w f(\tau)g(t - \tau)d\tau \quad (3.1)$$

Formülde f giriş görüntüsünü ve g filtreyi temsil etmektedir.

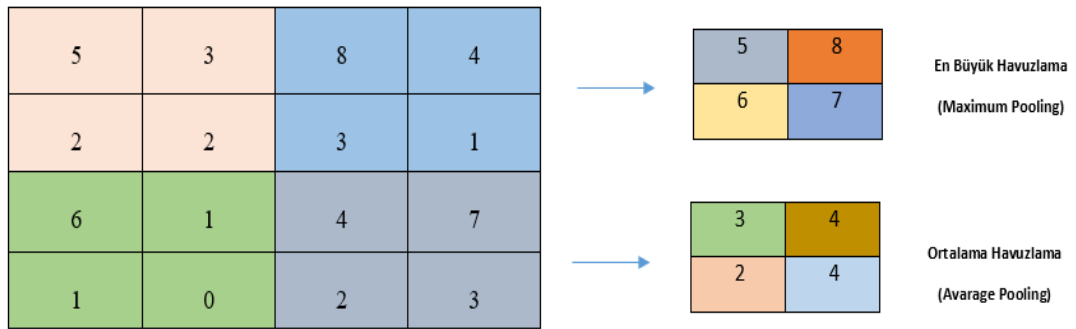
Evrişim işleminde çekirdek boyutu genellikle, 2 x 2, 3 x 3, 5 x 5 veya daha büyük bir kare matris olarak seçilmektedir [39]. Filtre sayısı evrişim işlemi sonucunda elde edilecek olan özellik harita sayısını göstermektedir. Evrişim işleminde giriş görüntü matrisi üzerinde yapılan işlemlerde ki ilerleme boyutuna adım sayısı denilmektedir. Adım sayısı arttıkça çıkış matrisinin boyutu küçülmektedir. Çıkış boyutunun küçülmesinde oluşacak problemlerin çözümü için dolgu (padding) yöntemi

bulunmuştur. Padding işlemi ile giriş matrisinin hem satırına hem de sütununa 0 eklenmektedir.

3.4.3. Havuzlama Katmanı (Pooling)

Evrişim katmanı sonrasında ağda çok fazla parametre ve hesaplama olduğundan dolayı, hesaplama maliyeti artmakta, eğitim yapılacak bilgisayar için yüksek donanım gerekmekte ve aynı zamanda ezberleme ihtimalini arttırmaktadır. Havuzlama katmanı ile giriş görüntüsünün boyutu kademeli olarak düşürülerek bahsedilen sorunlar çözülmeye çalışılmaktadır. Ancak havuzlama katmanı aynı zamanda bilgi kaybına da neden olmaktadır [60].

Havuzlama katmanı literatürde ortaklama katmanı veya alt örnekleme olarak da bilinmektedir. Havuzlama katmanında, en büyük havuzlama (maximum pooling) ve ortalama havuzlama (average pooling) en sık kullanılan yöntemlerdir. En büyük havuzlama yönteminde filtreye karşılık gelen bölümde ki en büyük değer alınmaktadır. Ortalama havuzlama yönteminde ise filtrenin denk geldiği bölümün aritmetik ortalama değeri alınmaktadır. Alınan bu değerler yeni özellik haritasını oluşturmaktadır. Havuzlama katmanının gerçekleşme şeması Şekil 3.5’ de gösterilmiştir.



Şekil 3.5. Havuzlama katmanı gerçekleşme yapısı.

3.4.4. Tam Bağlantılı Katman (Fully Connected Layer)

Tam Bağlantılı Katmanın temel amacı girdi olarak sisteme giren verilerin eğitim verilerine dayanarak çıkarılan özelliklere göre sınıflandırılmasıdır. Tam bağlantılı

katmana önceki evrişim ve havuzlama katmanından veriler matris olarak değil vektör olarak verilmektedir. Önceki katmanda yer alan nöronların her biri Tam bağlantılı katmanda ki nöronlara bağlanmaktadır [61].

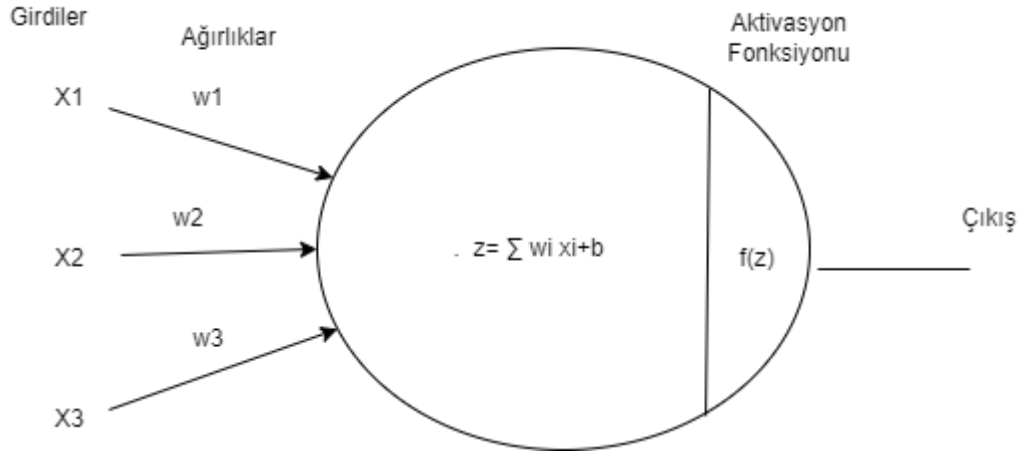
3.4.5. ESA'nın Eğitilmesinde Kullanılan Diğer Yöntemler Ve Parametreler

ESA ile yapılan sınıflandırma işleminde ağın iyileştirilerek başarılı sonuçların alınabilmesi için karar verilmesi gereken birçok özellik bulunmaktadır. Derin öğrenme yöntemlerinde hesaplama maliyetinin düşürülmesi, model başarısının artırılması ve ağın donanım ihtiyaçlarının azaltılabilmesini sağlamak amacıyla hiper-parametreler kullanılmaktadır [40].

ESA'nın eğitilmesinde kullanılan hiper-parametrelere, aktivasyon fonksiyonu, öğrenme katsayısı, filtre boyutu, giriş görüntü boyutu, seyreltme (dropout), katman sayısı, mini-batch size, epoch gibi parametreler örnek olarak verilebilmektedir [41].

3.4.5.1. Aktivasyon Fonksiyonu

Sinir ağlarında görüntü, ses, yazı gibi doğrusal olmayan giriş verileri kullanıldığında ağı eğitilebilmesi için doğrusal verilerin doğrusal olmayan verilere dönüştürülmesi gerekmektedir. Bu işlemin yapılabilmesini aktivasyon fonksiyonları sağlamaktadır. Giriş olarak verilen değer ağırlık ile çarpıldıktan sonra bir eşik değeri (bias) eklenir ve aktivasyon fonksiyonuna verilir. Aktivasyon fonksiyonundan çıkan değer başka bir ağ elemanına verilebilir veya son çıkış değeri olabilir. Aktivasyon fonksiyonu ağı sonu değerini etkilediği için doğru seçilmesi oldukça önemlidir. Problemlerin çözüm şekline göre aktivasyon fonksiyonu seçilmelidir. ESA mimarisinde, Sigmoid, ReLu ve Softmax fonksiyonları sıklıkla kullanılmaktadır [41]. Ayrıca aktivasyon fonksiyonları bir düğümün aktif olup olmama kararını da vermektedirler. Aktivasyon fonksiyonlarının çıktı üretme aşaması Şekil 3.6'da gösterilmiştir. Burada x girdiyi, w ağırlıkları, b eşik değeri ve z çıkış değerini temsil etmektedir.



Şekil 3.6. Aktivasyon fonksiyonu yapısı.

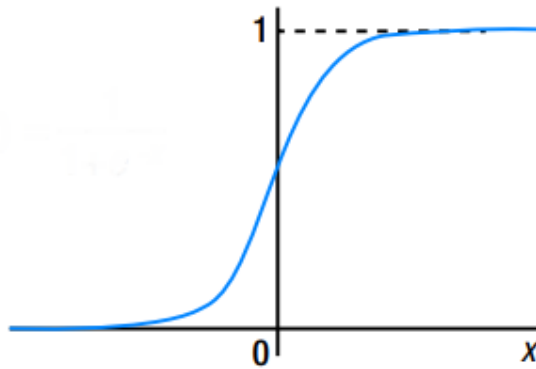
Sigmoid

Sigmoid aktivasyon fonksiyonu literatürde sıklıkla kullanılan ve türevi alınabilen bir fonksiyondur. Giriş olarak $-\infty$ ve $+\infty$ arasında değer alabilirken, çıktı olarak 0 ile 1 arasında bir değer üretmektedir [42].

Sigmoid fonksiyonunun matematiksel ifadesi denklem 3.2’ de gösterilmektedir.

$$f(x) = \frac{1}{1+e^{-x}} \quad (3.2)$$

Matematiksel gösterimden anlaşıldığı üzere x ifadesi çok büyük veya çok küçük bir değer aldığı anda türev 0 değerine yakınsamaktadır. Sigmoid aktivasyon fonksiyonunun grafiksel gösterimi Şekil 3.7’ de gösterilmektedir.



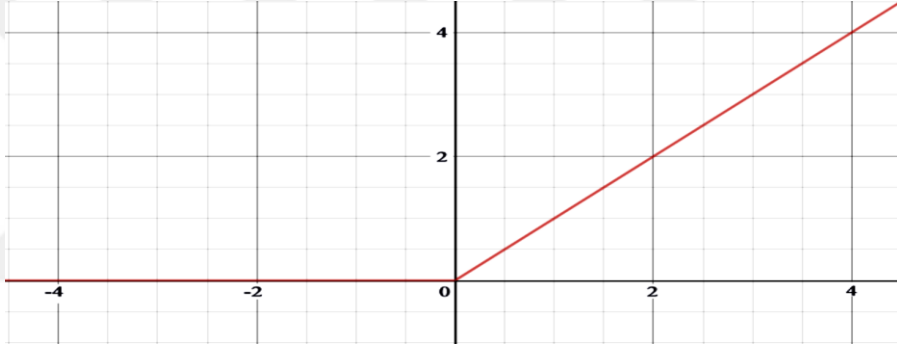
Şekil 3.7. Sigmoid grafiği [61].

ReLU

ReLU aktivasyon fonksiyonu, sıklıkla kullanılmaktadır çünkü diğer fonksiyonlara kıyasla yakınsama hızı oldukça fazladır. Ayrıca tüm düğümleri değil belirli sayıda düğümü etkinleştirdiği için hesaplama yükünü azaltmaktadır. Giriş değeri negatif ise 0, pozitif ise giriş değerini çıkışa vermektedir. ReLu aktivasyon fonksiyonunun matematiksel gösterimleri denklem 3.3’ de gösterilmektedir [65].

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \quad (3.3)$$

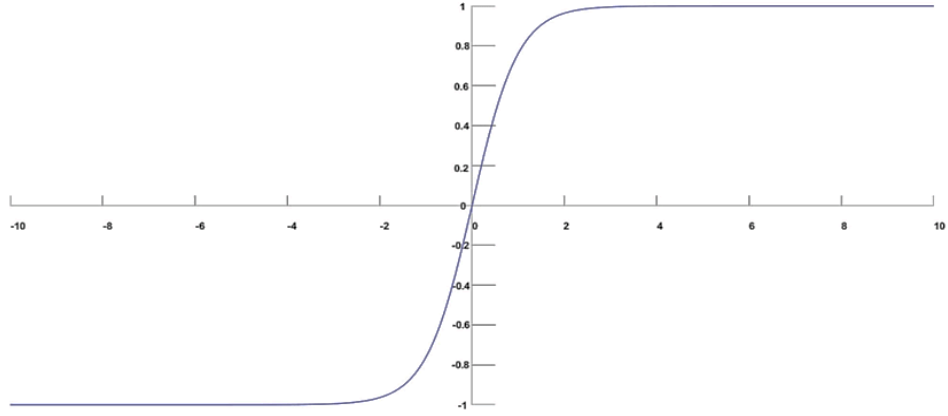
Relu aktivasyon fonksiyonunun grafiği Şekil 3.8 ‘de gösterilmiştir.



Şekil 3.8. ReLu grafiği.

Softmax

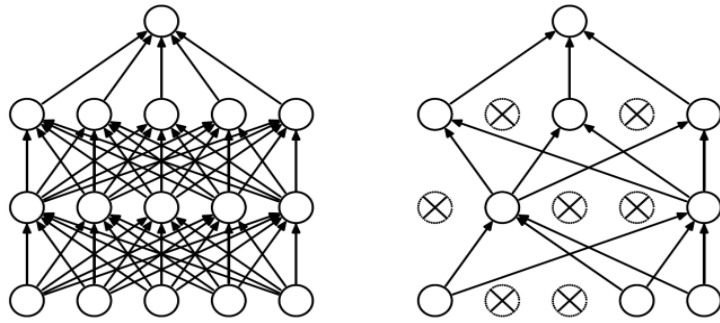
Çoklu sınıflandırma problemlerinde sıklıkla kullanılan bir aktivasyon fonksiyonudur. 0 ile 1 arasında değerlerden oluşan kategoriksel bir çıktı üretir. Tüm olasılık değerlerinin toplamı 1’dir [43]. Softmax aktivasyon fonksiyonu ve olasılık vektörü Şekilde 3.9’ da gösterilmiştir.



Şekil 3.9. Softmax fonksiyonu grafiği [44].

3.4.5.2. Seyreltme Katmanı (Dropout)

ESA’ da aşırı öğrenme yani ezberlemeye karşı alınan önlemlerden birisi dropout işlemidir. Bu işlemde sinir ağında ki bazı düğümler rastgele olarak devre dışı bırakılarak seyreltilme yapılmaktadır. Devre dışı bırakılan düğümlerin geçici olarak gelen giden bağlantıları devre dışı bırakılmalıdır. Seyreltme oranı bazı çalışmalarda 0.1 ile 0.3 arasında bazı çalışmalarda ise 0.5 olarak seçilmektedir [45,46]. Dropout işleminin genel yapısı Şekil 3.10’ da gösterilmiştir.



Şekil 3.10. Dropout işlemi [47].

3.4.5.3. Mini-Batch Değeri

ESA ile yapılan eğitim aşamasında ağda ki hata oranının bağlantılara dağıtılması için geri yayılım algoritması kullanılmaktadır. Bu aşamada tüm verilerin aynı anda işleme alınması zaman, hesaplama, donanım olarak oldukça maliyetli bir işlemdir. Bu

durumun çözülebilmesi için veri setinin aynı anda değil de parça parça eğitim aşaması uygulanmaktadır. Mini-batch değeri 2^n şeklinde belirlenerek aynı anda kaç adet verinin eğitime alınacağını ayarlanmaktadır [48]. Yani paket boyutunun yükselmesi eğitime alınacak veri boyutunun da yükselmesi anlamına gelmektedir. Bu durumda daha fazla hafıza ve donanım maliyeti gerektirmektedir [49].

3.4.5.4. Epoch ve Döngü Sayısı

Belirlenen mini-batch yani paket boyutu kadar verinin eğitilmesine döngü sayısı (iterasyon), tüm veri setinin eğitilmesine yani ileri ve geri yönde tüm ağ üzerinde işlem yapılmasına ise epoch denilmektedir [50]. Örneğin 2000 veriden oluşan bir veri setinde mini-batch boyutu 4 olarak ayarlanırsa, 1 epoch için 500 iterasyon gerekmektedir. Epoch sayısının artması ilk etapta ESA'nın başarısını arttırsa bile belli bir süre sonra bu başarının artmadığı görülmektedir. Bu durumda ağdaki eğitimin sonlandırılması gerekmektedir [51].

3.4.5.5. Öğrenme Katsayısı

Öğrenme katsayısı hata düzeltme katsayısı olarak da bilinmektedir. Sinir ağlarında ağ güncellemesi yapılma aşamasında, geriye doğru türev alınarak fark bulunur ve bu fark öğrenme katsayısı ile çarpılmaktadır. Öğrenme katsayısı çok büyük seçildiğinde hedefe ulaşmada sıkıntı yaşanırken, çok küçük seçildiğinde yakınsama çok uzun sürebilmektedir. Bu yüzden öğrenme katsayısının doğru seçilmesi ağ başarısı için oldukça önemlidir [52].

3.4.5.6. Optimizasyon

Optimizasyon, bir problemde mevcut çözümler arasında ki en iyi çözüm demektir [53]. ESA temel olarak en iyi çözüme ulaşmayı hedefleyen bir mimariye sahiptir. Dolayısıyla bir optimizasyon problemi olarak algılanabilmektedir. Ayrıca ESA'da kullanılan veri sayısı fazla olduğu için hesaplama maliyetleri yüksektir. ESA'da eğitim aşamasında hata oranını en aza indirmek ve maliyetleri azaltabilmek için eğitim aşamasına başlamadan önce mutlaka optimizasyon işlemi yapılmalıdır [54]. ESA'da

sıklıkla kullanılan optimizasyon algoritmaları, Gradyan azalma (Gradient Descent, GD), Adaptive Moment (Uyarlanabilir Moment Tahmini, ADAM), Adagrad, Adadelta ve RMSprop'dır [55].

Gradyan azalma (Gradient Descent)

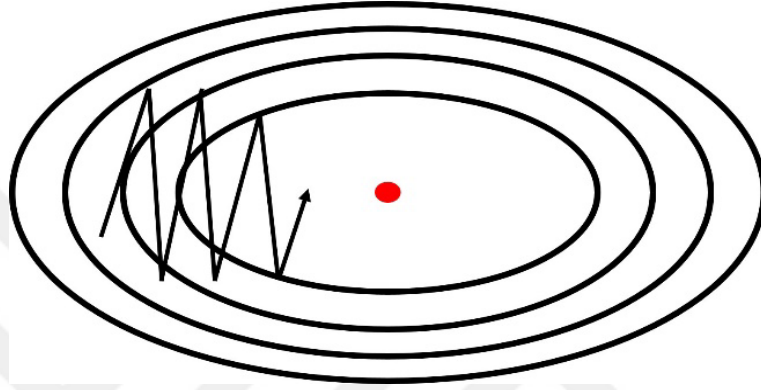
Derin öğrenme algoritmalarında en sık kullanılan optimizasyon algoritmalarından birisi Gradyan azalma yöntemidir. ESA'da ağ güncellemesi yapılırken geri yayılım işlemi yapılmaktadır. Bu işlemde geriye doğru türev alınmaktadır ve aradaki fark bulunarak öğrenme oranı ile çarpıldıktan sonra elde edilen sonuç değeri ağırlık değerinden çıkarılarak ağıdaki güncellenmiş ağırlık değeri bulunmuş olur. Amaç ağırlık değeri ile yeni bulunan ağırlık değerinin en az olmasıdır. Verinin boyutuna göre kullanılabilecek 3 farklı çeşit Gradyan azalma yöntemi bulunmaktadır [78].

Batch Gradient Descent yöntemi tüm veri seti üzerinde işlem yapmaktadır. Tüm veri seti üzerinde işlem yapıldığından dolayı hem yavaştır hem de hesaplama maliyetleri yüksektir. Olasılıksal eğim azalma (Stochastic Gradient Descent) yönteminde her eğitim verisi rastgele olarak seçilmektedir. Sadece seçilen eğitim örneği üzerinde ağırlık güncellemesi yapıldığı için çok daha hızlı ve hesaplama maliyetleri düşüktür. Ancak yakınsama hızı oldukça yavaştır. Mini Batch Gradient Descent yöntemi bahsedilen iki yöntemin avantajlarını birleştirerek oluşturulan yeni bir yöntemdir. Bu eğim düşüm yönteminde, eğitim setinde ki her bir paket için güncelleme gerçekleştirilir. Bu yöntemde global minimum noktasına yakınsama hızı oldukça yüksektir. Mini Batch Gradient Descent yönteminin hesaplanmasında kullanılan matematiksel ifadeler denklem 3.4'de verilmiştir [77-79].

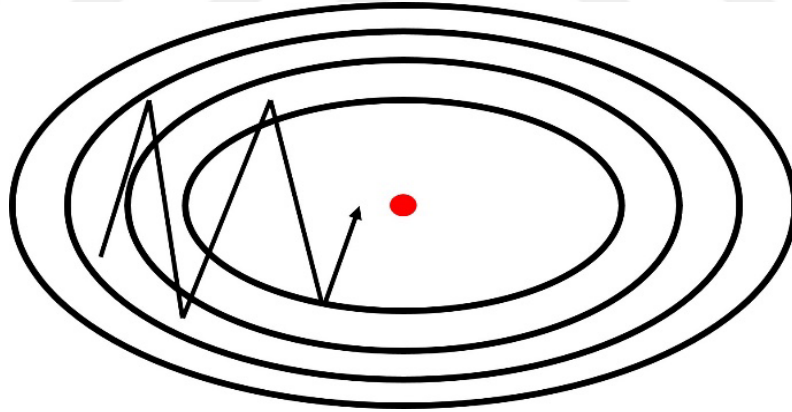
$$W_{i+1} = W_i - \eta * \nabla_{W_i} J(X^{i:i+b}, Y^{i:i+b}; W_i) \quad (3.4)$$

Burada W güncellenmesi gereken ağırlıkları, η öğrenme katsayısını J maliyet fonksiyonunu, i ise yineleme indeksini, y hedefi ve x bu durumda tek bir gözlemi ifade eder.

SGD işleminde en küçük değere ulaşılmasını kolaylaştırmak için kullanılan katsayıya Momentum Katsayı denilmektedir. Momentum değeri, yerel minimumlara takılmayı engellemektedir ancak bu katsayı değerinin seçimi oldukça önemlidir. Momentum katsayısı temel olarak, bir önceki iterasyon değişiminin belirli bir oranda yeni değişim miktarını etkilemesidir [55,56]. Momentum katsayısının kullanıldığı ve kullanılmadığı durumda salınım gerçekleşme gösterimi Şekil 3.11 ve Şekil 3.12’ de gösterilmiştir.



Şekil 3.11. Momentum kullanılmadan gerçekleşen salınım.



Şekil 3.12. Momentum kullanılarak gerçekleşen salınım.

Adagrad (Adaptive Gradient)

Bu yöntemde öğrenme katsayısı elle ayarlanmaz, parametrelere göre kendisi güncellenmektedir [58]. Güncelleme yapılırken öğrenme katsayısı eğitim süresinde büyümektedir ve her adımda daha büyük bir değere bölünmesi öğrenme katsayısının

çok küçük bir aşamaya gelmesine sebep olmalıdır [59]. Parametre güncellemesinin matematiksel formülü denklem 3.5'te gösterilmiştir.

$$\theta_{t+1,i} = \theta_{t,i} - \left(\frac{\mu}{\sqrt{G_{t,ii} + \epsilon}}\right) G_{t+1,i} \quad (3.5)$$

Verilen matematiksel ifadede η öğrenme oranı, $G_{t,i}$ maliyet fonksiyonunun kısmi türevi gösterirken, ϵ ise sıfıra bölünme hatasının önlenmesi için eklenen parametredir.

Adadelta

Adagrad algoritmasında parametre güncellenirken çok fazla küçülerek sıfıra yakınsama probleminin çözümü için ortaya çıkmış bir algoritmadır. Adagrad' da ki geçmiş bütün eğimlerin karesi değerlerinin tamamını kullanmaz. Bunun yerine belli bir kısmını kullanmaktadır. Başlangıç için varsayılan bir öğrenme katsayısı belirlenmesine gerek yoktur. Adadelta optimizasyon algoritmasının matematiksel ifadesi denklem 3.6'da verilmiştir [60],[61].

$$\theta_t = \theta_{t-1} - \frac{RMS[\Delta\theta]}{RMS[g]_{t-1}} g_{t-1} \quad (3.6)$$

Denklemden $RMS[g]_{t-1}$, g_{t-1} 'in karelerinin ortalama karekökünün hatasını göstermektedir.

Karekök Ortalama Yayılımı (Root Mean Square Propagation)

Karekök Ortalama Yayılımı (Root mean square propagation, RMSProp), Adadelta ile benzer zamanlarda geliştirilmiş ve uyarlanabilir öğrenme oranını için sunulmuştur. Adagrad algoritmasında ki öğrenme oranının fazla küçülmesinin önüne geçmeyi amaçlamaktadır. RMSProp algoritmasının matematiksel ifadesi denklem 3.7 ve denklem 3.8'de verilmiştir [62],[63].

$$E[g^2]_t = 0.9E[g^2]_{t-1} + 0.1g_{t-1}^2 \quad (3.7)$$

$$\theta_{t+1} = \theta_t - \frac{\mu}{\sqrt{E[g^2]_{tt} + \epsilon}} g_t \quad (3.8)$$

Verilen denklemlerde γ momentum faktörü, $E[g^2]_{tt}$ t. İterasyonda ki g^2 'nin ortalamasını göstermektedir.

Adam (Adaptive Moment Estimation)

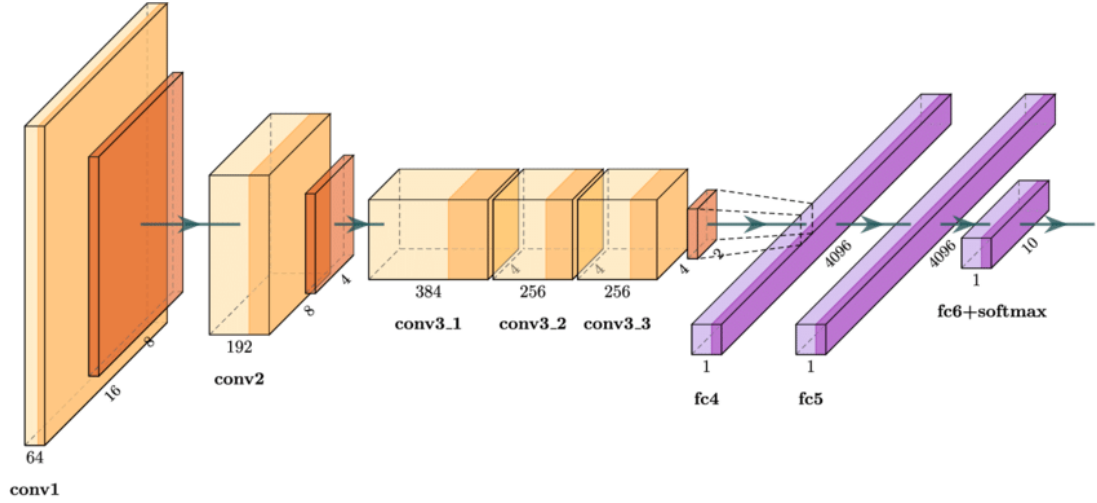
Bu algoritma Adagrad optimizasyon algoritması ile RMSProp optimizasyon algoritmalarının iyi özellikleri birleştirilerek oluşturulmuş bir algoritmadır. Adam optimizasyon algoritmasının matematiksel gösterimi denklem 3.9'da gösterilmektedir. [64].

$$\begin{aligned} V_t &= \beta_1 * V_{t-1} + (1 - \beta_1) * g_t \\ S_t &= \beta_2 * S_{t-1} + (1 - \beta_2) * g_t^2 \\ \Delta w_t &= -\eta \frac{V_t}{\sqrt{S_t + \epsilon}} * g_t \\ W_{t+1} &= W_t + \Delta w_t \end{aligned} \quad (3.9)$$

3.4.6. ESA Algoritmaları

3.4.6.1. AlexNet

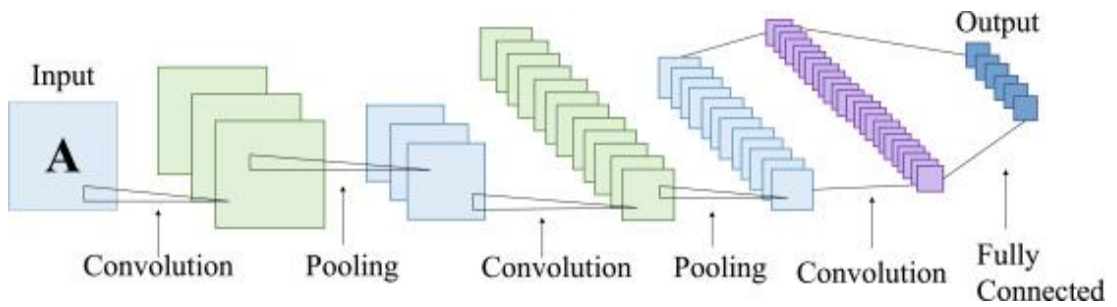
AlexNet modeli, yaklaşık 61 milyon parametrenin tanıtıldığı 8 katmanlı bir ESA mimarisidir. 2012 yılında düzenlenen ImageNet yarışmasında tanıtılmıştır. AlexNet mimarisi, girdi görüntü boyutunu 227×227 olarak alır ve 5 evrimsel katman, ardından 3 tam bağlantılı katman ve son olarak Softmax katmanından oluşur. Bu mimaride, evrimsel ve tam bağlantılı katmanlarda ReLU etkinleştirme işlevini kullanır. Softmax katmanındaki her çıkış değeri, giriş görüntüsüne karşılık gelen çıkış tarafından temsil edilen sınıfa oranıdır [65]. AlexNet mimarisi Şekil 3.13' de gösterilmiştir.



Şekil 3.13. AlexNet genel mimarisi [88].

3.4.6.2. InceptionV3

InceptionV3 Google tarafından geliştirilmiştir. Önceki versiyonlara ek olarak, toplu normalizasyon ve çarpanlara ayırma eklenmiştir. Temel amaç, parametre ve bağlantı sayısını azaltırken ağın verimliliğini düşürmemektir. Son katmanda Softmax kullanılır. InceptionV3 mimarisi toplamda 42 katmandan oluşmakta ve giriş katmanı 299×299 piksel görüntü almaktadır [66]. Şekil 3.14’ de InceptionV3 genel mimarisi gösterilmiştir.

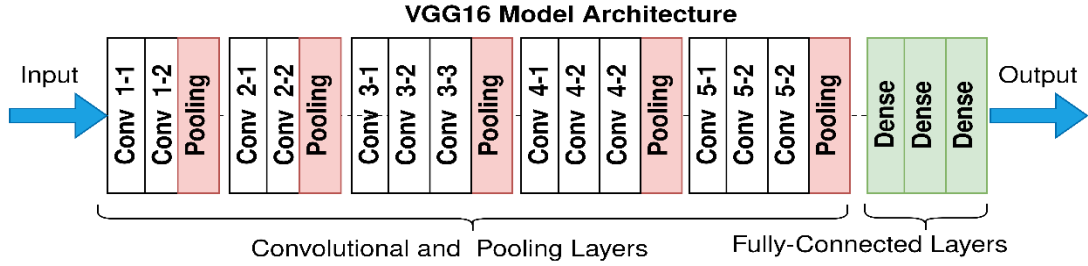


Şekil 3.14. InceptionV3 genel mimarisi [67].

3.4.6.3. Vgg16

Vgg16 2014 yılında önerilmiş yaklaşık 138 milyon parametrelili bir ESA mimarisidir. Bu mimari çok sayıda hiper- parametre kullanmak yerine her adımda 3 x 3 filtre ve 2

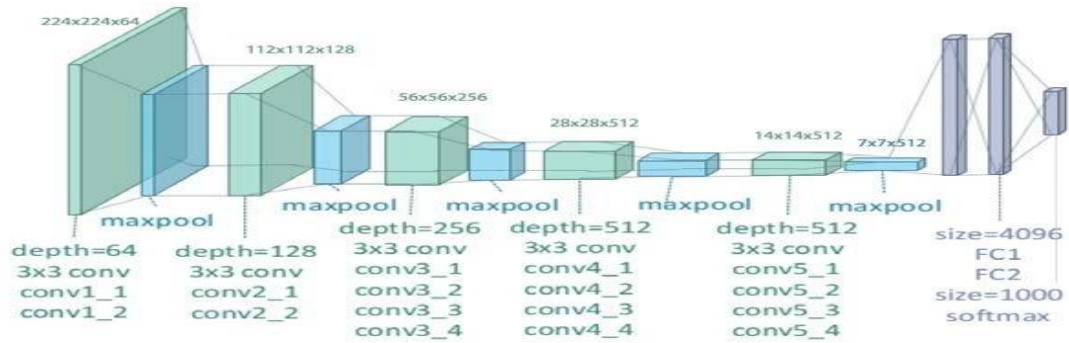
x 2 havuzlama uygulamaktadır. Tam bağlantılı katmanda ise ilk ikisi Relu sonuncusu Softmax olan 3 katman mevcuttur. Vgg16, 16 katman içerir ve giriş katmanı 224×224 piksellik görüntüler olarak çalışmaktadır[68]. Şekil 3.15’de Vgg16 genel mimarisi gösterilmiştir.



Şekil 3.15. VGG16 genel mimarisi [68].

3.4.6.4. Vgg19

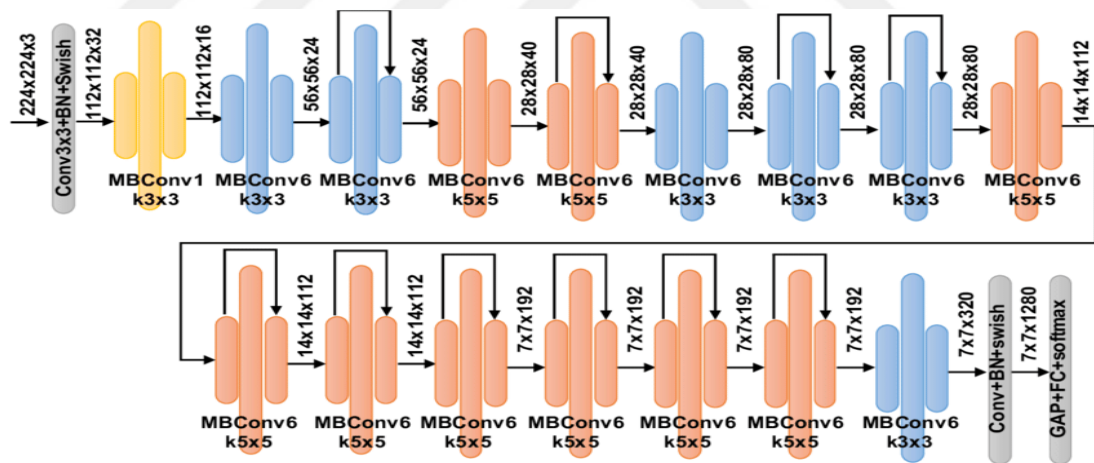
Vgg ağ mimarisi Simonyan ve arkadaşları tarafından önerilmiştir [68]. Vgg19mimarisi, beş blok evrişimli katman ile başlar ve ardından üç tam bağlantılı katmanla yapılandırılır. Evrişimli katmanlar 3×3 , her evrişim katmanının ardından Relu aktivasyonu ve sonrasında 2×2 havuzlama işlemi gerçekleştirilir. 1000 tam bağlantılı katman kullanılmaktadır ve çıkış için softmax aktivasyon fonksiyonu kullanılmaktadır [69]. Vgg19 genel mimarisi Şekil 3.16’da gösterilmiştir.



Şekil 3.16. Vgg19 genel mimarisi [70].

3.4.6.5. EfficientNet

ESA'ları ölçeklendirmek doğruluğun artırılabilmesi için kullanılan en yaygın yollardan birisidir. Derinlik/genişlik/görüntü boyutundan yalnızca bir tanesi veya iki tanesi ölçeklendirilerek daha iyi başarı elde edilebilmesi üzerine yapılan birçok çalışma bulunmaktadır. Bu üç boyut için katman sayısının artırılması ve ölçeklemenin manuel olarak ayarlanması oldukça zordur. Önerilen model, ölçeklendirmeyi sistematik olarak inceler ve ağ derinliğini, genişliğini ve çözünürlüğünü dikkatlice dengeleyen faktörleri tespit ederek daha iyi performans gösterebilmektedir. Bu gözleme dayanarak, basit ama oldukça etkili bir bileşik katsayı kullanarak tüm derinlik/genişlik/çözünürlük boyutlarını eşit olarak ölçekleyen yeni bir ölçekleme yöntemi önerilmiştir. Model en iyi ESA'dan 8,4 kat daha küçük ve çıkarımda 6,1 kat daha hızlı iken, ImageNet'te %97,1 doğruluk bulmuştur. EfficientNet ayrıca CIFAR-100 veri setinde %91,7, Flowers veri setinde %98,8 doğruluğa ulaşmıştır [71]. EfficientNet genel mimarisi Şekil 3.17'de gösterilmiştir.

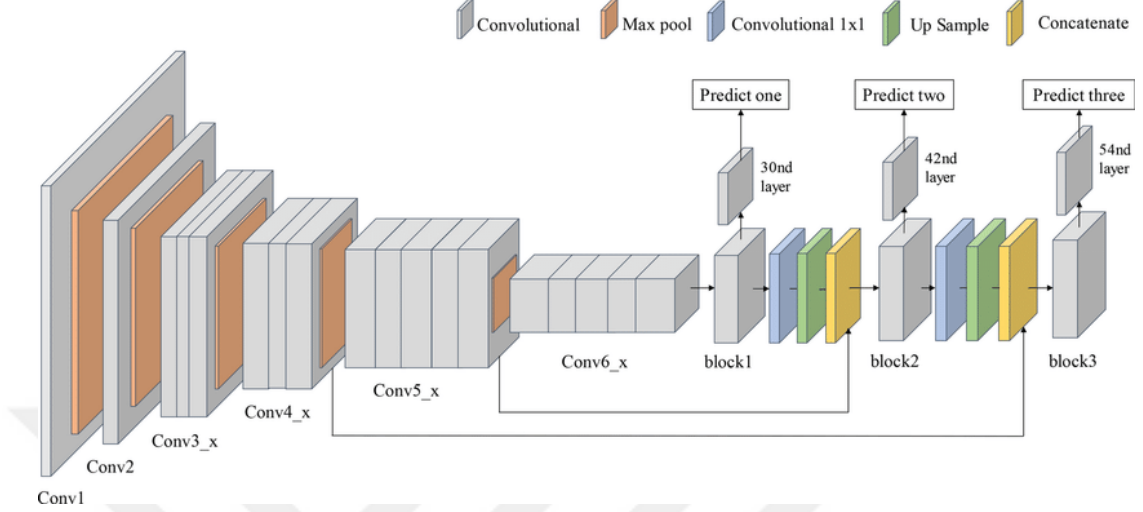


Şekil 3.17. EfficientNet genel mimarisi [72].

3.4.6.6. DarkNet19

DarkNet19 ESA mimarisi üzerinde 64 katman bulunmaktadır. Bu katmanlar, girdi katmanı, Evrişim Katmanı, Toplu Normalleştirme (BN), Leaky ReLU, maksimum havuzlama, genel ortalama havuzlama, softmax ve çıktı katmanlarıdır. Aktivasyon için

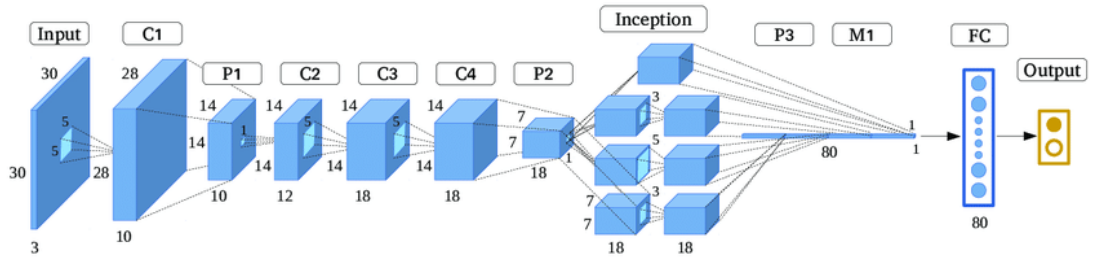
geleneksel ReLU türünün geliştirilmiş hali olan Leaky Relu fonksiyonu kullanılmaktadır [73]. DarkNet19 genel mimarisi Şekil 3.18’de gösterilmiştir.



Şekil 3.18. DarkNet19 genel mimarisi [74].

3.4.6.7. GoogleNet

GoogleNet ağındaki alıcı alanın boyutu, ortalama çıkarma ile RGB renk kanalları olarak 224×224 'tür. Ağıın inşası için kullanılan toplam katman sayısı yaklaşık 100'dür. GoogleNet mimarisinde, 5×5 filtre boyutuna sahip havuzlama katmanı, 128 filtrelili 1×1 filtre boyutuna sahip evrişim katmanı, 1024 birim ve Relu ile tam bağlantılı katman, sınıflandırıcı için ise softmax kullanılmaktadır [75]. GoogleNet mimarisi Şekil 3.19’da gösterilmiştir.



Şekil 3.19. GoogleNet genel mimarisi [76].

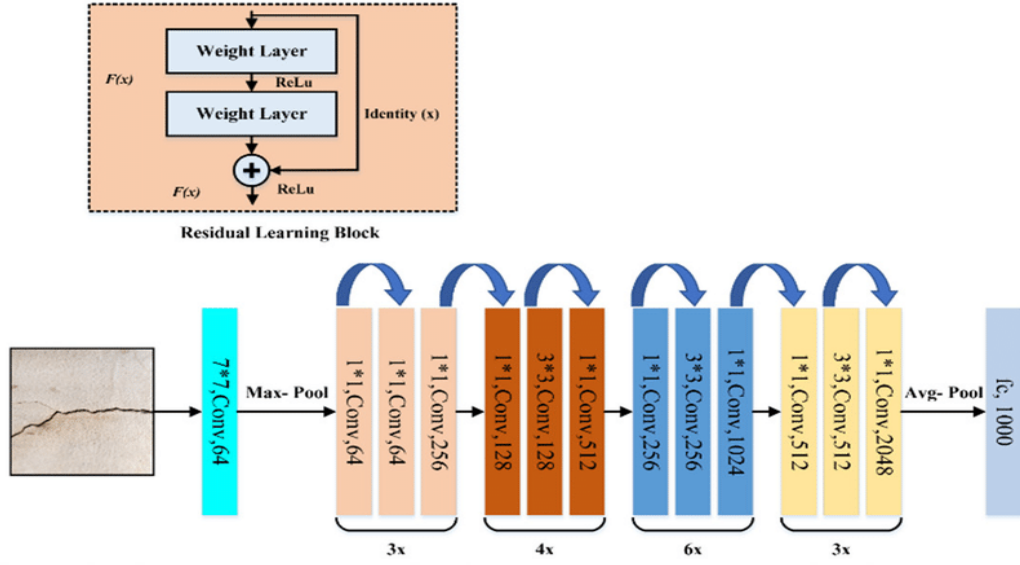
3.4.6.8. ResNet

ResNets mimarisi ILSVRC 2015 yarışmasında son derece başarılı sonuçlar elde etmiş bir ESA modelidir [77]. ResNet etkin bir şekilde 150 katman üstünde bile son derece başarılı eğitimler yapabilmektedir. Resnet18, ResNet50, ResNet101 gibi farklı modelleri mevcuttur [78]. ResNet, çok sayıda artık bloktan oluşmaktadır. Bu artık bloklar, ResNet modellerini oluşturmak için yapı taşları olarak kullanılır. Tüm bloklar evrişim ve havuzlama katmanlarından oluşmaktadır ve 224×224 piksellik giriş görüntülerini alarak çalışmaktadırlar [77]. ResNet18 modelinde 71 katman bulunurken ResNet50 ve ResNet101 ESA modelleri sırasıyla 177 ve 347 katmandan oluşmaktadır [79].

Artık blok, bir katmanın çıktısının alınıp blokta daha derindeki başka bir katmana ekleneceği şekilde ayarlanmış bir katman yığınıdır. Katman girdisini (x) gideceği katman çıkışına taşıyan düz çizgiye artık bağlantı (residual connection) denir. Bu bağlantılar bir veya daha fazla katmanı atlayabilirler. Artık bağlantı sonrası katman çıktısı formülü denklem 3.10'da gösterilmektedir [77].

$$H(x) = f(wx + b) + x \quad (3.10)$$

Artık bloklar da girdiler, katmanlar arasında kalan bağlantılar üzerinden daha hızlı yayılabilir. Ayrıca katmanları atlamak, ağdaki benzer eşlemelerin daha kolay öğrenilmesini sağlamaktadır. Katman sayısı artıp, ağ derinleşmeye başladığında, ESA modellerinin performansları düşmeye başlamaktadır. Bu durumun çözümü için artık blok tabanlı ESA modelleri geliştirilmiştir. ResNet genel mimarisi Şekil 3.20'de gösterilmiştir.

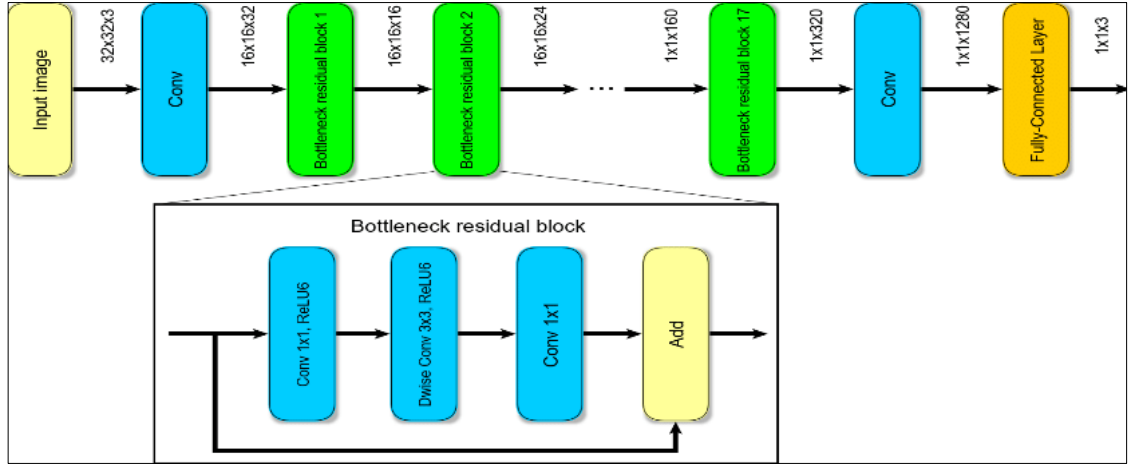


Şekil 3.20. ResNet genel mimarisi [80].

3.4.6.9. MobileNetV2

Derin öğrenme algoritmaları ile görüntü tanıma problemleri genel olarak yüksek bilgisayar kaynağına ihtiyaç duymaktadır. Bu sorunun çözümü için hafif derin öğrenme modelleri üretilerek, model parametrelerinin azaltılması ve hesaplamalarının hızlandırılması sağlanmıştır [81].

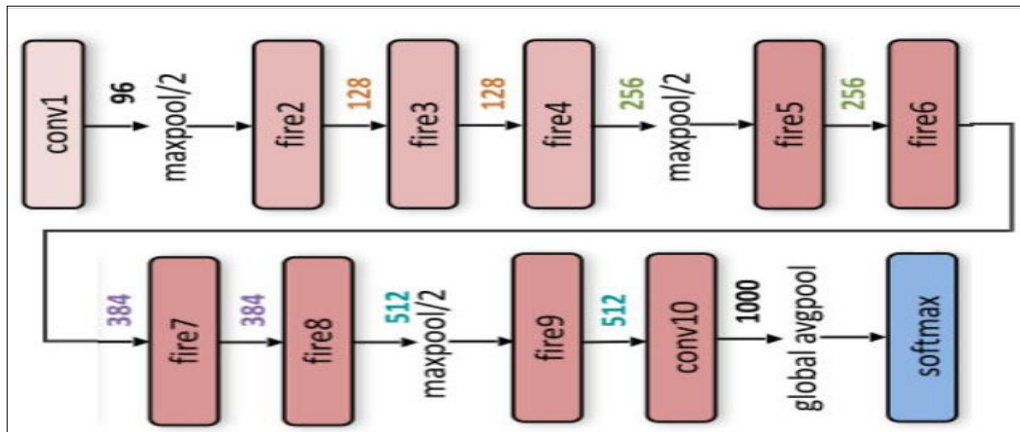
Hafif derin öğrenme algoritmalarından birisi MobilnetV2 ESA algoritmasıdır. MobilenetV2 daha düşük kaynaklarla, daha hızlı eğitim yapmayı amaçlamaktadır. Ters çevrilmiş artık bloklar ve darboğazlar V1'e eklenmiştir. Ters yapıdaki artık bloklar ve aktivasyon fonksiyonları model kararlılığını artırarak eğitimi hızlandırmaktadır [82]. MobilnetV2 ESA algoritmasının mimarisi Şekil 3.21'de verilmiştir



Şekil 3.21. MobilnetV2 mimarisi [83].

3.4.6.10. SqueezeNet

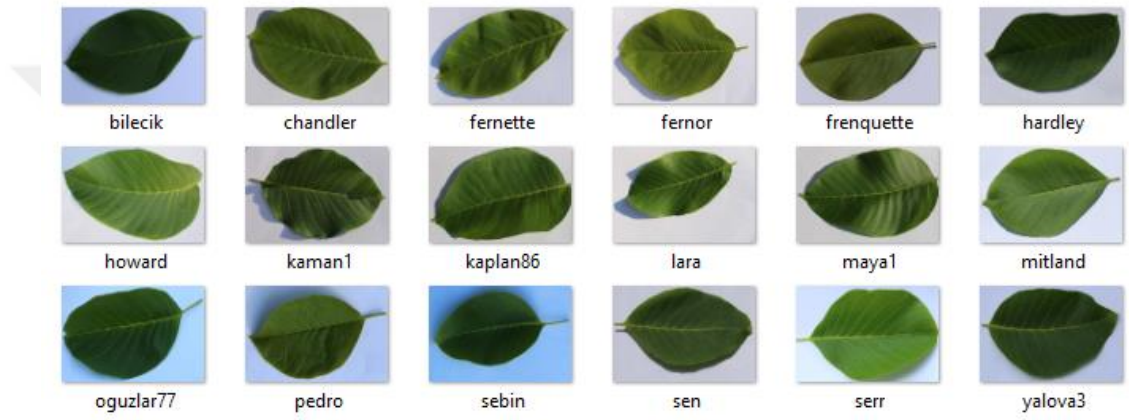
SqueezeNet ilk kez 2016 yılında duyurulmuş, AlexNet'ten 50 kat daha az parametreye sahip ancak 3 kat daha hızlı bir ESA modelidir. MobilnetV2 gibi hafif derin öğrenme modellerinden birisidir. Daha küçük boyutlu olması sayesinde iletişim yükü azdır, kolay modelleme ve kolay taşınabilme özelliğine sahiptir [84]. SqueezeNet mimarisinde 5 farklı katman olmak üzere toplam 15 adet katman bulunmaktadır. Bunlar, iki evrişim, üç max havuzlama, sekiz yangın katmanı, bir global ortalama havuzlama ve bir çıkış katmanıdır [85]. SqueezeNet'in bu avantajlarından dolayı, önerilen model için kullanılacak ESA modeli olarak seçilmiştir. SqueezeNet mimarisinin genel yapısı Şekil 3.22'de gösterilmiştir.



Şekil 3.22. Squeezenet mimarisi [86].

3.5. VERİ SETİ

Veri seti 18 farklı ceviz çeşidinden alınan 1751 yaprak görüntüsü ile oluşturulmuştur. Görüntüler Yalova Atatürk Bahçe Kùltürleri Merkez Araştırma Enstitüsü Uygulama Bahçesinde yer alan ceviz ağaçlarından Canon Eos D2000 fotoğraf makinesi ile çekilmiştir. Veri setinde yer alan ceviz çeşitlerinin sınıflandırılması alanında uzman kişilerden yardım sağlanmıştır. Veri setinde bulunan 18 çeşit ceviz türü örnekleri Şekil 3.23’de gösterilmiştir.



Şekil 3.23. Toplanan veri setinde ki 18 çeşit ceviz yaprağı görüntü örnekleri.

Veri setinde yer alan ceviz çeşitleri ve her ceviz çeşidine ait veri sayıları Çizelge 3.1’de verilmiştir.

Çizelge 3.1. Toplanan veri setindeki 18 Ceviz çeşidine karşılık gelen sayılar.

Çeşit İsmi		Görüntü Sayısı	Çeşit İsmi		Görüntü Sayısı
1	Bilecik	96	10	Lara	63
2	Chandler	82	11	Maya1	74
3	Fernette	89	12	Mitland	147
4	Fernor	104	13	Oguzlar77	59
5	Frenquette	126	14	Pedro	77
6	Hardley	95	15	Sebin	88
7	Howard	85	16	Sen	157
8	Kaman1	98	17	Ser	119
9	Kaplan86	84	18	Yalova3	108
				Toplam	1751

3.6. ÖN İŞLEME

Deneysel testler yapılmadan önce görüntüleri iyileştirmek için veri setinde yer alan görüntülere ön işlemler uygulanmıştır. Görüntülerde çekimlerden kaynaklı hatalar ve gürültüler mevcuttur. Veri seti için çekilen ham fotoğraflardan daha iyi sınıflandırma yapabilmek için sadece yaprak görüntüsünün olduğu alanı tespit edip kesmek için görüntü işleme metotları kullanılarak ön işlemler yapılmıştır.

İlk aşamada görüntünün siyah beyaza çevrilmesi işlemi gerçekleştirilmiştir. Önce görüntülere yerel görüntü aralığı (Local range of image) filtresi 7,7 komşuluk ile uygulanarak, görüntülere ait yerel aralıklar elde edilmiştir. Elde edilen bu yerel aralık değerleri 20 eşik değeri baz alınarak en uygun siyah beyaz görüntüsü elde edilmiştir. İkinci aşama da, görüntülerde ki gürültüleri yok etmek için morfolojik metotlar kullanılmıştır. Kullanılan morfolojik işlemler sırasıyla aşınma (erosion), genişleme (dilation) ve kapama (closing) işlemleridir.

Aşınma ve genişleme işlemlerinde parametre olarak morfolojik yapı elemanı oluşturulmuştur. Bu eleman, gerçek piksellerin morfolojik hesaplamaya dahil edildiği ve yanlış piksellerin dahil edilmediği, iki boyutlu veya çok boyutlu, ikilik değerlere sahip bir komşuluk matrisidir. Aşınma işleminde öncelikle genişliği 10 piksel olan bir kare yapılandırma elementi oluşturulmaktadır. Bu işlemten sonra görüntünün karanlık alanlarının çevrelediği aydınlık alanlar daraltılır ve parlak alanların çevrelediği karanlık alanlar büyütülür. Bu işlemten sonra genişleme işlemi, genişliği 75 piksel olan bir kare elementi ile uygulanmıştır. Genişleme işleminde görüntüdeki karanlık alanların çevrelediği parlak alanlar genişletilirken, özelliğin boyutuna ve karanlık alana bağlı olarak parlak alanların çevrelediği karanlık alanlar zayıflar ve hatta kaybolur. Genişleme işlem formülü denklem 3.11'de verilmiştir. Burada A'nın B tarafından genişlemesi Bx'in A ile kesişimi boş olmayacak şekilde x' in tüm noktaları için ayarlanır [87].

$$A \oplus B = \{ x \mid (Bx) \cap A \neq \emptyset \} \quad (3.11)$$

Bx , A 'nın bir alt kümesidir. Küme işlemleri açısından A 'nın B tarafından aşınma formülü denklem 3.12'de verilmiştir [87].

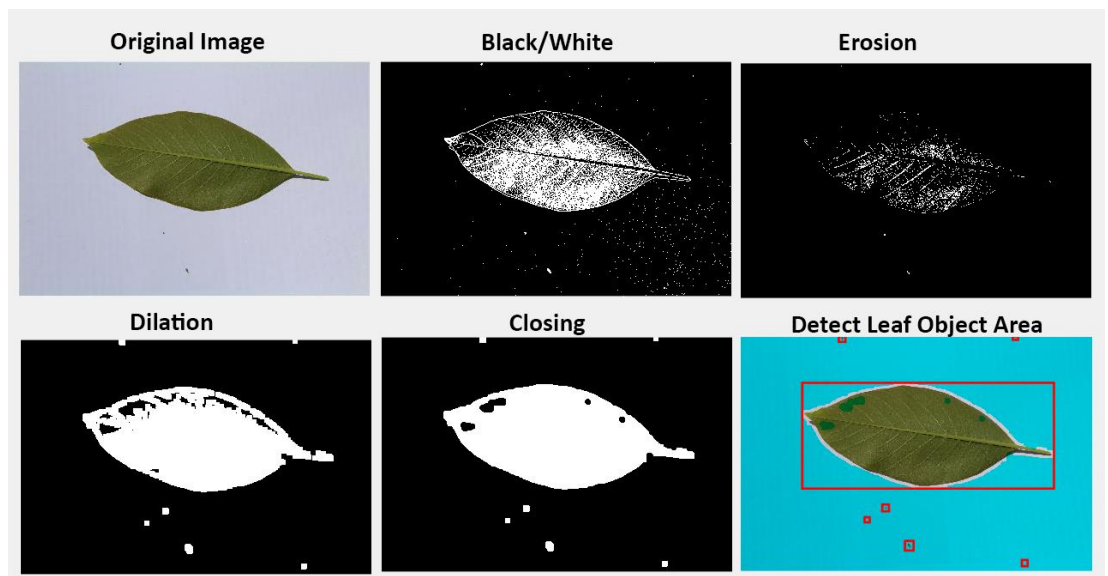
$$A \ominus B = \{x \mid (Bx) \cap A \neq \emptyset\} \quad (3.12)$$

Kapama işlemi için yapılandırma elementi yerine disk yöntemi uygulanmıştır. Bu yöntemi kullanarak morfolojik işlemlerin daha hızlı çalışması sağlanmaktadır. Kapama işlemi, genişleme ve aşınma işlemlerinin ikilik görüntüye ardışık olarak uygulanmasıyla görüntü üzerindeki gürültüler en aza indirgenmiştir. Kapatma işleminde konturdaki boşluklar doldurulmaktadır. Kapatma işleminin formülü denklem 3.13'de gösterilmiştir [88].

$$A \circledast B = \{(A \oplus B) \ominus B\} \quad (3.13)$$

Görüntü üzerinde kalan nesneler 2 boyutlu ikilik görüntü üzerinde bağlı bileşenleri etiketlenmesi (Label connected components in 2-D binary image) metodu ile nesneler tespit edilmiştir. Son olarak nesnelerden en büyük olan yaprağın dış çerçevesi alınarak sadece yaprak bölgesi alınarak veri seti görüntüleri elde edilmiştir.

Veri setine uygulanan ön işlem aşamaları Şekil 3.24'de gösterilmiştir.



Şekil 3.24. Verinin ön işleme aşamaları.

3.7. VERİ ARTIRMA

Veri arttırma (data augmentation), derin öğrenmede yaygın olarak kullanılan bir yöntemdir. Veri arttırma teknikleri, modelin genelleme yeteneğini arttırmak için kullanılmaktadır [89].

Ceviz yaprak görüntülerinden oluşturulan ön işleme uygulanmış veri setimize veri arttırma teknikleri uygulanmıştır. Veri setine 0-30 derece arasında dönüş (rotation), 1-1.5 arasında parlaklık (brightness), kırpma (shear), yakınlaştırma (zoom) ve çevirme (flip) işlemleri uygulanmıştır. Yapılan işlemler sonucunda veri setinde yer alan görüntü sayısı 6606'ya çıkartılarak veri setimiz yaklaşık 4 kat arttırılmıştır. Deneysel testlerde arttırılmış veri seti olarak kullanılmıştır. Arttırılmış veri setindeki türlerin sayıları Çizelge 3.2'de verilmiştir.

Çizelge 3.2. Arttırılmış veri seti görüntü sayıları.

	Çeşit İsmi	Görüntü Sayısı		Çeşit İsmi	Görüntü Sayısı
1	Bilecik	378	10	Lara	249
2	Chandler	324	11	Maya1	291
3	Fernette	351	12	Mitland	418
4	Fernor	406	13	Oguzlar77	232
5	Frenquette	488	14	Pedro	303
6	Hardley	375	15	Sebin	344
7	Howard	332	16	Sen	528
8	Kaman1	389	17	Ser	441
9	Kaplan86	330	18	Yalova3	427
				Toplam	6606

3.8. GRADYAN AĞIRLIKLIL SINIF AKTIVASYON EŞLEMESİ (GRADIENT-WEIGHTED CLASS ACTIVATION MAPPING)

Grad-Cam, görüntünün hangi bölümlerinin sınıflandırma için en önemli olduğunu anlamak için ağı tarafından belirlenen evrişimsel özelliklere göre sınıflandırma puanının gradyanını kullanır [90]. Grad-Cam, verilen görseli tanımlarken ısı haritaları kullanarak, görselin ayırt edici özelliklerini belirgin hale getiren yöntemdir. Hangi özelliğin ne kadar önemli olduğunu derecelendirebilmek için gradyanın tüm

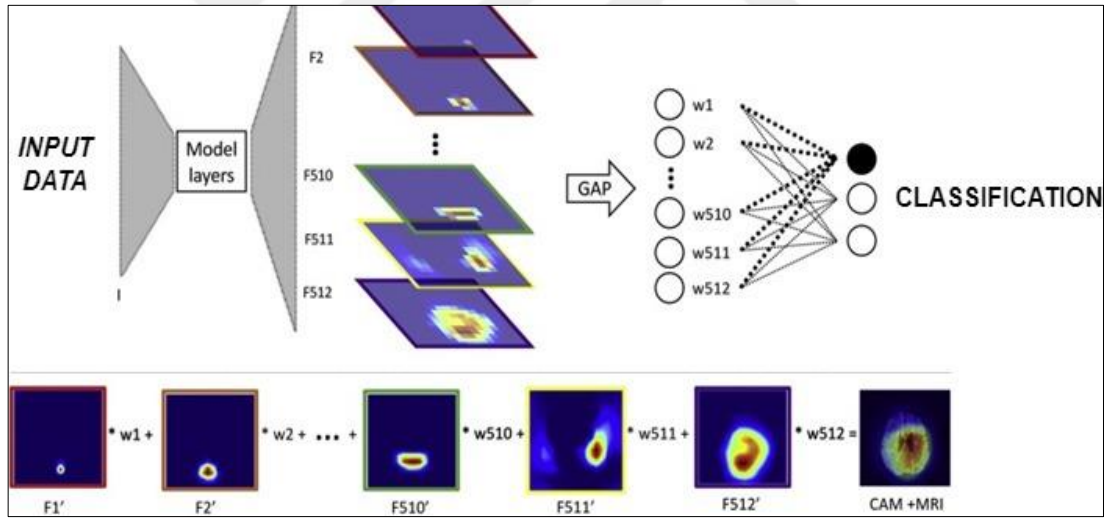
elemanlarının ortalaması alınmalıdır. Grad-Cam iyi bir yöntemdir ancak hassas veriler ve hassas modeller için uygun bir yöntem değildir [90].

Grad-Cam bir derin öğrenme modelinin herhangi bir katmanında kullanılabilmektedir. Matematiksel formülü denklem 3.14' de verilmiştir [91].

$$w_k^c = \frac{1}{Z} \sum_{i=0}^n \sum_{j=1}^h \frac{\partial y^c}{\partial A_{ij}^k} \quad (3.14)$$

Burada; y^c c sınıfının puanını belirtirken ve A_k 'nin boyutu ise $W \times H$ 'dir. y^c 'nin A_k 'ya göre diferansiyel işlemi yoluyla, c sınıfı için A_k haritasının ağırlığı hesaplanır ve w_k^c olarak gösterilmektedir. Z, normleştirme faktörüdür.

Grad-Cam metodunun çalışma mimarisi Şekil 3.25'de gösterilmiştir.



Şekil 3.25. Grad-Cam çalışma mimarisi [92].

3.9. GRİ SEVİYELİ EŞ-OLUŞUM MATRİSİ (GRAY LEVEL CO-OCCURRENCE MATRIX)

Gri seviyeli eş oluşum matrisi (Gray-Level Co-Occurrence Matrix, GLCM), Haralick tarafından önerilmiş doku tabanlı bir öznitelik çıkarma yöntemidir [93]. GLCM, komşu hücre konumlarında belirli görüntü hücresi değerleri çiftlerinin ne sıklıkta meydana geldiğini hesaplayarak çalışmaktadır[94]. Görüntünün değişik

bölgelerindeki birbiri ile ilişki olan yoğunluk değerlerine bakılarak dokusal özellik çıkarma işlemi için yaygın olarak kullanılan bir algoritmadır [95]. GLCM ile toplam 22 farklı dokusal özellik elde edilebilmektedir. Bunların bazıları Entropi (Entropy), Enerji (Energy), Korelasyon (Correlation) ve Kontrast (Contrast)'tır. Matematiksel formülleri sırasıyla denklem 3.15, 3.16, 3.17 ve 3.18'de verilmiştir [96].

$$\text{Entropi} = \sum_{i,j=0}^{N-1} -\ln(p_{ij})p_{ij} \quad (3.15)$$

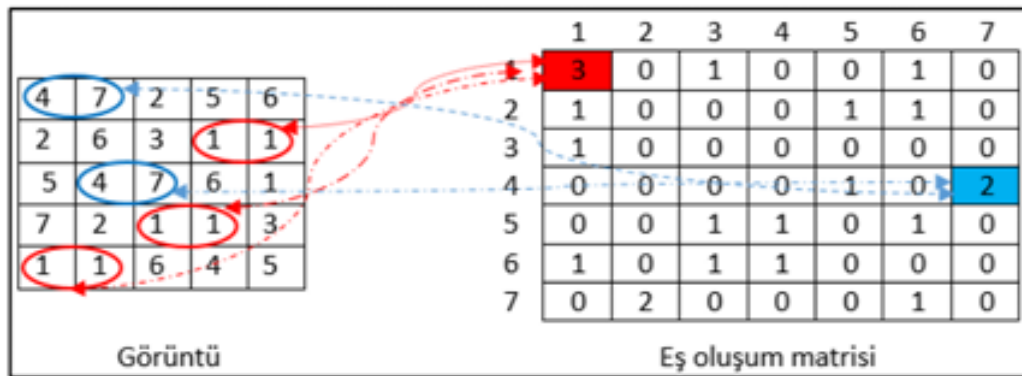
$$\text{Enerji} = \sum_{i,j=0}^{N-1} (p_{ij})^2 \quad (3.16)$$

$$\text{Korelasyon} = \sum_{i,j=0}^{N-1} p_{ij} \frac{(i-\mu)(j-\mu)}{\sigma^2} \quad (3.17)$$

$$\text{Kontrast} = \sum_{i,j=0}^{N-1} p_{ij} (i-j)^2 \quad (3.18)$$

Burada, p_{ij} GLCM matrisinin elemanını, μ GLCM matrisinin ortalamasını, σ tüm piksellerin yoğunluklarının varyansını göstermektedir.

GLCM matrisinin bir görüntü üzerinde nasıl oluşacağı Şekil 3.26'da gösterilmiştir.



Şekil 3.26. GLCM'nin görüntü üzerinde oluşturulma örneği [95].

3.10. ATOM ARAMA OPTİMİZASYONU

Atom Arama Optimizasyon (AAO) algoritması Zhao ve arkadaşları tarafından 2019 yılında tanıtılmış, popülasyon tabanlı sezgisel bir optimizasyon algoritmasıdır. AAO'da arama uzayında yer alan her bir atomun konumu, kendisinden ağır ya da hafif bir atomu işaret eden çözüme sahiptir. Popülasyonda yer alan her atom birbirini iter veya çeker. Hafif atomlar hızlıdır ve arama uzayında iyi çözümleri bulmak için tarama yapar. Ağır atomlar ise yavaştır ve yerel uzayda iyi çözümleri taramaktadır [97].

Atomlar, en iyi atoma doğru hareket etmek için zorlanmaktadırlar ve i . atomun kısıt formülü denklem 3.19'da verilmiştir [98].

$$Q_i(t) = [|x_i(t) - x_{best}(t)|^2 - b_{i,best}^2] \quad (3.19)$$

Verilen formülde $x_{best}(t)$ popülasyonda yer alan en iyi atomun t . iterasyonda ki konumu ve $b_{i,best}^2$ popülasyondaki en iyi atom ile i . atom arasındaki sabit bağ uzunluğunu göstermektedir.

Kısıtlama kuvvetinin formülü denklem 3.20'de verilmiştir [98].

$$G_i^d(t) = \lambda(t)(x_{best}^d(t) - x_i^d(t)) \quad (3.20)$$

Verilen formülde, $\lambda(t)$ T Langragian çarpanı, β ise çarpan ağırlığıdır.

Atomların konum ve hız vektörlerinin formülleri denklem 3.21 ve denklem 3.22'de gösterilmiştir [99].

$$v_i^d(t+1) = rand_i^d v_i^d(t) + a_i^d(t) \quad (3.21)$$

$$x_i^d(t+1) = x_i^d + a v_i^d(t+1) \quad (3.22)$$

AAO algoritmasında, hem keşfi hem de sömürüyü arttırabilmek için atomların K komşuları kadar iyi uygunluk değerine sahip birçok atomla etkileşime girmeleri

gerekmektedir. K değeri her iterasyon sonucunda azalmaktadır. K değeri denklem 3.23'te gösterildiği şekilde hesaplanmaktadır [98].

$$K(t) = N - (N - 2) * \sqrt{\frac{t}{T}} \quad (3.23)$$

Denklem 4.5'te toplam atom sayısı N , iterasyon sayısı t ve maksimum iterasyon sayısı T olarak belirtilmiştir.

AAO algoritması kısıtlı problemlerin çözümünde yüksek başarı gösteren popülasyon tabanlı sezgisel bir optimizasyon algoritmasıdır [97]. Ayrıca AAO algoritmasının yapılan çalışmalar da öznitelik seçim işlemlerin de başarılı sonuçlar verdiği görülmektedir [100].

3.11. BALİNA OPTİMİZASYON ALGORİTMASI

Balina optimizasyon algoritması (BOA), Kambur Balinaların kabarcık avlanma stratejisinden esinlenilmiş, Mirjalili ve Lewia tarafından 2016 yılında önerilmiş meta sezgisel bir optimizasyon yöntemidir [101]. Kambur balinaların avlanma stratejisi Şekil 3.27'de gösterilmiştir.



Şekil 3.27. Kambur balina avlanma şekli [102].

BOA avın etrafını sarma, ava doğru hareket etme ve av arama şeklinde temel olarak üç bölümden oluşmaktadır [101].

3.11.1. Avın Etrafını Sarma

Kambur balinalar öncelikle avın yerini tanır ve etrafını sarar. En iyi çözüm bulunduktan sonra diğer çözümlerin konumları, en iyiye göre güncellenmektedir. Matematiksel ifadeler denklem 3.24 ve 3.25’de gösterilmiştir [101].

$$D^{\rightarrow} = |\vec{C}. X^{\rightarrow} * (t) - X^{\rightarrow}(t)| \quad (3.24)$$

$$\vec{X}(t + 1) = \vec{X} * (t) - \vec{A}.D \quad (3.25)$$

Verilen denklemlerde t mevcut iterasyonu, $A^{\rightarrow}$ ve $C^{\rightarrow}$ katsayı vektörlerini, X^* en iyi çözüm vektörünü göstermektedir.

$A^{\rightarrow}$ ve $C^{\rightarrow}$ vektörlerinin hesaplanması denklem 3.26 ve denklem 3.27’de gösterilmektedir [101].

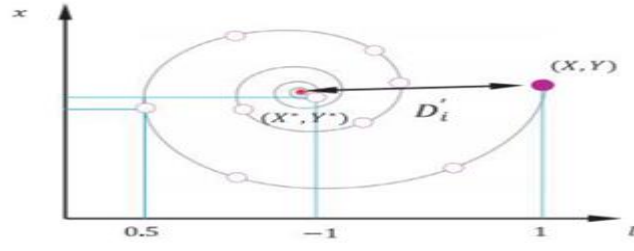
$$\vec{A} = 2\vec{a}.\vec{r} - \vec{a}^{\rightarrow} \quad (3.26)$$

$$\vec{C} = 2.r^{\rightarrow} \quad (3.27)$$

Verilen denklemlerde $a^{\rightarrow}$ iterasyon boyunca 2’den 0’a doğru azalan vektörü, $r^{\rightarrow}$ rastgele vektörünü göstermektedir.

3.11.2. Ava Hareket Edilmesi

Av bulunduktan sonra, balina çemberi daraltma ve spiral hareket etme şeklinde iki farklı hareketle ava yaklaşmaktadır. Spiral hareket etme işlemi Şekil 3.28’de gösterilmiştir [101].



Şekil 3.28. Spiral hareket.

Spiral hareket için arama ajanı ve en iyi arama ajanı arasında ki farkın bulunması gerekmektedir. Spiral hareket formülü denklem 3.28 ve denklem 3.29’da verilmiştir.

$$X(t+1) = \vec{D'} \cdot e(bl) \cdot \cos(2\pi l) + \vec{X}^*(t) \quad (3.28)$$

$$\vec{D'} = \vec{X}^*(t) - \vec{X}(t) \quad (3.29)$$

b logaritmik spiral sabitini, l ise $[-1,1]$ aralığında rastgele bir sayıyı göstermektedir. $\vec{A}$ ve $\vec{C}$ katsayılarıyla bulunan en iyi arama ajanının etrafında ki noktalara yerleşilebilmektedir.

Balinalar ava hareketi sırasında, doğrusal veya spiral hareketlerden birisini seçmelidir. Hangi hareketin seçileceğine denklem 3.30 ve denklem 3.31’deki formüle göre karar verilmektedir.

$$\vec{X}(t+1) = \{ \vec{X}(t) - \vec{A} \cdot \vec{D} \quad p < 0,5 \quad (3.30)$$

$$\vec{D} \cdot ebl \cdot \cos(2\pi l) + \vec{X}^*(t) \quad p \geq 0,5 \quad (3.31)$$

p , $[0,1]$ arasında rastgele üretilmiş bir sayıdır.

3.11.3. Av Arama

Genel veya yerel aramalardan hangisinin gerçekleşeceğine $\vec{A}$ vektör değerine göre karar verilmektedir. $\vec{A} > 1$ veya $\vec{A} < -1$ olduğu durumlarda genel arama

gerçekleşmektedir. Çünkü bu durumlarda en iyi noktalardan daha uzak noktalar seçilebilmektedir. Av aramanın matematiksel formülü denklem 3.32 ve 3.33’de gösterilmektedir.

$$\vec{D'} = \vec{C} \cdot \vec{X}rand - \vec{X} \quad (3.32)$$

$$\vec{X}(t + 1) = \vec{X}rand - \vec{A} \cdot \vec{D'} \quad (3.33)$$

$\vec{X}rand$, rastgele seçilen bir arama ajanını göstermektedir.

Literatürde BOA’nın birçok özellik seçimi, sınıflandırma, kümeleme ve yapay sinir ağı çalışmalarında kullanıldığı ve başarılı sonuçlar verdiği görülmektedir. BOA’nın kullanıldığı veri seti çalışmalarına örnek olarak yüksek benzerlik gösteren Enron-Spam veri seti üzerinde yapılan çalışma gösterilebilmektedir. Bu çalışma da, BOA ile karşılaştırma işlemi için RBF yöntemlerinden faydalanılmıştır. Elde edilen sonuçlara göre BOA’nın ürettiği değerlerin daha iyi sonuç verdiği görülmüştür [103]. BOA yöntemi, verileri kümelemek içinde kullanılmıştır. Veri seti olarak karaciğer için çekilmiş MR görüntüleri seçilmiştir. Yapılan deneysel testler sonucunda BOA yönteminin yüksek oranda doğruluk sonucu elde ettiği görülmüştür [104]. BOA yöntemi yapay sinir ağlarını eğitmek için de kullanılmıştır. 20 farklı veri seti üzerinde, geri yayımlı öğrenme algoritması ve altı evrimsel algoritma ile eğitim işlemi gerçekleştirilmiştir. Yapılan deneysel testler sonucunda BOA yönteminin hem yerel optimal değerinden kaçınma hem de yakınsama hızı açısından daha üstün performansa sahip olduğu görülmüştür [105].

Bahsedilen avantajlardan dolayı önerilen modelde, ESA algoritmalarından çıkarılan özellikler BOA optimizasyon algoritmasına sokularak belirlenen mevcut en iyi özellikler seçilmiştir.

3.12. Kİ- KARE TESTİ İLE ÖZELLİK SEÇME

Elde edilen özelliklerden en ağırlıklı olanları seçen ve literatürde en sık kullanılan özellik seçim yöntemlerinden birisidir. Ki-kare testi, gerçek frekans ile beklenen

frekansın anlamlı olup olmadığına bakılmaktadır. Ki-kare yöntemi kategorik veriler için kullanılmaktadır ve veriler yüzdesel olarak verilmişse bu yöntemin kullanılması mümkün olmamaktadır [106]. Ki-kare ile özellik seçimi yapılırken χ^2 ile hesaplanan özellik değerleri büyükten küçüğe doğru sıralanır ve en üstten başlanarak istenilen sayıda özellik seçilmektedir [107]. Ki-kare testinin matematiksel ifadesi denklem 3.34’de verilmiştir [108].

$$\chi^2 = \sum \frac{(O-E)^2}{E} \quad (3.34)$$

Verilen denklemde gerçek frekans O , beklenen frekans ise E olarak gösterilmiştir. Önerilen modelde çalışmanın başarısını arttırabilmek için oluşturulan öznitelik havuzuna Ki-kare yöntemi uygulanmış ve en iyi özellikler seçilmiştir.

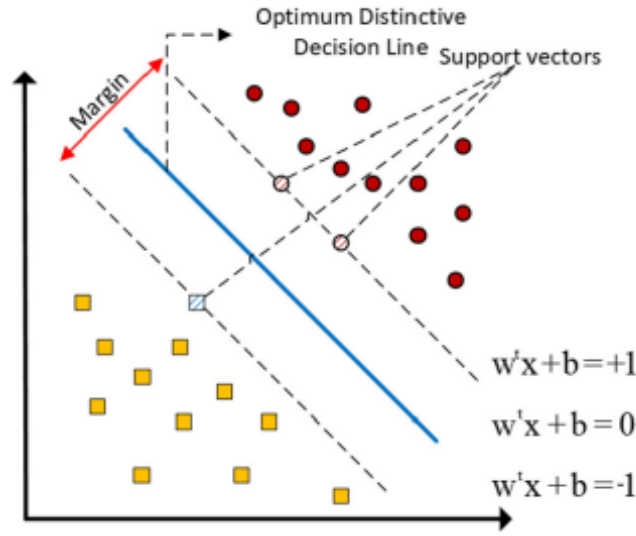
3.13. DESTEK VEKTÖR MAKİNELERİ

Destek Vektör Makineleri (DVM) problemleri çözebilmek için kullanılan denetimli bir makine öğrenme algoritmasıdır [109].

DVM noktaları ayırmak için düzlemde bir çizgi çizer ve bu çizgi iki nokta arasında ki maksimum mesafede olmayı amaçlar. Bu düzleme hiper-düzlem denilmektedir ve noktaları en iyi ayıran düzlemdir. Denklem 3.35’de DVM düzlem formülü verilmiştir [110].

$$f(x) = \sum_{i=1}^n (w_i x_i) + b = 0 \quad (3.35)$$

Denklem 4.20’ de düzlem uzayında ki değişkenler x , ağırlık değeri w ve bias değeri b ile temsil edilmektedir. Genel bir DVM sınıflandırma mimari örneği Şekil 3.29’da gösterilmiştir.



Şekil 3.29. DVM ile sınıflandırma örneği [111].

DVM'nin doğrusal olarak ayıramadığı problemlerde, çekirdek (kernel) fonksiyonları kullanılır. Bu fonksiyonlar sayesinde daha yüksek boyutta ki uzayda veriler sınıflandırılmaya çalışılır [112]. Literatürde sıklıkla kullanılan çekirdek fonksiyonlarına, doğrusal (linear), kuadratik (quadratic), kübik (cubic) çekirdek fonksiyonları örnek olarak gösterilebilmektedir [113].

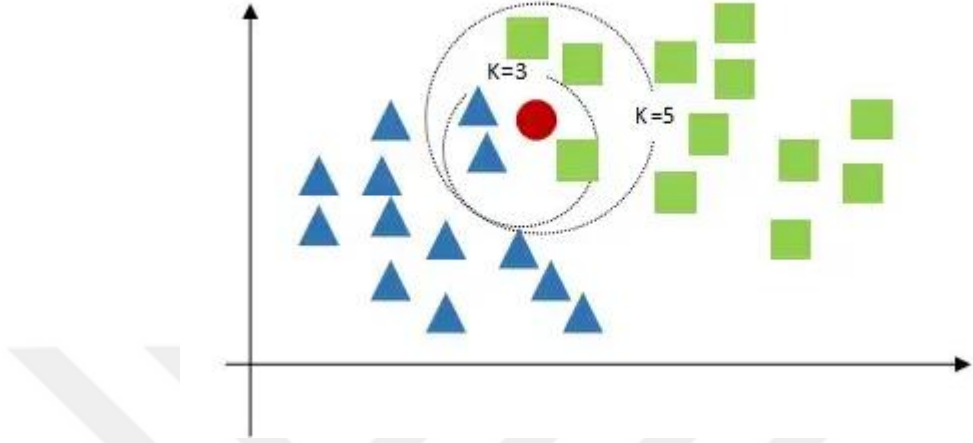
Önerilen modellerde hem orijinal hem de arttırılmış veri setleri üzerinde yapılan sınıflandırma işlemlerinde DVM yöntemi kullanılmıştır.

3.14. EN YAKIN KOMŞU ALGORİTMASI

K En Yakın Komşu Algoritması (KNN) ilk olarak, Evelyn Fix ve Joseph L. Hodges Jr. tarafından 1950'li yılların başlarında geliştirilmiştir. 1960'lı yıllarda ise N.J. Nilsson ve Thomas M. Cover tarafından yapılan mesafe tabanlı sınıflandırma çalışmaları ile gelişimi hız kazanmıştır [113][115].

KNN, hangi sınıfa ait olduğu bilinmeyen verilerin, eğitim setinde kullanılan veriler ile bir uzaklık ölçümü aracılığıyla karşılaştırılarak, uzaklık mesafesine göre en uygun sınıfa atanmasıdır [116]. K değeri sınıflandırma yapılacak olan verinin, kaç adet veriye bakılarak sınıfının belirleneceğini göstermektedir. Uzaklık ölçümleri için Manhattan,

Euclidian, Chebyshev, Cityblock gibi matematiksel yöntemler kullanılmaktadır [117]. KNN sınıflandırılmasının çalışma mantığı Şekil 3.30'da verilmiştir.



Şekil 3.30. KNN sınıflandırma örneği.

Şekil 4.8'de görüldüğü gibi K değeri 3 seçilirse kırmızı olan verinin sınıfı mavi, K değeri 5 seçilirse kırmızı olan verinin sınıfı yeşil olmaktadır.

KNN sınıflandırma algoritması basit ve takibi kolaydır. Ancak veri sayısı ve adım sayısı artarsa, maliyeti ciddi şekilde artmaktadır. Ayrıca verimlilik için uygun parametre seçimi oldukça zordur [118],[119].

BOA ile özellik seçimi yapılan modelde, DVM'nin yanı sıra KNN ile de sınıflandırma işlemi yapılmıştır. KNN ile yapılan sınıflandırma işleminde sonuçların daha başarılı çıktığı görülmüştür.

3.15. PERFORMANS ÖLÇÜTLERİ

Veri setimizde 18 farklı kategori bulunduğu için, çoklu sınıflandırma yapılmıştır. Bu tür sınıflandırmada, karmaşıklık matrisinde (confusion matrix) elde edilen değerler kullanılarak, Doğru Pozitif (DP), Doğru Negatif (DN), Yanlış Pozitif (YP) ve Yanlış Negatif (YN) gibi endeksler bulunmaktadır.

Burada DP, her bir kategoride doğru sınıflandırılan görüntülerin sayısını, DN ise ilgili kategori dışındaki diğer tüm kategorilerde doğru sınıflandırılan görüntülerin toplamını temsil etmektedir. YN, ilgili kategoriden yanlış sınıflandırılmış görüntülerin sayısını vermektedir. YP, ilgili kategori dışındaki tüm diğer kategorilerde yanlış sınıflandırılan görsellerin sayısını vermektedir.

Çalışmamızda kullanılan performans ölçütleri, doğruluk (accuracy), duyarlılık (sensitivity), özgüllük (specificity), kesinlik (precision), hassasiyet (recall), ve f-skor (f-score) şeklindedir.

Duyarlılık, doğru tahmin edilen pozitiflerin tüm gerçek pozitiflere oranıdır. Özgüllük, doğru tahmin edilen negatiflerin tüm gerçek negatiflere oranıdır. Doğruluk, tüm örnekler arasından doğru sınıflandırılmış örneklerin oranını gösterir. Kesinlik, tüm pozitif tanımlamalardan doğru tahmin edilen pozitiflerin oranıdır. Ayrıca Hassasiyet pozitif olarak tahmin etmemiz gereken işlemlerin ne kadarını pozitif olarak tahmin ettiğimizi gösteren bir metriktir. Hassasiyet duyarlılık ile aynı formüle sahiptir. F-skor değeri bize kesinlik ve hassasiyet değerlerinin harmonik ortalamasını göstermektedir. Bahsedilen performans hesaplama metrikleri için gerekli hesaplamalar aşağıdaki denklemlerde gösterilmektedir [120].

$$\text{Özgüllük} = \frac{DN}{DN+YP} \quad (3.36)$$

$$\text{Duyarlılık} = \frac{DP}{DP+YN} \quad (3.37)$$

$$\text{Doğruluk} = \frac{DP+DN}{DP+YN+DN+YP} \quad (3.38)$$

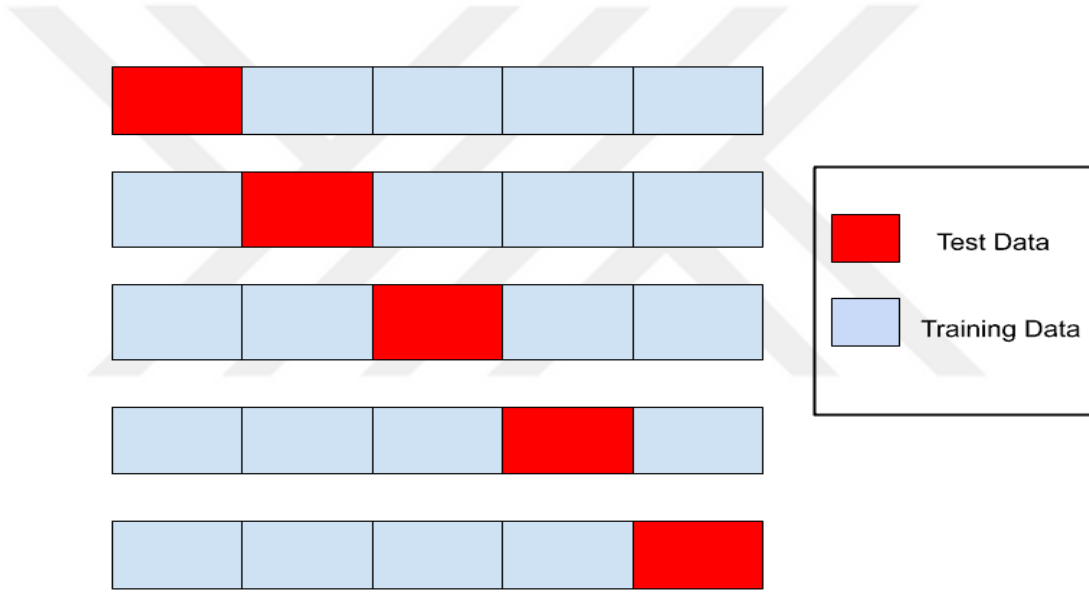
$$\text{Kesinlik} = \frac{DP}{DP+YP} \quad (3.39)$$

$$\text{Hassasiyet} = \frac{DP}{DP+YN} \quad (3.40)$$

$$F - \text{skor} = 2 * \frac{\text{Kesinlik} * \text{Hassasiyet}}{\text{Kesinlik} + \text{Hassasiyet}} \quad (3.41)$$

3.16. APRAZ DOĞRULAMA

Çapraz Doğrulama (Cross Validation), derin öğrenme çalışmalarında uygulanan yöntemin başarısının sınanması için kullanılan yöntemlerden birisidir [121]. Çapraz doğrulama, verileri bölerek karşılaştırma işlemi yapan matematiksel bir yöntemdir. Birbirinden farklı metotları bulunur ve bunlardan birisi K- Parçalı Doğrulama (k-fold validation) yöntemidir. Bu yöntemde veriler birbirine eşit k parçaya ayrılır ve her iterasyonda farklı bir parça doğrulama için kullanılmaktadır. Geri kalan k-1 parça ise eğitimde kullanılır. Bu şekilde k adet eğitim ve doğrulama gerçekleştirilmektedir [122]. Şekil 3.31’de 5-parçalı doğrulamanın çalışma sistemi anlatılmıştır.



Şekil 3.31. 5-parçalı doğrulama çalışma örneği.

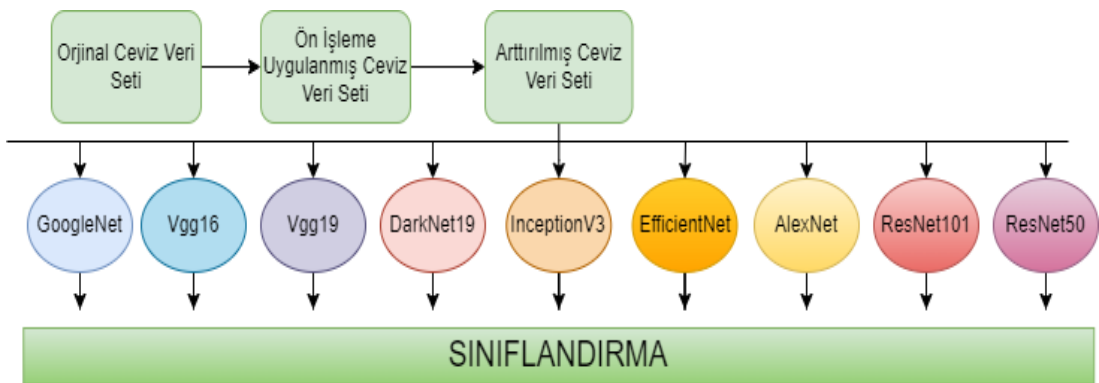
Yapılan deneysel testlerde, test güvenliğinin sağlanabilmesi amacıyla çapraz doğrulama yöntemi uygulanmıştır.

BÖLÜM 4

ÖNERİLEN DERİN ÖĞRENME MODELLERİ

4.1. ESA MİMARİLERİ İLE SINIFLANDIRILMA (ESA-9)

ESA modelleri ile deneysel testlere başlamadan önce başlık 3.6’da bahsedilen ön işleme ve başlık 3.7’de anlatıldığı gibi veri arttırma yöntemleri ile ön işlemeli ve ön işlemeli arttırılmış olarak 2 adet yeni veri seti elde edilmiştir. Ön işleme işlemleri sona erdikten sonra deneysel testlerin daha verimli çalışması için görüntüler 600 x 600 boyutlarına ölçeklendirilmiştir. Deneysel testlerin yapıldığı ilk veri seti 1751 adet ön işleme uygulanmış ve 600 x 600 boyutuna ölçeklendirilmiş yaprak görüntüsünden oluşmaktadır. Bu veri setine veri arttırma yöntemleri uygulanarak yaklaşık 4 kat arttırılarak arttırılmış veri seti elde edilmiştir. Hem ilk veri seti hem de arttırılmış veri seti literatürde yaygın olarak kullanılan 9 farklı ESA metodu ile ayrı ayrı sınıflandırılmıştır. Bu çalışmaya ESA-9 adı verilmiştir ve tezin geri kalanında bu isimle kullanılacaktır. Çalışmanın akış şeması Şekil 4.1’de gösterilmiştir.



Şekil 0.1. ESA-9’un akış şeması.

4.2. GRAD-CAM TABANLI GÖRÜNTÜLERİN VGG16 ESA MODELİ İLE GÖRSELLEŞTİRİLEREK SINIFLANDIRILMASI (GRADCAM-VGG16)

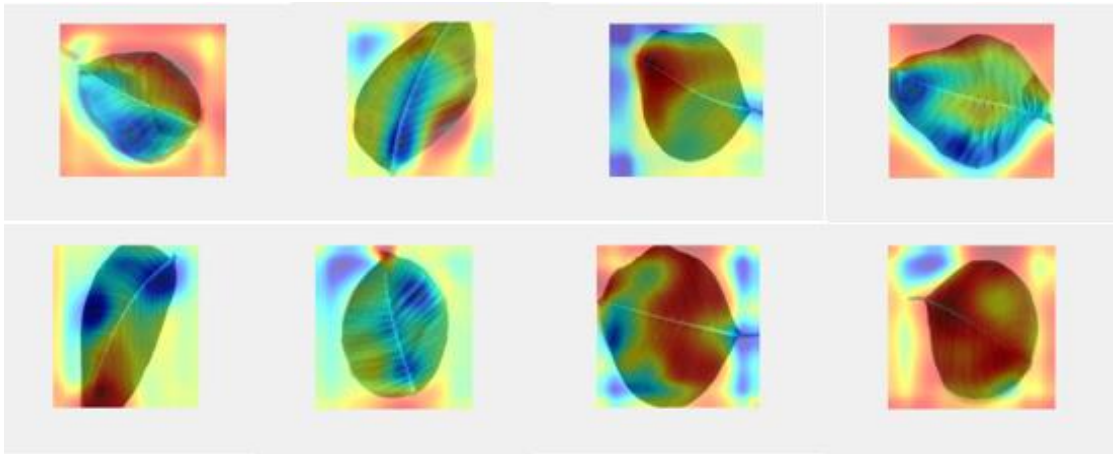
Literatürde yaygın olarak kullanılan ESA algoritmaları ile yapılan ESA-9 çalışmasından elde edilen test sonuçlarında, en yüksek doğruluk oranına sahip olan ESA algoritması Vgg16 olarak bulunmuştur.

Önerilen bu modelde, görüntülerin en önemli özelliklerinin seçilmesi için oluşturulacak Grad-Cam görüntülerinin elde edilmesinde, literatürde yaygın kullanılan ESA modellerinden ceviz veri setinin sınıflandırılmasında en yüksek sonucu veren Vgg16 ESA modeli kullanılmıştır.

Grad-Cam, görüntülerde ki en belirgin özellikleri ısı haritaları kullanarak renklendirir ve böylece ayırt edici özellikler ön plana çıkmış olmaktadır [90].

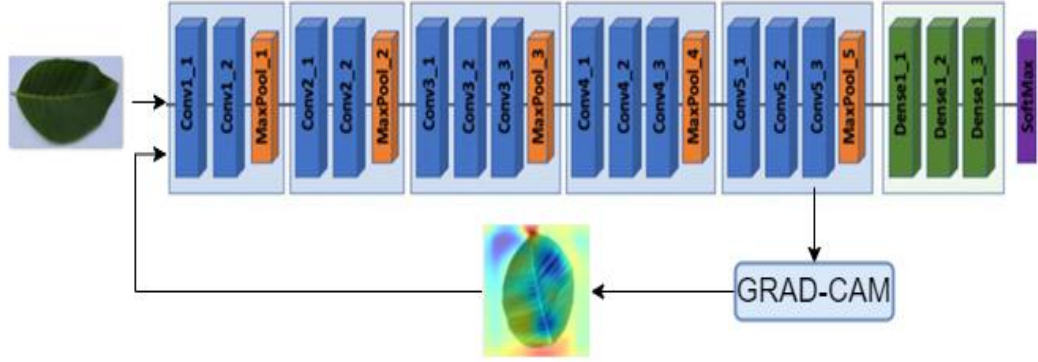
Vgg16 ile Grad-Cam görüntüleri elde edilirken Vgg16'nın birçok katmanından en önemli özellikler belirlenebilmektedir. Literatürde yaygın olarak kullanılan katman son evrişim katmanı olan conv5_3 katmanı olduğu için bu çalışmada aynı katmandan alınan özellikler kullanılmıştır.

Ön işleme uygulanmış ve arttırılmış ceviz veri setinden oluşturulan Grad-Cam görüntü örnekleri Şekil 4.2'de gösterilmiştir.



Şekil 0.2. Ceviz yapraklarının Grad-Cam görüntü örnekleri.

Oluşturulan yeni Grad-Cam uygulanmış ceviz görüntülerinin sınıflandırılması için, yine orjinal ceviz veri setinde en yüksek sonucu vermiş olan Vgg16 ESA modeli kullanılmıştır. Grad-Cam ve Vgg16 ESA mimarisinin kullanıldığı bu çalışmaya GRADCAM-VGG16 ismi verilmiştir ve tezin geri kalanında bu isimle kullanılacaktır. GRADCAM-VGG16 metodunun akış şeması Şekil 4.3’de gösterilmiştir.



Şekil 0.3. GRADCAM-VGG16 modeli akış şeması.

4.3. GLCM, HAFİF AĞIRLIKLIL DERİN ÖĞRENME MODELLERİ ve Kİ-KARE TABANLI MODEL ÖNERİSİ (GLCM-KİKARE-ESA)

Önerilen model temel olarak 3 aşamadan oluşmaktadır. Bunlar derin özelliklerin çıkarılması, çıkarılan özelliklerden en iyilerinin seçilmesi ve seçilen en iyi özelliklerden sınıflandırma işleminin gerçekleştirilmesidir.

İlk olarak ön işlem yöntemleri uygulanmış olan artırılmış veri seti literatürde popüler olan SqueezeNet ve MobilNetV2 ile eğitilmiştir. Bu iki ESA modeli hafif ağırlıklı (lightweight) derin öğrenme modeli olarak literatüre girmiştir. Hafif ağırlıklı ESA modelleri daha az parametre ile daha hızlı ve daha kararlı eğitimler yapabilmektedir. ESA modellerinin çok iyi birer özellik çıkarma yöntemi olması ve hafif ağırlıklı modellerin bahsedilen avantajlarından dolayı önerilen modelde SqueezeNet ve MobilNetV2 ESA mimarileri tercih edilmiştir.

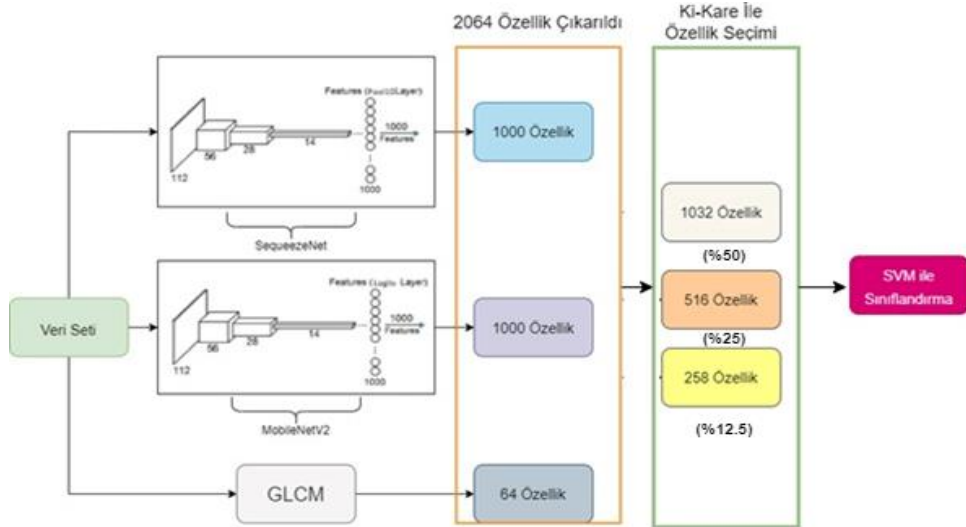
Eğitilen MobilnetV2’nin tam bağlantılı katmanı olan Logits katmanından 1000 özellik, SqueezeNet’in tam bağlantılı katmanı olan Pool10 katmanından 1000 özellik ve dokusal özellik çıkarma algoritması olan GLCM’den 64 özellik çıkarılmıştır.

GLCM algoritması ESA'lerden farklı olarak bazı dokusal özelliklerinde katkı sağlaması için modele eklenmiştir. Burada amaç geleneksel ve yeni teknolojilerin avantajlarından yararlanılarak hibrit ve yüksek doğruluk oranına sahip özgün bir model önermektir. Geleneksel ve ESA mimarilerinden her bir görüntüye ait 2064 özellik elde edilerek görüntülere ait detaylı bir özellik seti oluşturulmuştur.

İkinci aşamada, modelin başarısını arttırabilmek için oluşturulan özellik havuzuna Ki-kare yöntemi eklenerek en iyi özellikler seçilmiştir. Ki-Kare yöntemi, özelliklerin sınıflandırma amacına göre önem derecesi bakımından değerlerini hesaplar ve özellikleri ayırt edicilik açısından sıralar. Az sayıda öznitelik ile yüksek doğruluk oranı elde etmek için farklı sayıda öznitelik seçilmiştir. Özellik sayısı azaltılarak performans metriklerini arttırırken hesaplama maliyeti de düşürülmüştür. Burada denklem 3.34'de verilen formülde ki χ^2 değerine göre özellikler büyükten küçüğe doğru en üst sıradan başlanarak sırasıyla en etkili 1032, 516, 258 öznitelik seçilmiştir. Buda toplam özellik sayısının sırasıyla %50, %25 ve %12.5'ine tekabül etmektedir.

Son aşamada ise oluşturulan 1032, 516 ve 258 özellik DVM'ye giriş olarak verilerek sınıflandırma yapılmıştır. Bu çalışmaya GLCM-KİKARE-ESA ismi verilmiş ve tezin geri kalanında bu isim ile kullanılacaktır.

Önerilen modelin çalışması Şekil 4.4'de gösterilmiştir.



Şekil 0.4. GLCM-KİKARE-ESA modelinin şeması.

4.4. CEVİZ ÇEŞİTLERİNİN ÖNERİLEN ARTIK BLOK TABANLI ESA MODELİ İLE SINIFLANDIRILMASI (RES-AAO)

18 farklı ceviz yaprağı türü ile ceviz çeşidinin belirlenmesi için oluşturduğumuz özgün veri setinde ön işleme yapılmadan veri seti sınıflandırma için kullanılmıştır. Sınıflandırma işleminin yapılması için önerilen model 5 aşamadan oluşmaktadır.

Birinci aşamada Residual Blok tabanlı ResNet ESA modellerinden Resnet18, Resnet50 ve Resnet101 ile veri seti eğitilerek sınıflandırılmıştır. ResNet ESA mimarilerinde kullanılan artık bloklar benzer özelliklerin kolaylıkla öğrenilmesini ve farklı özelliklere odaklanılmasını sağlamaktadır [77].

İkinci aşamada her bir ESA modelinin tam bağlantılı katmanı olan Fc1000 katmanından 1000'er adet özellik çıkarılmıştır. Özellik çıkarma işlemi için ESA modelleri oldukça fazla kullanılmaktadır. Sınıflandırma katmanından önceki katman olan Tam Bağlantılı Katmanda ESA algoritması tüm işlemlerini bitirdiği için bu katmandan özellik çıkarmak avantajlıdır.

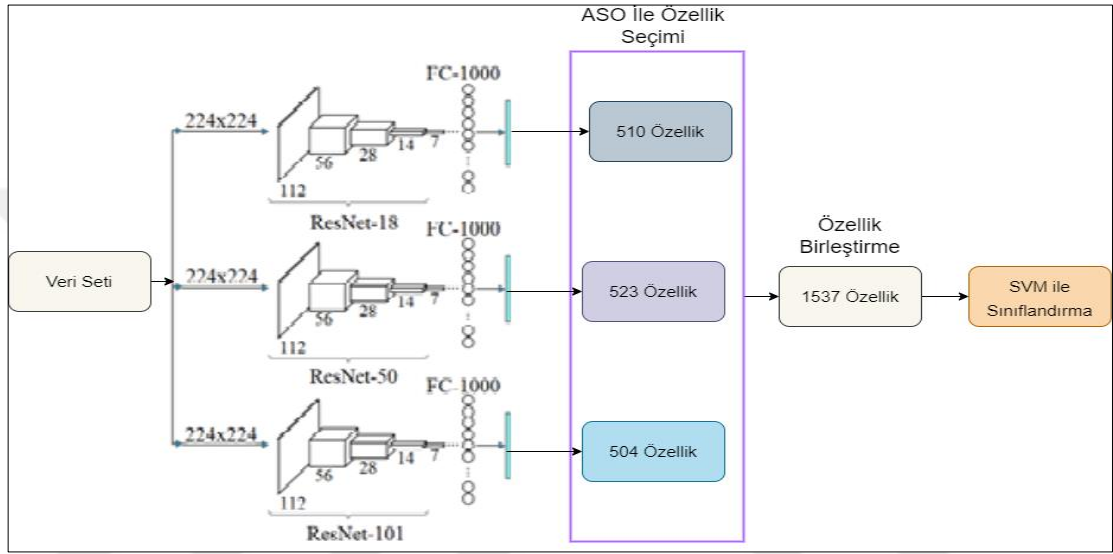
Üçüncü aşamada, sınıflandırma performansını arttırmak için, AAO algoritması özellik seçmek için kullanılmıştır. Her bir ESA modelinden elde edilen özellikler, AAO algoritmasına giriş olarak verilerek eğitilmiş ve etkili olan özellikler belirlenmiştir. Burada ResNet18, ResNet50 ve ResNet101 için sırasıyla, 510, 523 ve 504 adet özellik en iyi özellikler olarak AAO ile seçilmiştir. AAO algoritması, kısıtlı problemlerin çözümünde ve özellik seçim işlemlerinde yüksek başarı gösteren popülasyon tabanlı bir optimizasyon algoritmasıdır [97],99]. Bu yüzden özellik seçim işleminde AAO algoritması tercih edilmiştir.

Dördüncü aşamada belirlenmiş olan etkili özellikler ayrı ayrı DVM ile sınıflandırılmıştır.

Son aşamada ise AAO optimizasyon algoritması ile elde edilen özellikler birleştirilerek yeni bir özellik seti elde edilmiştir. Bu özellik setinde toplam 1537 adet özellik bulunmaktadır. Oluşturulan yeni özellik seti DVM'nin Lineer, Quadratic ve

Cubic çekirdekleri ile sınıflandırılmıştır. Ayrıca model güvenilirliğini arttırmak için 5-fold çapraz doğrulama (cross-validation) testi uygulanmıştır. Önerilen bu modele, uygulanan yöntemlerin kısaltması olarak RES-AAO ismi verilmiş ve tezin geri kalanında bu model ile kullanılacaktır.

Önerilen modelin akış diyagramı Şekil 4.5’de gösterilmiştir.



Şekil 0.5. RES-AAO modelinin çalışma şeması.

AAO’nun sözde kodu Şekil 4.6’da verilmiştir.

```
Başlangıç parametrelerini gir ( X, V, FitBest)

While ( Durma Kriteri )
    For Herbir X atomu için
        Fitnes değeri hesapla ( Fit )

        if Fiti<FitBest
            FitBest=Fit;
            Xbest= X ;
        End if

        K komşusunu belirle denklem 3.23
        En iyi atoma hareketi (Q) ve Kısıtlama kuvvetini (G) hesapla denklem 3.20
        Hız vektörünü hesapla (V) denklem 3.21
        Pozisyonu hesapla (X) denklem 3.22
    End for
End While
En iyi çözümü bul (Xbest)
```

Şekil 0.6. AAO sözde kodu.

Özellik seçimi için kullanılan AAO algoritmasına ait parametreleri aşağıdaki gibidir;

- Problem Boyutu (Problem Dimension), $D=1000$
- Çözüm Boyutu Sayısı (Number of Solution Size), $N=10$
- Popülasyon Matrisi (Population Matrix), $X(N, D)$, matristeki her bir eleman x_i^d 'dir
- En büyük tekrar sayısı (Maximum number of iterations), $MaxItr = 100$
- İşlev değerlendirmelerinin sayısı (Number of function evaluations) : $NFE = MaxItr(100) * N(10) = 1000$
- Doğrulama verilerinin oranı (Ratio of validation data) $ho=0.3$
- Eşik değer (Threshold Value), $th = 0.5$
- Derinlik ağırlığı (Depth weight), $alpha = 50$
- Çarpan ağırlığı (Multiplier weight), $beta = 0.2$

Başlangıç $X(N, D)$ çözüm matrisi, $0 < x_i^d < 1$ değerleri arasında rastgele elde edilmiştir. $x_i^d > th(0,5)$ durumunda seçilmiş 1 aksi durumda seçilmemiş 0 olarak alınıp uygunluk (fitness) değerleri hesaplanmıştır. AAO algoritması ile her bir döngüde (iterasyonda) geliştirilen çözüm matrisi değerlerinin uygunluk değerleri (fitness) KNN algoritması kullanılarak hesaplanmıştır.

Uygunluk değeri (fitness) hesaplamak için kullanılan KNN algoritmasına ait parametreler aşağıdaki gibidir;

- KNN numarası (Number of k in K-nearest neighbor) $k = 5$
- $alpha = 0.99$,
- $beta = 0.01$

Uygunluk değeri (fitness) hesaplama formülasyonu denklem 4.11 ve 4.2 ve 4.3' de gösterilmiştir.

$$ErrorRate = 1 - ACC \quad (4.1)$$

$$SF = \sum_{i=0}^D x_i = 1 \quad (4.2)$$

$$Fitnes Function = \alpha * ErrorRate + \beta * (SF/D) \quad (4.3)$$

Burada ACC seçilen özelliklerin KNN sınıflandırmasında elde edilen doğruluk oranı, α (0.99), β (0.01) değerlerine bağlı olarak hesaplanarak elde edilmiştir. *Selected Features (SF)* ise seçilen özelliklerin toplamını ifade etmektedir.

4.5. BALİNA OPTİMİZASYON ALGORİTMASI İLE ESA ÖZELLİKLERİNİN SEÇİLEREK CEVİZ VERİ SETİNİN SINIFLANDIRILMASI (BOA-ESA-KNN)

Oluşturulan özgün veri seti üzerinde yapılan ilk uygulamada orijinal ceviz görüntüleri üzerinde ön işleme yapıldıktan sonra, veri arttırma yöntemleri ile görüntü sayısı yaklaşık dört kat arttırılarak 6843 adet görüntü elde edilmiştir. Arttırılmış veri setine literatürde sıklıkla kullanılan ESA algoritmaları uygulanarak deneysel test sonuçları alınmıştır. Bu deneysel test sonuçlarında, doğruluk oranı en yüksek olan üç ESA modelinin Vgg16, Vgg19 ve AlexNet olduğu görülmektedir.

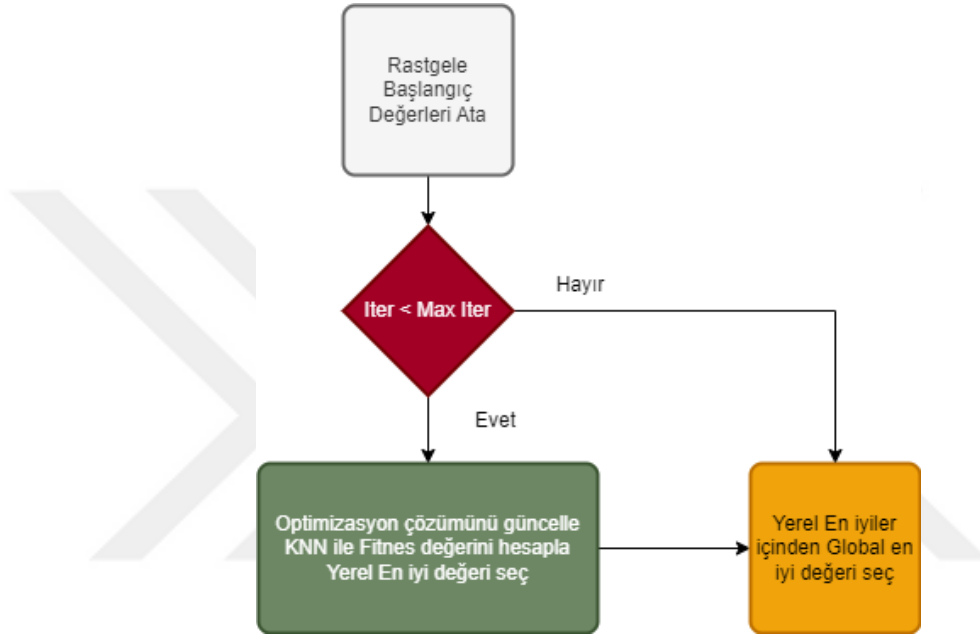
Çalışmada en yüksek doğruluk oranını veren ESA modellerinden özellik çıkarılarak yeni bir model üretilmiştir. Modelde Vgg16, Vgg19 ve AlexNet'in tam bağlantılı katmanı olan fc8 katmanından 1000'er adet özellik çıkarılmıştır. Çıkarılan özellikler birleştirilerek, öncelikle yeni bir öznitelik havuzu oluşturulmuş ve KNN ile sınıflandırılmıştır.

ESA algoritmalarından çıkarılan özellikler BOA optimizasyon algoritmasına sokularak belirlenen mevcut en iyi özellikler seçilmiştir. BOA ile özellik seçimi gerçekleştirilirken kullanılan parametreler Çizelge 4.1'de gösterilmiştir.

Çizelge 0.1. BOA özellik seçiminde kullanılan parametreler.

Parametre	Değer
Problem Boyutu (D)	1000
Popülasyon Büyüklüğü	10
Komşu Sayısı	5
Eşik Değer	0.5

BOA, birçok özellik seçim yönteminde kullanılmaktadır ve hem yerel en iyi değerden kaçınmada hem de yakınsama hızı açısından oldukça başarılı sonuçlar vermektedir [102-104]. Bu yüzden özellik seçim aşamasında BOA tercih edilmiştir. BOA ile özellik seçimi yapılırken kullanılan parametreler ve formüller RES-AAO modelinde özellik seçimi için kullanılan parametreler ve formüller ile aynıdır. BOA ile özellik seçiminin akış şeması Şekil 4.7 'de gösterilmiştir.



Şekil 0.7. BOA ile özellik seçiminin akış diyagramı.

Kullanılan özellik seçim algoritması olan BOA'nın sözde kodu Şekil 4.8'de verilmiştir.

```

Başlangıç parametrelerini gir ( X, i, N)
Herbir çözüm için fitnes hesapla (X*= en iyi çözüm)

While ( t< Max_iter )

    for Herbir çözüm için

        Güncelle a,A,C,I ve p

        if (p<0.5)

            if (|A|<1)
                Konumları güncelle denklem 3.25
            else if (|A|=>1)
                Rastgele çözüm seç ve hesapla denklem 3.33
            End if
        else if (p>=0.5)
            Konumları güncelle denklem 3.28
        End if
    End for

    En iyi değerleri hesapla
    t++
End While

En iyi çözüm X*

```

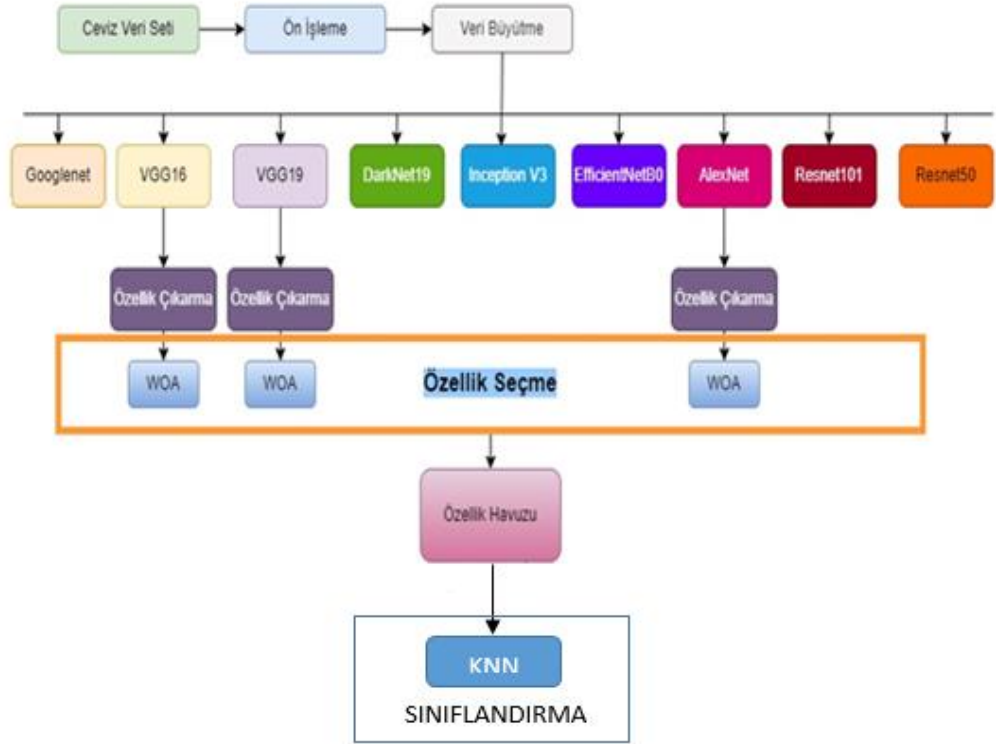
Şekil 0.8. BOA'nın sözde kodu.

BOA algoritması ile AlexNet'ten 708, Vgg16'dan 249 ve Vgg19'dan 777 adet özellik seçilmiştir.

Seçilen özellikler birleştirilerek ortak bir özellik havuzu oluşturulmuştur. Özellik havuzunda ki toplam özellik sayısı 1734'tür. Seçilen özellik havuzuna sınıflandırma işlemi için ve KNN sınıflandırma yöntemi uygulanmıştır.

Son olarak model güvenliğini sağlamak için 5 parçalı çapraz doğrulama testi uygulanmıştır. Bu çalışmaya BOA-ESA-KNN ismi verilmiş ve tezin geri kalanında bu isimle kullanılacaktır.

Önerilen çalışmanın aşamaları Şekil 4.9'da gösterilmiştir.



Şekil 0.9. BOA-ESA-KNN modelin çalışma aşamaları.

BÖLÜM 5

DENEYSEL ÇALIŞMALAR VE BULGULAR

5.1. ESA-9 MODELİNİN DENEYSEL TEST SONUÇLARI

Yapılan ESA-9 çalışmasında literatürde yaygın olarak kullanılan 9 farklı ESA modeli ile oluşturulan özgün ceviz veri seti ve arttırılmış veri seti sınıflandırılmıştır. Veri setleri rastgele olarak %70 oranında eğitim ve %30 oranında test setlerine bölünmüştür. Tüm optimizasyon yöntemleri için öğrenme oranı 0.001 olarak seçilmiştir.

Görüntüler eğitime başlamadan önce ESA giriş boyutları, AlexNet için 227×227 piksel, ResNet50, ResNet101, GoogleNet, EfficientNetB0, VGG16 ve VGG19 için 224×224 piksel, Inception V3 için 299×299 piksel, DarkNet19 için 256×256 piksel olarak ayarlanmıştır.

Bu deneysel çalışmada mini-batch size 16 olarak ayarlanmıştır. Çizelge 5.1’de ilk deneysel çalışmada gerçekleştirdiğimiz tüm deneylerde kullanılan hiper-parametreler gösterilmektedir.

Çizelge 0.1. ESA-9 modelinin deneysel testlerinde kullanılan hiper-parametreler.

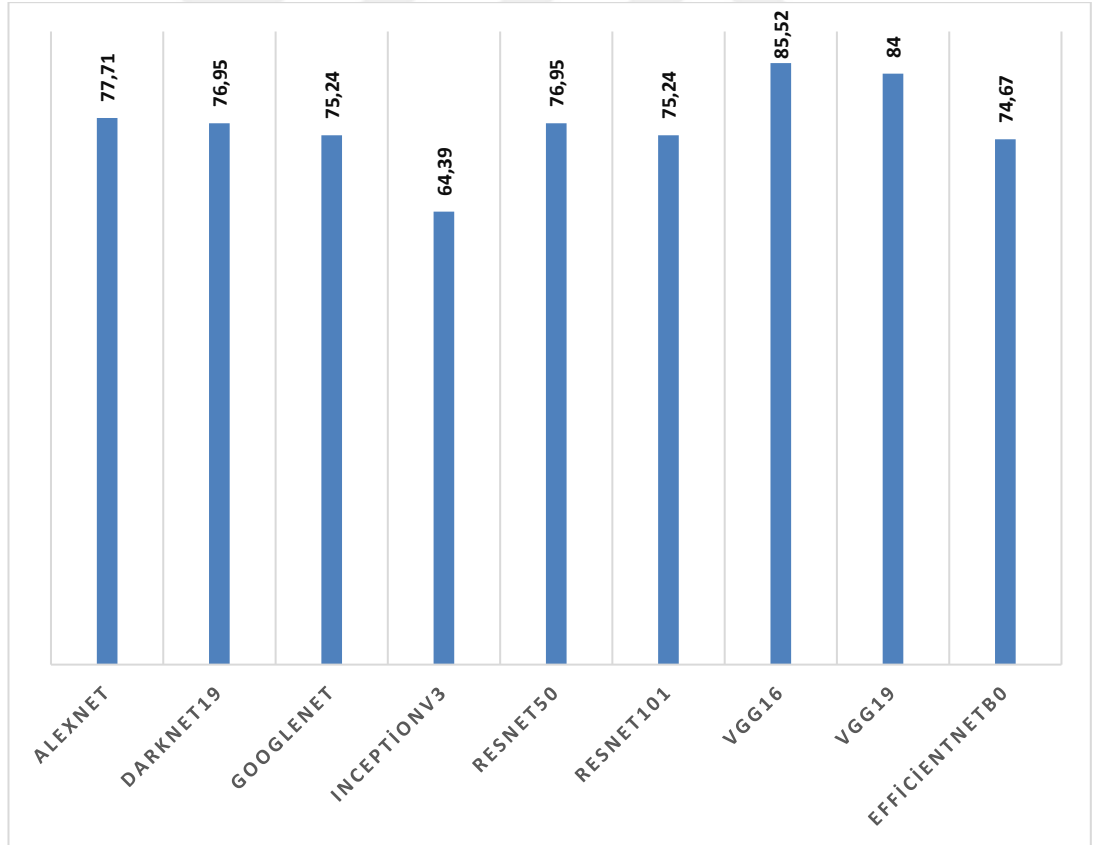
ESA Modeli	Görüntü Boyutu	mini-batch Boyutu	Öğrenme Oranı	Epoch
AlexNet	227 x 227	16	0.0001	32
VGG16	224 x 224			
VGG19	224 x 224			
DarkNet19	256 x 256			
Inception V3	229 x 229			
EfficientNetB0	224 x 224			
Googlenet	224 x 224			
Resnet101	224 x 224			
Resnet50	224 x 224			

Tüm deneysel çalışmalar hem orijinal hem de arttırılmış veri setleri üzerinde gerçekleştirilmiştir. Test veri seti bahsedilen tüm ESA modelleri ile test edilerek,

doğruluk, duyarlılık, özgünlük, kesinlik ve f-skör sonuçları bulunmuştur. Orijinal veri seti ile yapılan testler sonucunda bulunan değerler Çizelge 5.2’de gösterilmiştir. ESA modellerinin test sonuçlarının doğruluk oranlarının grafikleri Şekil 5.1’de verilmiştir.

Çizelge 0.2. ESA-9 çalışmasının deneysel test sonuçları.

ESA Modeli	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skör
AlexNet	0.7771	0.7514	0.9869	0.7757	0.7584
DarkNet19	0.7695	0.7362	0.9865	0.7448	0.7377
GoogleNet	0.7524	0.7251	0.9855	0.7330	0.7264
InceptionV3	0.6439	0.6112	0.9790	0.6309	0.6131
ResNet50	0.7695	0.7362	0.9865	0.7575	0.7425
ResNet101	0.7524	0.7197	0.9855	0.7411	0.7272
VGG16	0.8552	0.8315	0.9915	0.8300	0.8363
VGG19	0.8400	0.8137	0.9906	0.8220	0.8157
EfficientNetB0	0.7467	0.7230	0.9851	0.7334	0.7245



Şekil 0.1. ESA-9 çalışmasının test sonuçlarının doğruluk oranlarının grafikleri.

Çizelge 5.2’de görüldüğü şekilde, yapılan deneysel testler sonucunda %85.52 doğruluk oranıyla Vgg16 en yüksek başarıyı elde etmiştir. En yüksek başarıyı elde eden üç ESA algoritması Vgg16, Vgg19 ve AlexNet’tir. Bu ESA algoritmalarının karmaşıklık matrisleri sırasıyla Şekil 5.2, Şekil 5.3 ve Şekil 5.4’te verilmiştir.

Output Class	Frenquette	bilecik	chandler	fernette	fernor	hardley	howard	kaman1	kaplan86	lara	maya1	mitland	oguzlar77	pedro	sebin	sen	serr	yalova3
Frenquette	33 6.3%	4 0.8%	0 0.0%	0 0.0%	1 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 0.8%	0 0.0%	78.6%
bilecik	0 0.0%	21 4.0%	0 0.0%	0 0.0%	0 0.2%	1 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	5 1.0%	0 0.0%	0 0.0%	2 0.4%	0 0.0%	0 0.0%	72.4%
chandler	0 0.0%	0 0.0%	19 3.6%	0 0.0%	0 0.2%	1 0.2%	0 0.0%	1 0.2%	4 0.8%	5 1.0%	0 0.0%	0 0.2%	1 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	59.4%
fernette	3 0.6%	0 0.0%	1 0.2%	27 5.1%	4 0.8%	0 0.0%	0 0.0%	1 0.2%	4 0.8%	1 0.2%	0 0.0%	1 0.2%	1 0.2%	1 0.2%	0 0.0%	1 0.2%	6 1.1%	52.9%
fernor	0 0.0%	0 0.0%	1 0.2%	0 0.0%	12 2.3%	1 0.2%	1 0.2%	0 0.0%	0 0.2%	0 0.0%	0 0.0%	1 0.2%	0 0.4%	2 0.4%	0 0.0%	0 0.0%	0 0.0%	63.2%
hardley	0 0.0%	2 0.4%	1 0.2%	0 0.0%	5 1.0%	23 4.4%	2 0.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.4%	2 0.6%	3 0.6%	2 0.4%	5 1.0%	5 1.0%	45.1%
howard	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.4%	0 0.0%	13 2.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	86.7%
kaman1	0 0.0%	0 0.0%	2 0.4%	0 0.0%	7 1.3%	1 0.2%	5 1.0%	28 5.3%	2 0.4%	4 0.8%	3 0.6%	0 0.0%	1 0.2%	0 0.0%	0 0.0%	1 0.2%	0 0.0%	51.9%
kaplan86	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 0.8%	0 0.0%	8 1.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	33.3%
lara	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 0.8%	0 0.0%	1 0.2%	9 1.7%	1 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	60.0%
maya1	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	13 2.5%	5 1.0%	1 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	26.3%
mitland	2 0.4%	0 0.0%	1 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	42 8.0%	2 0.4%	1 0.2%	0 0.0%	1 0.2%	3 0.6%	80.8%
oguzlar77	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100%
pedro	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.2%	1 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.4%	16 3.0%	0 0.0%	0 0.0%	0 0.0%	94.1%
sebin	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	22 4.2%	0 0.0%	0 0.0%	100%
sen	0 0.0%	2 0.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.4%	0 0.0%	0 0.0%	0 0.0%	38 7.2%	2 0.4%	76.0%
serr	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.6%	3 0.6%	0 0.0%	0 0.0%	0 0.0%	15 2.9%	83.3%
yalova3	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100%
	86.8%	72.4%	76.0%	100%	38.7%	82.1%	50.0%	96.6%	16.0%	47.4%	22.7%	95.5%	11.1%	89.6%	84.6%	80.9%	41.7%	67.4%
	13.2%	27.6%	24.0%	0.0%	61.3%	17.9%	50.0%	3.4%	84.0%	52.6%	77.3%	4.5%	88.9%	30.4%	15.4%	19.1%	58.3%	32.6%

Şekil 0.2. Vgg16 modeli ile sınıflandırma sonucu elde edilen karmaşıklık matrisi.

Output Class	Frenquette	37	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	94.9%
		7.0%	0.0%	0.2%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	5.1%
	bilecik	0	28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
		0.0%	5.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	chandler	0	0	21	0	0	0	0	0	0	0	0	0	0	1	0	0	0	95.5%
		0.0%	0.0%	4.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	4.5%
	fernette	1	0	0	25	3	0	0	1	0	0	1	1	0	0	0	1	0	75.8%
		0.2%	0.0%	0.0%	4.8%	0.6%	0.0%	0.0%	0.2%	0.0%	0.0%	0.2%	0.2%	0.0%	0.0%	0.0%	0.2%	0.0%	24.2%
	fernor	0	0	0	1	24	0	1	1	0	0	0	1	1	0	0	0	0	82.8%
		0.0%	0.0%	0.0%	0.2%	4.6%	0.0%	0.2%	0.2%	0.0%	0.0%	0.0%	0.2%	0.2%	0.0%	0.0%	0.0%	0.0%	17.2%
	hardley	0	0	0	0	0	22	0	0	0	0	0	0	0	0	0	0	0	100%
		0.0%	0.0%	0.0%	0.0%	0.0%	4.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	howard	0	0	0	0	1	0	18	1	0	4	0	0	0	0	0	0	0	75.0%
		0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	3.4%	0.2%	0.0%	0.8%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	25.0%
	kaman1	0	0	0	0	2	0	2	26	1	0	0	0	0	0	0	0	0	83.9%
		0.0%	0.0%	0.0%	0.0%	0.4%	0.0%	0.4%	5.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	16.1%
	kaplan86	0	0	0	0	0	0	0	7	0	17	0	0	0	0	0	0	0	29.2%
		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	1.3%	0.0%	3.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	70.8%
	lara	0	0	1	0	0	0	4	0	15	0	0	0	0	0	0	0	0	75.0%
		0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.8%	0.0%	2.9%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	25.0%
	maya1	0	0	1	0	0	0	0	16	0	4	0	0	1	0	0	0	0	18.2%
		0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	3.0%	0.0%	0.8%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	81.8%
	mitland	0	0	1	0	0	0	1	0	0	0	41	0	0	0	0	0	0	95.3%
		0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	7.8%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	4.7%
	oguzlar77	0	0	0	0	0	0	0	0	0	0	0	15	0	1	0	0	0	93.8%
		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.9%	0.0%	0.2%	0.0%	0.0%	0.0%	6.3%
	pedro	0	0	0	1	0	0	0	0	0	0	0	0	20	0	0	0	0	95.2%
		0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%	4.8%
	sebin	0	0	0	0	0	1	0	0	0	0	0	1	1	25	0	0	0	89.3%
		0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.2%	4.8%	0.0%	0.0%	0.0%	10.7%
	sen	0	1	0	0	0	2	0	0	0	0	0	0	0	0	47	0	0	94.0%
		0.0%	0.2%	0.0%	0.0%	0.0%	0.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	9.0%	0.0%	0.0%	0.0%	6.0%
	serr	0	0	0	0	1	2	0	0	1	0	0	1	1	0	0	34	0	85.0%
		0.0%	0.0%	0.0%	0.0%	0.2%	0.4%	0.0%	0.0%	0.2%	0.0%	0.0%	0.2%	0.2%	0.0%	0.0%	6.5%	0.0%	15.0%
	yalova3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	32	97.0%
		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	6.1%	3.0%
		97.4%	96.6%	84.0%	82.6%	77.4%	78.6%	69.2%	89.7%	28.0%	78.9%	18.2%	93.2%	83.3%	87.0%	96.2%	100%	94.4%	100%
		2.6%	3.4%	16.0%	7.4%	22.6%	21.4%	30.8%	10.3%	72.0%	21.1%	81.8%	6.8%	16.7%	13.0%	3.8%	0.0%	5.6%	0.0%
																			16.0%

Şekil 0.3. Vgg19 modeli ile sınıflandırma sonucu elde edilen karmaşıklık matrisi.

Output Class	Frenquette	33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
		6.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	bilecik	0	25	1	0	0	1	0	0	0	0	0	3	1	0	0	1	0	78.1%
		0.0%	4.8%	0.2%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.6%	0.2%	0.0%	0.0%	0.2%	0.0%	21.9%
	chandler	0	0	17	0	1	0	0	0	1	0	0	1	0	2	0	0	0	77.3%
		0.0%	0.0%	3.2%	0.0%	0.2%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.2%	0.0%	0.4%	0.0%	0.0%	0.0%	22.7%
	fernette	1	0	1	22	1	0	0	0	0	0	0	0	0	0	0	0	0	88.0%
		0.2%	0.0%	0.2%	4.2%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	12.0%
	fernör	0	0	1	2	26	0	4	1	0	1	0	0	0	0	0	0	0	74.3%
		0.0%	0.0%	0.2%	0.4%	5.0%	0.0%	0.8%	0.2%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	25.7%
	hardley	0	1	0	1	0	23	0	0	0	0	0	1	0	2	0	0	0	82.1%
		0.0%	0.2%	0.0%	0.2%	0.0%	4.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.4%	0.0%	0.0%	0.0%	17.9%
	howard	0	0	0	2	0	0	12	0	0	1	0	0	0	0	0	0	1	75.0%
		0.0%	0.0%	0.0%	0.4%	0.0%	0.0%	2.3%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	25.0%
	kaman1	0	0	2	0	0	0	2	24	0	1	0	0	0	1	0	0	0	80.0%
		0.0%	0.0%	0.4%	0.0%	0.0%	0.0%	0.4%	4.6%	0.0%	0.2%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	20.0%
	kaplan86	0	0	0	0	1	0	1	0	3	2	20	0	0	0	0	0	0	11.1%
		0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.2%	0.0%	0.6%	0.4%	3.8%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	88.9%
	lara	0	0	1	0	1	0	4	4	0	11	0	0	0	0	0	0	0	52.4%
		0.0%	0.0%	0.2%	0.0%	0.2%	0.0%	0.8%	0.8%	0.0%	2.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	47.6%
	maya1	0	0	0	0	0	0	3	0	20	3	1	0	0	1	0	0	0	3.6%
		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.6%	0.0%	3.8%	0.6%	0.2%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	96.4%
	mitland	3	0	0	0	0	0	0	0	0	0	0	42	0	0	0	0	1	91.3%
		0.6%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	8.0%	0.0%	0.0%	0.0%	0.0%	0.2%	8.7%
	oguzlar77	0	0	0	0	0	0	0	0	0	0	0	13	0	1	3	0	0	76.5%
		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.5%	0.0%	0.2%	0.6%	0.0%	0.0%	23.5%
	pedro	0	0	1	0	0	0	0	0	1	0	1	0	0	16	1	0	0	80.0%
		0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.2%	0.0%	0.0%	3.0%	0.2%	0.0%	0.0%	20.0%
	sebin	0	0	0	0	0	0	0	0	0	0	0	1	0	23	0	1	0	92.0%
		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	4.4%	0.0%	0.2%	0.0%	8.0%
	sen	1	3	1	0	0	1	0	0	0	0	0	0	0	0	42	2	0	84.0%
		0.2%	0.6%	0.2%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	8.0%	0.4%	0.0%	16.0%
	serr	0	0	0	0	1	2	0	0	0	0	0	1	0	1	0	30	0	85.7%
		0.0%	0.0%	0.0%	0.0%	0.2%	0.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.2%	0.0%	5.7%	0.0%	14.3%
	yalova3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	2	0	31.4%
		0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.4%	0.0%	6.1%
		13.2%	13.8%	32.0%	18.5%	16.1%	17.9%	53.8%	17.2%	88.0%	42.1%	95.5%	4.5%	27.8%	30.4%	11.5%	10.6%	16.7%	0.0%
		86.8%	86.2%	68.0%	81.5%	83.9%	82.1%	46.2%	82.8%	12.0%	57.9%	4.5%	95.5%	72.2%	69.6%	88.5%	89.4%	83.3%	100%
																			75.2%
																			24.8%

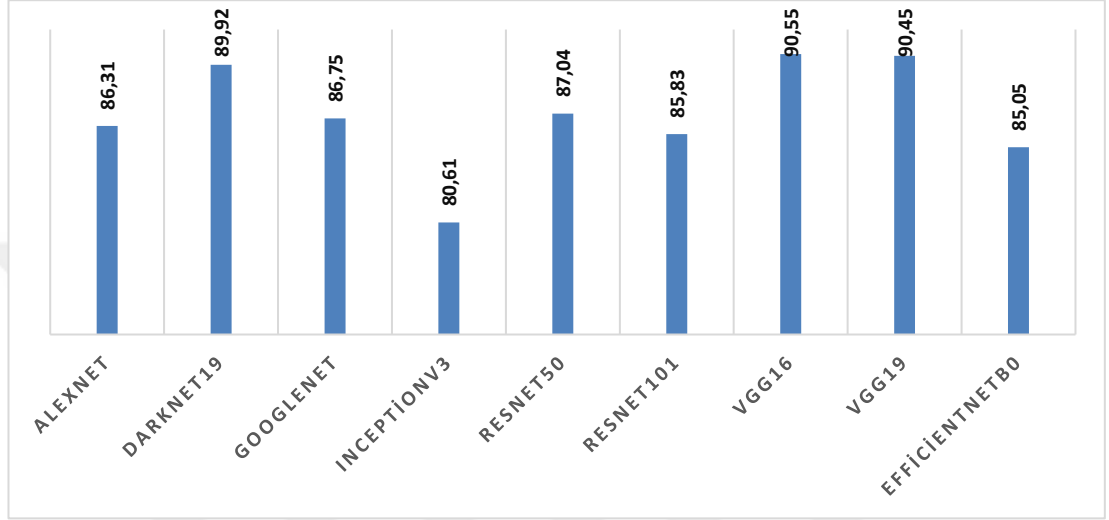
Şekil 0.4. AlexNet modeli ile sınıflandırma sonucu elde edilen karmaşıklık matrisi.

Çizelge 0.3. Arttırılmış veri seti ile ESA-9 modelinin deneysel test sonuçları.

ESA Modeli	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skör
AlexNet	0.8631	0.8504	0.9920	0.8543	0.8518
DarkNet19	0.8992	0.8872	0.9941	0.8878	0.8870
GoogleNet	0.8675	0.8512	0.9922	0.8605	0.8532
InceptionNetV3	0.8061	0.7919	0.9886	0.7968	0.7928
ResNet50	0.8704	0.8547	0.9924	0.8634	0.8577
ResNet101	0.8583	0.8424	0.9917	0.8520	0.8460
VGG16	0.9055	0.8900	0.9945	0.8894	0.8891
VGG19	0.9045	0.8918	0.9944	0.8941	0.8924
EfficientNetB0	0.8505	0.8341	0.9912	0.8408	0.8365

Çizelge 5.3'te görüldüğü şekilde, arttırılmış veri seti ile yapılan deneysel testler sonucunda % 90.55 doğruluk oranıyla orijinal veri setindeki gibi Vgg16 en yüksek başarıyı elde etmiştir. Ancak bazı performans ölçütlerinde Vgg19 daha iyi sonuçlar

elde etmiştir. Duyarlılık ve f-Score metriklerinde en iyi sonuçlar Vgg19 modelindedir. Ayrıca hemen hemen tüm metrikler de Vgg16 ve Vgg19 birbirine çok yakın sonuçlar üretmişlerdir. DarkNet19 da Vgg16 ve Vgg19 modellerine oldukça yakın performans göstermektedir. Arttırılmış veri seti ile ESA modellerinin test sonuçlarının doğruluk oranlarının grafikleri Şekil 5.3'te verilmiştir.



Şekil 0.5. Arttırılmış veri seti ile ESA-9 modelinin test sonuçlarının doğruluk oranlarının grafikleri.

Arttırılmış veri setinde yapılan deneysel test sonuçlarında en yüksek doğruluk oranına sahip olan ESA modellerinden Vgg16, Vgg19 ve DarkNet'in karışıklık matrisi sırasıyla Şekil 5.6, Şekil 5.7 ve Şekil 5.8'de gösterilmektedir.

	Frenquette	Bilecik	Chandler	Fernette	Fernor	Hardley	Howard	Kaman1	Kaplan86	Lara	Maya1	Mithland	Oguzlar77	Pedro	Sebin	Sen	Serr	Yalova3
Frenquette	144	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
Bilecik	0	106	0	0	1	0	0	0	0	0	0	0	0	0	0	2	1	3
Chandler	0	1	92	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0
Fernette	0	0	0	104	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Fernor	0	1	0	4	109	0	2	2	2	3	0	0	0	0	0	0	0	0
Hardley	0	0	3	0	0	103	0	2	0	0	0	0	2	1	0	1	0	0
Howard	0	0	0	1	3	0	94	1	0	1	0	0	0	0	0	0	0	0
Kaman1	0	0	0	0	0	0	1	113	0	0	0	0	0	1	0	0	0	0
Kaplan86	0	0	2	0	0	1	0	0	49	3	43	0	0	1	0	0	0	0
Lara	0	0	1	2	1	0	4	1	0	62	1	0	0	0	1	0	0	2
Maya1	0	0	2	0	0	0	1	0	44	1	40	0	0	0	0	0	0	0
Mithland	1	0	0	0	0	0	0	0	0	0	0	169	0	0	0	0	0	0
Oguzlar77	0	0	0	0	0	0	0	0	0	0	0	0	67	0	0	1	0	2
Pedro	0	1	4	1	0	3	1	0	1	0	0	0	0	79	1	0	0	0
Sebin	0	0	0	0	0	0	0	1	0	0	0	0	6	0	95	0	2	0
Sen	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	177	1	2
Serr	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	136	1
Yalova3	0	4	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	120

Şekil 0.6. Arttırılmış veri setinin Vgg16 karışıklık matrisi.

	Frenquette	Bilecik	Chandler	Fernette	Fernor	Hardley	Howard	Kaman1	Kaplan86	Lara	Maya1	Mithland	Oguzlar77	Pedro	Sebin	Sen	Serr	Yalova3
Frenquette	141	0	1	1	0	0	0	0	0	0	0	3	0	0	1	0	0	0
Bilecik	0	112	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Chandler	0	1	92	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0
Fernette	0	0	1	99	1	0	1	0	0	3	0	0	0	0	0	0	0	0
Fernor	1	0	0	2	106	2	3	8	0	0	0	0	0	0	0	0	1	0
Hardley	0	1	0	1	0	108	0	0	0	0	0	0	0	1	0	0	0	1
Howard	0	0	0	0	1	0	93	2	0	3	1	0	0	0	0	0	0	0
Kaman1	0	0	0	0	0	0	0	115	0	0	0	0	0	0	0	0	0	0
Kaplan86	0	0	0	1	0	0	0	1	52	0	44	0	0	0	0	0	1	0
Lara	0	0	0	4	1	0	4	0	0	83	1	0	0	0	0	0	1	1
Maya1	0	0	0	0	0	0	1	1	44	2	37	0	0	0	0	0	3	0
Mithland	1	0	0	0	0	0	0	0	0	0	0	187	0	0	0	0	2	0
Oguzlar77	0	0	0	0	0	0	0	0	3	0	0	0	85	0	2	0	0	0
Pedro	0	0	2	0	0	0	2	0	1	0	0	0	0	96	0	0	0	0
Sebin	0	0	0	0	0	0	0	0	0	0	0	0	0	0	103	0	1	0
Sen	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	174	1	6
Serr	2	0	3	0	1	0	0	0	1	0	0	3	0	0	0	3	124	1
Yalova3	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	120

Şekil 0.7. Arttırılmış veri setinin Vgg19 karışıklık matrisi.

	Frenquette	Bilecik	Chandler	Fernette	Fernor	Hardley	Howard	Kaman1	Kaplan86	Lara	Maya1	Mithland	Oguzlar77	Pedro	Sebin	Sen	Serr	Yalova3
Frenquette	145	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Bilecik	0	109	0	0	0	1	0	0	0	0	0	1	0	0	0	1	1	0
Chandler	0	0	91	0	0	0	1	1	1	0	0	2	0	0	0	0	0	0
Fernette	1	0	0	92	5	0	0	0	0	7	0	0	0	0	0	0	0	0
Fernor	0	0	0	3	117	0	1	2	0	0	0	0	0	0	0	0	0	0
Hardley	0	0	0	0	0	109	0	0	0	0	0	0	0	2	0	1	0	0
Howard	0	0	0	0	5	0	86	2	1	5	0	0	0	1	0	0	0	0
Kaman1	0	0	0	0	2	0	3	107	0	2	0	0	0	0	0	0	0	1
Kaplan86	0	0	0	0	0	0	0	2	50	0	46	0	1	0	0	0	0	0
Lara	0	0	0	2	1	0	3	1	3	82	1	0	0	2	0	0	0	0
Maya1	0	0	0	0	1	0	0	0	37	0	49	0	1	0	0	0	0	0
Mithland	0	0	0	0	0	0	0	0	0	0	0	168	0	0	0	0	2	0
Oguzlar77	0	0	0	0	0	0	0	0	0	0	0	0	69	0	1	0	0	0
Pedro	0	2	0	0	1	0	3	0	0	0	1	0	1	83	0	0	0	0
Sebin	0	0	0	0	0	0	0	0	0	0	0	0	2	3	97	0	2	0
Sen	0	4	0	0	0	0	0	0	0	0	0	0	1	0	0	170	1	5
Serr	0	3	2	0	0	0	0	0	0	0	0	5	0	1	0	0	127	0
Yalova3	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	115

Şekil 0.8. Arttırılmış veri setinin DarkNet karışıklık matrisi.

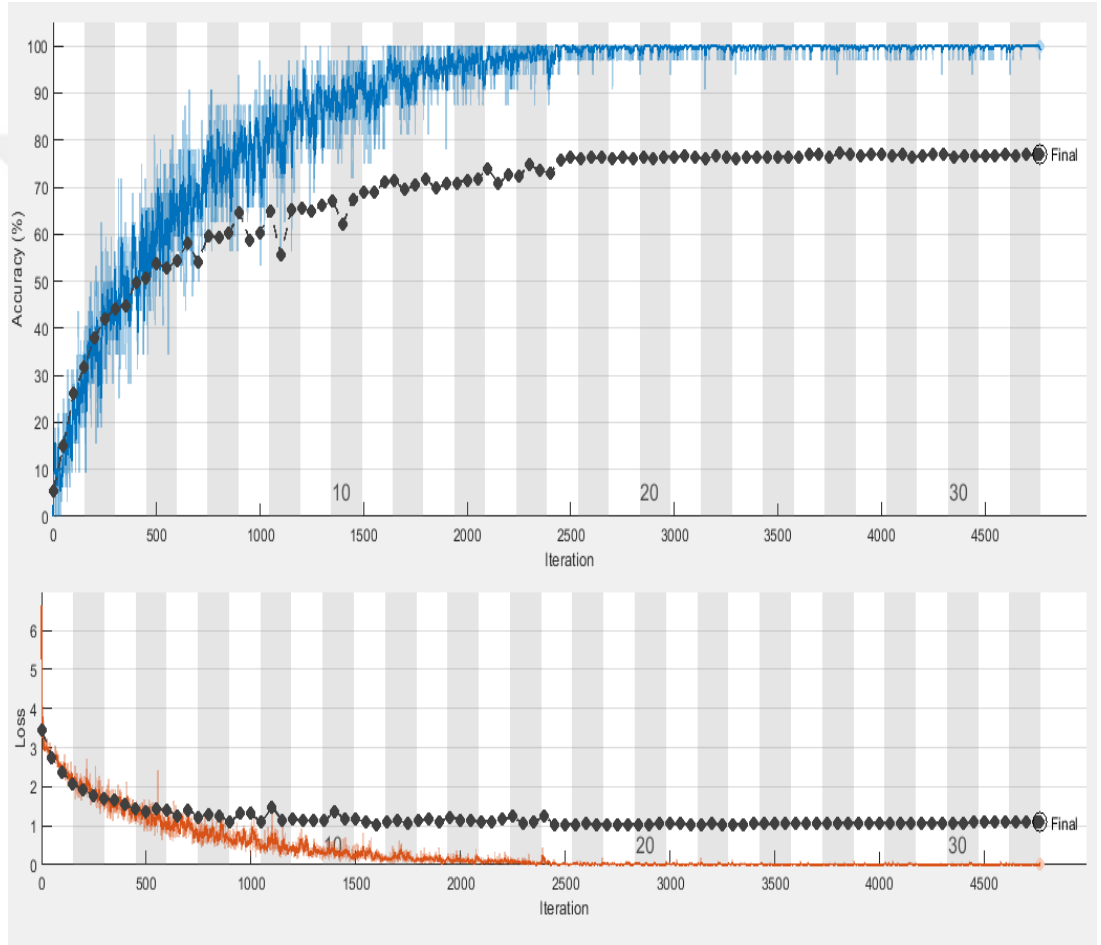
Çizelge 5.2 ve Çizelge 5.3’de görüldüğü üzere arttırılmış veri setinin performansı orijinal veri setine kıyasla tüm ESA modellerinde daha başarılıdır. Ayrıca arttırılmış veri setinde Vgg19’un duyarlılık ve f-Score performans metriklerinde Vgg16’dan daha başarılı sonuçlar vermiştir. Bu durum veri setindeki görüntü sayısının ESA modelleri için önemini de göstermektedir.

5.2. GRADCAM-VGG16 MODELİNİN DENEYSEL TEST SONUÇLARI

Literatürde yaygın olarak kullanılan ESA algoritmaları ile yapılan ESA-9 isimli çalışmadan elde edilen test sonuçlarında, en yüksek doğruluk oranına sahip olan ESA

algoritması Vgg16 olarak bulunmuştur. Bu yüzden önerilen bu modelde Vgg16 ESA modeli temel alınmıştır.

Grad-Cam görüntülerinden oluşan yeni veri seti yine Vgg16 ESA modeli ile sınıflandırılmıştır. Yapılan deneysel testler sonucunda elde edilen doğruluk oranı %77.11 olarak bulunmuştur. Deneysel test sonuçları Şekil 5.9'da ve Çizelge 5.4'de, karışıklık matrisi ise Şekil 5.10'da gösterilmiştir.



Şekil 0.9. GRADCAM-VGG16 modelinin test sonuçları.

	Frenquette	Bilecik	Chandler	Fernette	Fernor	Hardley	Howard	Kaman1	Kaplan86	Lara	Maya1	Mithland	Oguzlar77	Pedro	Sebin	Sen	Serr	Yalova3
Frenquette	135	1	1	2	1	0	1	2	1	0	0	2	0	0	0	0	1	0
Bilecik	4	82	0	1	1	4	0	0	0	0	0	0	2	2	0	6	4	7
Chandler	2	0	65	3	3	2	1	5	1	3	2	1	0	5	0	2	1	0
Fernette	3	0	0	90	2	1	0	2	0	3	0	2	0	0	1	1	0	0
Fernor	2	0	3	9	93	0	2	1	1	5	1	1	0	2	0	0	1	2
Hardley	2	3	4	0	0	95	0	0	0	1	0	0	0	4	0	3	0	0
Howard	0	0	1	5	9	0	68	2	3	3	1	0	1	3	1	1	1	1
Kaman1	2	1	0	1	5	1	3	91	3	0	4	0	0	1	0	0	3	0
Kaplan86	0	0	1	2	3	2	3	6	54	0	25	0	0	2	0	0	1	0
Lara	0	1	5	5	2	0	6	3	0	47	1	0	1	2	0	1	0	1
Maya1	0	0	3	1	0	3	4	1	28	1	44	0	0	0	0	0	3	0
Mithland	1	0	0	0	0	0	0	0	0	0	0	166	0	0	0	1	1	1
Oguzlar77	0	1	0	1	1	0	1	0	0	0	2	0	54	0	8	0	0	2
Pedro	1	3	4	0	4	1	0	2	1	1	0	0	1	69	2	1	1	0
Sebin	0	2	0	0	0	1	0	0	0	1	0	0	6	0	85	4	2	3
Sen	1	12	0	2	0	2	0	0	0	0	0	1	1	1	1	151	6	3
Serr	2	5	3	2	1	2	0	2	0	0	1	2	7	1	2	13	91	4
Yalova3	2	5	0	0	1	1	0	0	0	0	0	1	2	0	1	8	2	103

Şekil 0.10. GRADCAM-VGG16 modeli karışıklık matrisi.

Çizelge 0.4. GRADCAM-VGG16 test sonuçları.

Model	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
Vgg16-GradCam	77.11	71.82	98.62	71.49	76.82

5.3. GLCM-KİKARE-ESA MODELİNİN DENEYSEL TEST SONUÇLARI

GLCM-KİKARE-ESA modelinde ilk deneysel çalışmada MobilnetV2 ve SqueezeNet ile sınıflandırılma yapılmıştır. MobilNetV2 ve SqueezeNet'ten sırasıyla %82.12 ve %84.17 doğruluk oranı elde edilmiştir. Ayrıntılı deneysel test sonuçları Çizelge 5.5'te gösterilmiştir.

Çizelge 0.5. MobilNetV2 ve SqueezeNet’ in deneysel test sonuçları.

ESA Model	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
MobilenetV2	82.12	80.52	98.95	80.83	80.53
SqueezeNet	84.17	83.12	99.07	83.04	82.99

MobilenetV2 ve SqueezeNet ile yapılan eğitimlerde kullanılan hiper-parametreler literatürde yer alan araştırmalar ve deneme yanılma yöntemi ile bulunan en iyi sonuçlara göre ayarlanmıştır. Çalışmadaki hiper-parametreler Çizelge 5.6’da verilmiştir.

Çizelge 0.6. GLCM-KİKARE-ESA kullanılan hiper-parametreler.

ESA Modeli	Görüntü Boyutu	mini-batch Boyutu	Öğrenme Oranı	Epoch
MobilnetV2	224x224	16	0.0001	32
Squeezenet	227X227	16	0.0001	32

Bir sonraki aşamada MobilnetV2 ve SqueezeNet ESA modellerinin sırasıyla Logits ve Pool10 katmanlarından her bir görüntüye ait 1000 özellik çıkarılmıştır. Ayrıca literatürde geleneksel özellik çıkarma yöntemleri arasında yaygın olarak kullanılan GLCM yöntemi ile 64 özellik çıkarılmıştır. Her bir ESA modelinden çıkarılan özellikler ve GLCM’den çıkarılan özellikler ayrı ayrı DVM yöntemi ile sınıflandırılmıştır. Sınıflandırma işlemlerinde veri setinin %70’i eğitim için, %30’u ise test amaçlı kullanılmıştır. Deneysel test sonuçları Çizelge 5.7’de verilmiştir.

Çizelge 0.7. GLCM-KİKARE-ESA çıkarılan özelliklerin deneysel test sonuçları.

ESA Model	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
MobilenetV2	79.63	77.82	98.80	79.09	78.31
SqueezeNet	75.58	73.67	98.56	74.58	73.99
GLCM	51.83	49.63	97.16	50.50	49.86

Sonraki aşamada Squeezenet, MobilnetV2 ve GLCM ile çıkarılan özellikler birleştirilerek 2064 adet özellikten oluşan bir öznitelik havuzu oluşturulmuştur. Bu öznitelik havuzu aynı parametrelerle DVM ile sınıflandırılmıştır. Deneysel test sonuçları Çizelge 5.8’de verilmiştir.

Çizelge 0.8. GLCM-KİKARE-ESA öznitelik havuzu deneysel test sonuçları.

Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
83.67	82.14	99.04	83.36	82.60

Model başarısını arttırabilmek için çıkarılan özelliklerden en iyileri, literatürdeki popüler özellik seçim yöntemlerinden birisi olan Ki-kare testi ile seçilerek ayrı ayrı 258, 516 ve 1032 adet özellik elde edilmiştir. Ki-kare ile seçilerek oluşturulan bu 3 özellik seti DVM ile sınıflandırılmıştır. Kubik DVM ile yapılan bu sınıflandırmalar sonucunda en yüksek doğruluk oranı olarak %84.75değeri elde edilmiştir. Yapılan bu deneysel testlerin sonuçları Çizelge 5.9’ da verilmiştir.

Çizelge 0.9. GLCM-KİKARE-ESA deneysel test sonuçları.

Özellik Sayısı	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
258	83.87	82.13	99.05	82.99	82.45
516	84.75	83.46	99.10	83.93	83.61
1032	81.53	79.70	98.91	81.15	80.24

Deneysel testler sonucunda en iyi doğruluk sonucunu veren 516 özellikten oluşan modelin karmaşıklık matrisi Şekil 5.11’de verilmiştir.

	Frenquette	Bilecik	Chandler	Fernette	Fernor	Hardley	Howard	Kaman1	Kaplan86	Lara	Maya1	Mithland	Oguzlar77	Pedro	Sebin	Sen	Serr	Yalova3
Frenquette	126	4	2	1	0	0	1	0	0	0	0	7	0	0	0	0	0	2
Bilecik	2	103	0	0	0	0	0	0	0	0	0	0	1	0	0	5	1	1
Chandler	0	2	86	1	1	0	1	1	0	0	0	0	0	1	0	2	2	0
Fernette	0	0	0	96	3	0	0	0	1	1	2	0	0	1	0	0	1	0
Fernor	2	0	0	4	111	0	1	0	1	3	1	0	0	0	0	0	0	0
Hardley	1	1	2	0	0	101	0	1	0	0	0	0	0	4	0	0	1	0
Howard	0	0	0	3	3	0	86	5	0	1	0	0	0	2	0	0	0	0
Kaman1	0	0	0	3	3	0	4	93	4	6	0	0	0	0	0	2	0	0
Kaplan86	0	0	1	3	1	1	1	5	48	2	3	0	0	0	0	0	1	0
Lara	0	0	2	3	7	0	0	3	0	56	4	0	0	1	0	0	0	0
Maya1	1	0	0	0	0	0	1	0	37	1	43	0	0	3	0	0	1	0
Mithland	2	0	0	0	0	0	0	0	0	0	0	164	0	0	0	0	5	0
Oguzlar77	0	1	0	0	0	4	0	0	0	0	0	0	62	0	2	0	0	0
Pedro	0	3	3	0	2	0	0	0	3	0	0	0	0	76	0	0	1	1
Sebin	0	0	0	0	1	0	0	0	0	0	0	0	1	2	97	1	0	3
Sen	0	5	0	0	0	0	0	0	0	0	0	6	0	0	0	165	0	4
Serr	0	2	3	1	2	1	0	0	0	0	3	4	1	0	1	2	118	0
Yalova3	1	7	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	108

Şekil 0.11. GLCM-KİKARE-ESA karmaşıklık matrisi.

5.4. RES_AAO MODELİNİN DENEYSEL TEST SONUÇLARI

Deneysel çalışmanın ilk aşamasında Ceviz yaprak görüntüleri Resnet18, Resnet50 ve ResNet101 ESA modelleri ile eğitilmiştir. ESA modellerini eğitirken, sonuçların daha etkili olabilmesi için hiper-parametreler ayarlanmıştır. Deneysel çalışmaları test etmek için Intel Core i7 işlemci 8 GB RAM ve 4 GB Nvidia grafik işlemci kartı kullanılmıştır. Kodlar Matlab2021b yazılım ortamında derlenmiştir. Her iterasyonda eğitilen veri sayısı olan mini-batch boyutu 16, tüm veri setinin bir sefer eğitilmiş olduğu maksimum epoch değeri 32 ve öğrenme oranı 1×10^{-4} olarak belirlenmiştir. Bu hiper-parametreler veri setinde bulunan veri sayısı ve makinanın donanım özellikleri göz önünde bulundurulmuştur. Deneysel çalışma sonucunda ResNet101 modeli ile en iyi performans değerleri elde edilmiştir. ResNet101 modeli ile %81.77 doğruluk oranı

elde edilirken ResNet18 ve ResNet50 modelleri ile sırasıyla %74.9 ve %78.94 doğruluk oranı elde edilmiştir. ESA modelleri ile veri setinin eğitilmesi sonucu elde edilen performans sonuçları Çizelge 0.10’da verilmiştir.

Çizelge 0.10. ResNet modellerinin performans sonuçları.

ESA Model	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
Resnet18	74.90	73.34	98.52	74.29	73.59
Resnet50	78.94	77.55	98.76	0.78.44	77.90
Resnet101	81.77	80.29	98.93	0.80.93	77.90

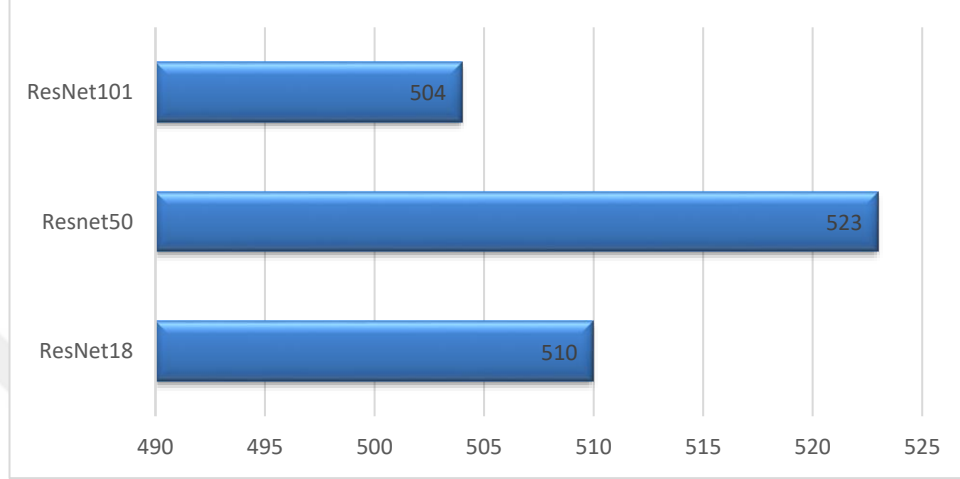
Deneysel çalışmanın ikinci aşamasında ESA modelleri ile veri seti sınıflandırıldıktan sonra her bir ResNet modelinin fc1000 tam bağlı katmanından 1000 adet özellik çıkarılmıştır. Bu özellik setleri Linear DVM ve DVM’nin farklı çekirdekleri ile sınıflandırılarak performans sonuçlarına bakılmıştır. Sınıflandırma sonucunda en iyi performans sonuçları ResNet50 ESA modelinin Cubic çekirdeği ile elde edilmiştir. Cubic çekirdeği ile %83.24 doğruluk oranı elde edilirken duyarlılık, özgünlük, kesinlik ve f-skor değerleri sırasıyla %82.07, %99.01, %82.85 ve %82.24 olarak elde edilmiştir. Fc1000 katmanında görüntülere ait 1000 özelliğin Linear DVM ve DVM’nin kernelleri ile sınıflandırılması sonucu elde edilen performans sonuçları Çizelge 0.11’ de verilmiştir.

Çizelge 0.11. ResNet metotlarının DVM ile birlikte sınıflandırma performansı.

ESA Model	DVM Kernel	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
Resnet18	Linear	64.50	63.36	97.90	64.37	63.66
	Quadratic	74.05	72.81	98.47	73.81	73.13
	Cubic	74.41	73.07	98.49	74.19	73.37
Resnet50	Linear	74.51	73.34	98.50	75.42	73.92
	Quadratic	82.99	81.86	99.00	82.95	82.19
	Cubic	83.24	82.07	99.01	82.85	82.24
Resnet101	Linear	73.62	72.59	98.44	73.93	73.01
	Quadratic	81.02	79.93	98.88	80.93	80.24
	Cubic	81.61	80.68	98.92	81.29	80.81

Deneysel çalışmanın üçüncü aşamasında, Fc1000 katmanında çıkarılan özelliklerden sınıflandırma amacına yönelik etkili olan özelliklerin belirlenmesi için AAO

algoritması kullanılmıştır. AAO algoritması ile ResNet18 modelinden 510 özellik seçilirken, ResNet50 ve ResNet101 modellerinden sırasıyla 523 ve 504 adet özellik seçilmiştir. AAO algoritması ile ResNet modellerinden seçilen özellikler Şekil 5.12’de gösterilmiştir.



Şekil 0.12. Resnet modellerinden AAO ile seçilen özellik sayısı.

Daha sonra seçilen özellikler sırasıyla DVM ile sınıflandırılmıştır. Deneysel çalışma sonucunda ResNet50 ESA modelinden elde edilen 523 özellik DVM’nin quadratic çekirdeği kullanılmasıyla %83.04 doğruluk oranı ile sınıflandırarak en iyi performans sonuçları elde edilmiştir. AAO ile seçilen özelliklerin DVM ile sınıflandırılması sonucu elde edilen performans değerleri Çizelge 5.12’ de verilmiştir.

Çizelge 0.12. AAO kullanarak sınıflandırmanın performans sonuçları.

ESA Model	DVM Kernel	Selected Feature	Doğruluk	Duyarlılık	Özgünlük	Kesinlik	F-skor
Resnet18	Linear	510	63.00	61.54	97.82	63.32	62.07
	Quadratic		73.25	71.81	98.42	73.47	72.36
	Cubic		75.24	73.67	98.54	75.69	74.33
Resnet50	Linear	523	75.04	73.91	98.53	74.95	74.09
	Quadratic		83.04	81.90	99.00	82.68	82.13
	Cubic		82.76	81.55	98.99	82.32	81.77
Resnet101	Linear	504	72.24	70.92	98.36	73.07	71.66
	Quadratic		81.27	79.86	98.90	81.34	80.35
	Cubic		80.72	79.33	98.87	80.73	79.73

Deneysel çalışmanın dördüncü aşamasında, AAO algoritması ile ResNet18 modelinden elde edilen 510 öz nitelik, ResNet50 modelinden elde edilen 523 öz nitelik

ve ResNet101 modelinden elde edilen 504 öznitelik birleştirilerek her bir görüntüye ait toplamda 1537 öznitelikten oluşan yeni bir özellik seti oluşturulmuş ve DVM ile sınıflandırılmıştır. Sonuç olarak en iyi performans sonuçları DVM'nin Cubic çekirdeği ile %86.12 doğruluk oranı elde edilmiştir. Deneysel çalışmaların performans sonuçları Çizelge 0.13'te verilmiştir.

Çizelge 0.13. Tüm ESA özelliklerine AAO uygulanarak oluşturulan niteliklerin sınıflandırılması.

DVM Kernel	Acc (%)	Se (%)	Sp (%)	Prec (%)	F-Sc (%)
Linear	80.73	79.44	98.87	80.64	79.84
Quadratic	80.95	78.99	98.88	76.51	80.12
Cubic	86.12	84.91	99.18	85.72	85.24

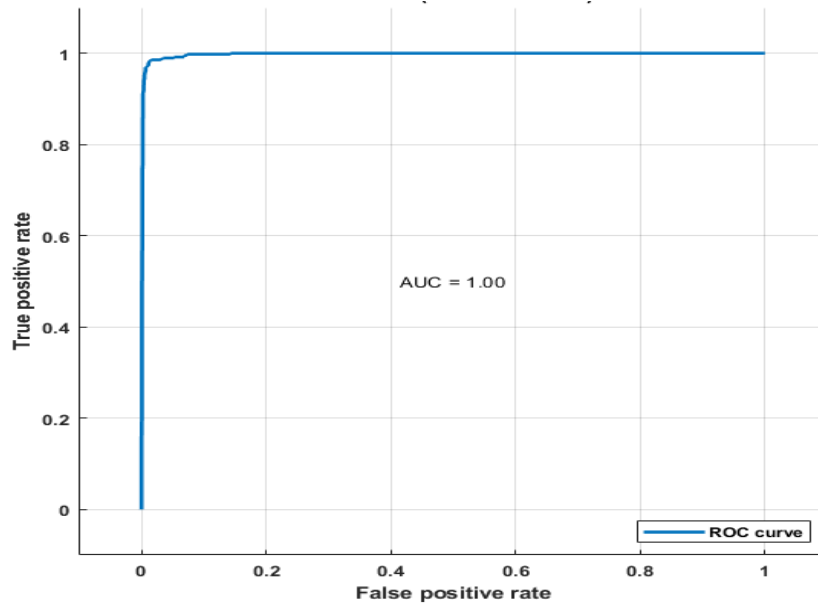
Son aşamada önerilen modelin güvenilirliği için 5-fold çapraz doğrulama (cross validation) testi uygulanmıştır. Çapraz doğrulama testi yapıldıktan sonra önerilen modelin doğruluk oranı %87.42 doğruluk oranı ile DVM'nin quadratic çekirdeği ile elde edilmiştir. Deneysel çalışmalar sonucunda DVM'nin quadratic çekirdeği kullanılarak elde edilen karmaşıklık matrisi (confusion matrixi) ve ROC eğrisi sırasıyla Şekil 5.13 ve Şekil 5.14' de verilmiştir. Çapraz doğrulama testi ile elde edilen performans sonuçları Çizelge 0.14'de verilmiştir. Ayrıca AAO'nun yakınsama grafiği Şekil 5.15'de gösterilmektedir.

Çizelge 0.14. RES_AAO modelinin deneysel test sonuçları.

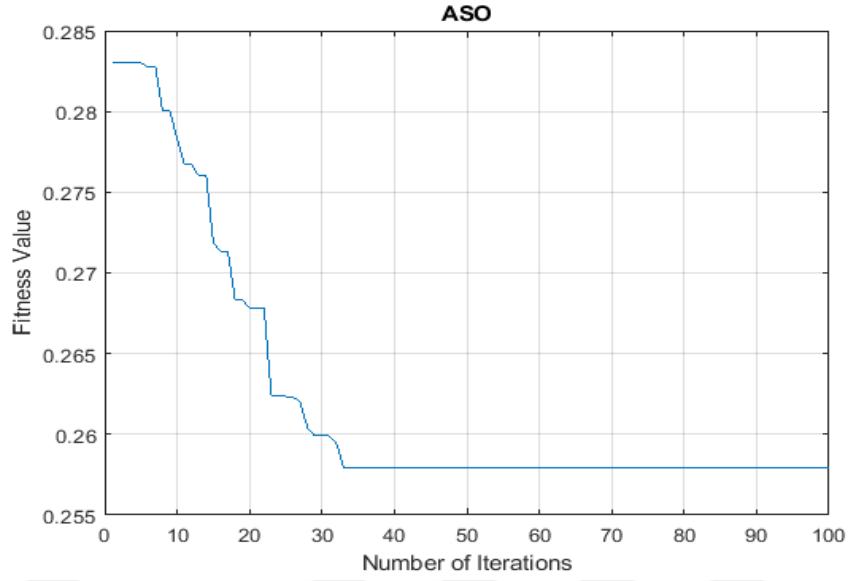
DVM Kernel	Acc (%)	Se (%)	Sp (%)	Prec (%)	F-Sc (%)
Linear	80.46	79.25	98.85	80.10	79.56
Quadratic	87.42	86.27	99.26	86.80	86.49
Cubic	86.95	81.60	99.23	82.09	81.79

	Frenquette	Bilecik	Chandler	Fernette	Fernor	Hardley	Howard	Kaman1	Kaplan86	Lara	Maya1	Mithland	Oguzlar77	Pedro	Sebin	Sen	Serr	Yalova3
Frenquette	457	0	4	2	0	0	0	2	1	0	0	15	0	0	0	3	4	0
Bilecik	2	355	1	0	0	0	0	0	0	0	0	0	0	4	0	6	3	7
Chandler	1	0	307	2	2	1	0	3	0	3	2	0	0	0	0	1	1	1
Fernette	4	1	1	324	6	1	0	1	2	6	0	1	0	2	0	0	2	0
Fernor	1	0	3	8	366	0	7	8	1	7	1	3	0	0	0	0	0	1
Hardley	0	6	10	1	1	342	0	0	1	0	0	1	0	7	2	2	1	1
Howard	0	1	1	0	12	0	277	19	4	13	3	0	0	1	0	0	1	0
Kaman1	0	0	2	6	7	1	7	350	7	5	1	0	0	0	0	1	1	1
Kaplan86	0	0	2	1	0	0	1	3	173	2	147	0	0	0	0	0	1	0
Lara	0	0	4	9	8	0	5	5	5	202	8	0	0	0	0	0	0	3
Maya1	0	0	2	0	1	0	0	1	169	0	117	0	0	0	0	0	0	1
Mithland	10	0	2	1	1	0	0	0	0	0	0	393	0	0	0	2	9	0
Oguzlar77	0	1	0	0	0	1	0	0	0	1	1	1	211	0	3	7	1	5
Pedro	0	2	3	3	2	0	1	0	0	1	6	1	0	280	1	2	0	1
Sebin	0	1	0	0	0	4	0	0	0	0	0	0	6	2	323	4	3	1
Sen	1	8	0	0	0	0	0	0	0	0	0	5	0	0	0	505	0	9
Serr	2	10	0	0	1	1	0	0	0	0	1	24	0	1	2	2	394	3
Yalova3	0	7	0	0	0	0	0	0	0	0	0	0	1	0	0	14	6	399

Şekil 0.13. RES_AAO modelinin karışıklık matrisi.



Şekil 0.14. RES_AAO modelinin ROC eğrisi.



Şekil 0.15. AAO'nun yakınsama grafiği.

5.5. BOA-ESA-KNN MODELİNİN DENEYSEL TEST SONUÇLARI

Deneysel çalışmaların ön işleme ve veri büyütme aşamaları yapılan ilk çalışmada anlatılmıştır. BOA-ESA-KNN modelinde öncelikle ESA-9 modelinde en yüksek doğruluk oranına sahip olan üç ESA tespit edilmiştir. Önceden eğitilen bu ESA modelleri Vgg16, Vgg19 ve AlexNet'ten 1000'er özellik çıkarılmıştır. Çıkarılan özellikler birleştirilerek yeni bir öznetelik havuzu oluşturulduktan sonra KNN ile sınıflandırılarak performans başarısı ölçülmüştür. Daha sonra ESA'lardan çıkarılan özelliklerden tahmin edilen en iyileri BOA ile ayrı ayrı seçilerek birleştirilmiştir. Son olarak seçilen özellikler KNN ile sınıflandırılmıştır. KNN ile sınıflandırma yapılırken ise mesafe tespiti için minkowski mesafesi kullanılmıştır. KNN ile yapılan tüm deneysel çalışmalarda K değeri 3 olarak seçilmiştir. Deneysel testler sonucunda başarı ölçütleri Çizelge 5.15'de gösterilmiştir.

Çizelge 0.15. BOA-ESA-KNN modeli KNN deneysel test sonuçları.

Model	Acc (%)	Se (%)	Sp (%)	Prec (%)	F-Sc (%)
KNN (3000)	80.36	79.23	98.84	80.53	79.68
KNN-BOA(1734)	92.59	92.06	99.56	92.17	92.06

Çizelgede görüldüğü gibi, KNN ile sınıflandırma yapıldığında önerilen model %92.59 doğruluk oranı ile oldukça başarılı sonuç vermiştir. KNN ile yapılan sınıflandırmanın karmaşıklık matrisi Şekil 5.16’ da gösterilmektedir.

	Frenquette	Bilecik	Chandler	Fernette	Fernor	Hardley	Howard	Kaman1	Kaplan86	Lara	Maya1	Mithland	Oguzlar77	Pedro	Sebin	Sen	Serr	Yalova3
Frenquette	476	2	0	1	1	0	0	0	0	0	0	8	0	0	0	1	0	0
Bilecik	4	349	1	1	0	6	1	0	0	0	0	1	5	1	0	5	1	1
Chandler	4	0	309	0	0	2	0	2	0	0	0	2	1	0	0	1	0	0
Fernette	5	0	3	335	1	0	0	1	0	0	0	1	0	2	0	2	1	0
Fernor	4	0	4	3	391	0	1	0	0	3	0	0	1	0	0	0	0	2
Hardley	1	1	0	0	1	366	1	0	0	0	0	0	0	0	0	0	0	2
Howard	0	0	4	0	10	6	306	1	0	0	4	0	0	1	0	0	0	0
Kaman1	2	0	5	3	0	1	4	368	0	0	1	0	0	0	0	0	0	0
Kaplan86	0	0	1	1	5	0	0	0	255	0	69	0	0	0	0	0	0	0
Lara	0	0	4	6	6	0	1	3	1	226	1	0	0	0	0	0	1	2
Maya1	0	0	1	2	5	2	2	2	83	0	196	0	0	0	0	0	0	0
Mithland	9	1	1	0	2	0	0	0	0	1	2	541	0	0	0	6	3	1
Oguzlar77	0	0	0	0	0	1	0	0	0	0	1	0	231	0	0	0	0	0
Pedro	0	1	2	1	5	11	3	0	1	0	0	0	1	272	0	3	1	2
Sebin	0	1	0	0	1	4	1	0	0	0	1	0	1	2	335	0	1	1
Sen	3	2	0	1	1	4	1	0	0	2	0	4	4	0	1	570	0	10
Serr	5	2	1	1	0	3	1	1	3	0	3	5	1	2	7	1	423	2
Yalova3	0	2	2	0	0	6	2	1	0	1	2	2	3	1	2	7	1	367

Şekil 0.16. BOA-ESA-KNN modeli karmaşıklık matrisi.

KNN doğruluk başarısının tespiti için KNN sınıflandırma yöntemlerinden Chebychev, Cityblock ve Oklit mesafeleri ile de deneysel testler yapılmıştır. Tüm KNN metotlarının deneysel test sonuçları Çizelge 5.16’da verilmiştir.

Çizelge 0.16. BOA-ESA-KNN modeli deneysel test sonuçları.

Model	Acc (%)	Se (%)	Sp (%)	Prec (%)	F-Sc (%)
Minkowski	92.59	92.06	99.56	92.17	92.06
Chebychev	89.20	88.54	99.36	89.18	88.65
Cityblock	92.37	91.84	99.55	91.99	91.86
Oklit	92.48	92.02	99.56	92.03	91.96

Yapılan deneysel testler sonucunda 4 farklı KNN metodu içerisinde en iyi sonuç, Minkowski uzaklık mesafesi kullanılarak %92.59 başarı oranına sahiptir.



BÖLÜM 6

TARTIŞMA

Ceviz çeşitlerinin erken aşamada ayırt edilebilmesi üretici açısından oldukça önemlidir. Yanlış çeşidin yanlış tarım arazisinde kullanılması üreticinin hem maddi açıdan zarar görmesine hem de yıllarca beklemesine neden olmaktadır. Ceviz yaprak görüntülerinden çeşit tanıma işlemini yapabilmek için, her yaprak ayrıntılı olarak incelenmelidir. Ceviz yaprakları birbirine çok benzediği için, gözle veya geleneksel yöntemlerle farkları ayırt etmek oldukça zordur. Ceviz yapraklarından ceviz çeşit tespiti yapılabilmesi için otomatik bir sistem geliştirilmesi hem fidancılara hem de üreticilere yararlı olacaktır.

Bu çalışmada, 18 farklı çeşit ceviz ağacı yaprağı ile oluşturulan özgün veri setinin sınıflandırılması yapılmıştır. Sınıflandırma işlemi için Residual Blok tabanlı bir model önerilmiştir. İlk olarak model başarısının ölçülebilmesi için ResNet18, ResNet50 ve ResNet101 modelleri ile eğitim gerçekleştirilmiştir. Eğitim sonucunda başarı oranları sırasıyla %74.90, %78.90 ve %81.77 olarak bulunmuştur.

Önerilen model de ilk olarak yaprak görüntülerinden özellik çıkarma işlemi yapılmıştır. ESA modelleri bir yaprakta morfolojik olan tüm özellikleri tek seferde ve bütün olarak çıkarırken, geleneksel yöntemler tek tek ve manuel olarak çıkarmaktadır. Bu yüzden ESA modelleri özellik çıkarma işleminde daha hızlı ve daha başarılı sonuçlar vermektedir [123]. Ayrıca Residual Blok tabanlı ESA modellerinden olan ResNet modellerinde, eğitim süreleri kısadır ve bir özellik bir kez öğrenildiğinde, başka özelliklere odaklanmaktadır [66]. Bu yüzden önerilen modelde özellik çıkarma işlemleri için Artık Blok tabanlı ResNet18, ResNet50 ve ResNet101 modelleri kullanılmıştır.

Özellik seçimi, hesaplama yükünü azaltmaya yardımcı olurken, aynı zamanda özellik seti içindeki önemli olan özellikleri alarak sınıflandırma başarısını arttırmaktadır. Ayrıca sınıflandırma performansı için özellik seçim algoritmasının kararlılığı önemlidir [124]. AAO az parametre içeren, basit, bir çok problemin global seçim aşamasında başarılı sonuçlar veren, popülasyon tabanlı sezgisel bir algoritmadır [125]. AAO birçok algoritmayla kıyaslanarak, çoğu problemi son derece başarılı bir şekilde çözen ve üstünlüğünü kabul ettirmiş bir algoritmadır [98]. Bu yüzden, önerdiğimiz modelde çıkarılan öznelilik setine, AAO algoritması uygulanarak mevcut en iyi özellikler seçilmiş ve yeni bir öznelilik seti oluşturulmuştur. Önerilen modelde 5-fold cross validation yöntemi uygulanarak DVM ile sınıflandırma yapılmış ve %87.42 doğruluk oranına ulaşılmıştır. Önerilen modelde yapılan özellik çıkarma, özellik seçimi, sınıflandırma işlemlerinin ResNet ESA modellerini doğrudan kullanmaktan daha başarılı olduğu görülmektedir. Dolayısıyla kullanılan özellik seçim yönteminin daha az özellik sayısına rağmen model başarısını arttırdığından dolayı doğru bir seçim yapıldığı ispatlanmıştır.

Bu tez kapsamında oluşturulan veri seti ile yapılan deneysel çalışmalar Çizelge 6.2’de verilmiştir.

Çizelge 0.1. Veri seti ile yapılan deneysel çalışmalar.

Model	Özellik Çıkarma	Doğruluk
ESA-9	Transfer Öğrenme	%80.61- %90.55
GRADCAM-VGG16	GradCam-Vgg16	%77.11
GLCM-KİKARE-ESA	SequeezeNet, MobilenetV2, GLCM Ki-Kare DVM	%84.75
RES_AAO	ResNet18, ResNet50, ResNet101, AAO, DVM	%87.42
BOA-ESA-KNN	Vgg16, Vgg19, AlexNet, BOA, KNN	%92.59

Yapılan ilk deneysel çalışmada dokuz farklı ESA modeli ile veri seti eğitilmiştir. Ardından veri arttırma teknikleri kullanılarak ESA modelleri ile tekrar eğitilmiştir. Deneysel çalışma sonucunda tüm modellerde modelin performans metriklerinde iyileşme görülmüştür. Bu durum veri arttırma tekniğinin kullanılması sonucunda modelin performans sonucunun iyileşmesi literatürü doğrular niteliktedir [126]. İkinci

deneysel çalışmada Grad-CAM tekniği ile ceviz yaprağı görüntüleri Vgg16 ESA modelinin son konvolüsyon katmanı ile eğitilmiştir. Ardından bu model ile görüntülerin ayırt edilmesi için renklendirilmiştir. Burada modelin ceviz yapraklarının sınıflarını ayırt etmesi için en ayırt edici bölgeler kırmızı ile renklendirilmiştir. Grad-CAM tekniği ile oluşturulan görüntüler VGG16 ile eğitildiğinde %77.11 doğruluk oranı elde edilmiştir. Bu teknik kullanılmadan ise VGG16 modeli ile %90.55 doğruluk oranı elde edilmiştir. Dolayısıyla Grad-CAM tekniği ile sadece son katman ile eğitilerek önemli bölgelerin renklendirilmesi ile modelde istenen düzeyde iyileştirme sağlanamamıştır. Dolayısıyla doğrudan VGG16 modelinin kullanılması ceviz yapraklarını ayırt etmede daha başarılı olduğu söylenebilir. Üçüncü deneysel çalışmada hafif ağırlıklı ESA modelleri ile görüntüler eğitilmiş ve önerilen model ile performans sonuçları kıyaslanmıştır. Önerilen modelde hafif ağırlıklı ESA modellerinden görüntülere ait özellikler çıkarıldıktan sonra GLCM algoritması ile özellik seti zenginleştirilmiştir. Özellik setinin geleneksel algoritmaları ile güçlendirilmesi sonucu model performansında bir iyileşme sağladığı söylenebilir. Literatürde yapılan çalışmalara bakıldığı zaman benzer şekilde ESA modelleri ile elde edilen özelliklerin geleneksel özellik çıkarma algoritmaları ile çıkarılan özellikler ile desteklendiğinde iyileşme sağladığı görülmüştür [127]. Dördüncü deneysel çalışmada artık blok ESA modellerinden elde edilen özelliklerin AAO algoritması ile elde edilen özelliklerin seçilmesi ile ESA modellerinin performanslarını iyileştirildiği ve daha özellikle ile daha iyi sonuçlar elde ettiği belirlenmiştir. Dolayısıyla AAO algoritması ile özellik seçiminin doğru bir tercih olduğu söylenebilir. Son deneysel çalışmada VGG tabanlı modeller tercih edilmiştir. Bu modelleri ile özellik seçiminin BOA algoritması ile yapıldıktan sonra modelde iyileşme sağladığı ve yapılan diğer deneysel çalışma sonuçları ile karşılaştırıldığından en iyi performans sonucunun elde edildiği belirlenmiştir. Çizelge 6.1'ye bakıldığı zaman önerilen modelin son deneysel çalışma sonucu elde edilen performans sonuçları ile tutarlı bir model önerisi olduğu söylenebilir.

BÖLÜM 7

SONUÇLAR VE ÖNERİLER

Ceviz yetiştiriciliğine, dikilecek olan ceviz çeşitlerinin fide döneminde yapraklarından ayırt edilmesi son derecede önemlidir. Bu durum üreticiye zaman ve maliyet bakımından önemli avantaj sağlamaktadır. Çünkü cevizin ürün verme süresi uzundur ve yanlış çeşit ceviz ile kurulan bahçelerdeki zararlar oldukça fazla olmaktadır. Bu çalışmada ilk olarak 18 farklı ceviz çeşidinden 1751 adet ceviz yaprak görüntüleri alınarak özgün bir ceviz veri seti literatüre kazandırılmıştır. Veri setinde yer alan yapraklar, Yalova Atatürk Bahçe Kültürleri Merkez Araştırma Enstitüsü Uygulama Bahçesinden gerekli izinler alınarak ve uzmanlar yardımıyla elde edilmiştir. Özgün ceviz veri setinin sınıflandırılması için farklı derin öğrenme modelleri önerilerek, yüksek doğruluk başarısı elde edilmiştir.

Özgün veri setine ilk önce ön işleme ve veri arttırma yöntemleri uygulanmıştır. Ön işleme ile yaprak görüntüleri üzerindeki çekimden kaynaklı hatalar giderilmiştir. Daha sonra, veri setine döndürme (rotation), parlaklık (brightness), kırpma (shear), yakınlaştırma (zoom) ve çevirme (flip) işlemleri uygulanarak görüntü sayısı yaklaşık 4 kat arttırılmıştır. Yapılan tüm deneysel çalışmalar hem özgün veri seti hemde arttırılmış veri seti üzerinde yapılarak test edilmiştir. Veri arttırma işlemi sonucunda yapılan deneysel testlerin performans sonuçlarında iyileştirme sağladığı belirlenmiştir. İlk olarak veri seti literatürde yaygın olarak kullanılan 9 ESA algoritması ile sınıflandırılmış. Bu deneysel testler ve doğruluk oranları, AlexNet %77.71, Dark.Net19 %76.95, GoogleNet %75.24, GoogleNet %64.39, InceptionV3 %76.95, ResNet50 %75.24, ResNet101 %85.52, Vgg16 %84.00, Vgg19 %74.67 olarak bulunmuştur.

Önerilen ilk modelde artırılmış veri seti görüntülerine Grad-Cam yöntemi uygulanarak yeni bir veri seti üretilmiştir. Üretilen Grad-Cam görüntüleri yine Vgg16 ile sınıflandırılmış ve %77.11 oranında doğruluk sonucu elde edilmiştir. Ulaşılan bu doğruluk sonucu rekabetçi bir seviyede olsa da istenilen düzeyde değildir.

Önerilen bir sonraki modelde artırılmış veri seti ilk olarak MobilnetV2 ve SqueezeNet ile eğitilmiştir. Eğitim sonrasında MobilNetV2, SqueezeNet ve GLCM ile özellikler çıkarılarak bir öznitelik havuzu oluşturulmuştur. Oluşturulan yeni özellik setindeki en iyi özellikler Ki-kare özellik seçim algoritması ile belirlenmiştir. Bu öznitelik havuzu DVM ile sınıflandırılarak yapılan deneysel testler sonucunda %84.75 doğruluk oranı elde edilmiştir.

Bir sonraki model önerisi için ResNet modellerinden yararlanılmıştır. Veri seti Artık Blok tabanlı ResNet modellerinden ResNet18, ResNet50 ve ResNet101 ile eğitilmiştir. Ham görüntülerin doğrudan ESA modelleri ile eğitilmesi sonucu en iyi sonuç ResNet101 modeli ile %81,77 doğruluk oranı elde edilmiştir. Eğitim sonrasında her bir ResNet modelinden ayrı ayrı özellik çıkarma işlemi yapılmıştır. Çıkarılan özelliklere AAO optimizasyon algoritması uygulanarak daha verimli yeni bir özellik seti oluşturulmuştur. Son olarak özellik setine 5-parçalı çapraz doğrulama uygulandıktan sonra DVM ile sınıflandırma yapılarak önerilen model test edilmiştir. Çapraz doğrulama sonucunda %87,42 doğruluk oranı elde edilmiştir. Doğrudan ESA modellerine (ResNet18, ResNet50, ResNet101) göre önerilen model sırasıyla %12.52, %8.48 ve %5.65 doğruluk oranında iyileştirme yapılmıştır. Deneysel çalışmalar sonucunda görüntülere ait derin özelliklerin kullanımı ve AAO yöntemi ile özellik seçiminin etkili olduğu ispatlanmıştır.

Önerilen son modelde, yapılan ilk çalışmada en yüksek başarı oranının elde edildiği Vgg16, Vgg19 ve AlexNet ESA algoritmaları temel alınmıştır. Bu algoritmalarla özellikler çıkarılarak yeni bir özellik seti oluşturulmuştur. Oluşturulan bu yeni özellik setinden en iyi özellikler BOA optimizasyon algoritması ile seçilmiştir. Seçilen özellikler KNN ile test edilerek en iyi sonuç KNN algoritmasında K'nın 3 seçildiği durumda bulunmuştur. Önerilen son modelde ise %92,59 doğruluk oranı ile en yüksek başarı oranı elde edilmiştir.

Gelecekteki çalışmalarda literatürde yer alan farklı sinir ağı modelleri ile başarı oranı arttırılmaya çalışılacaktır. Ayrıca oluşturulan özgün ceviz veri seti üzerinde ki görüntüler üzerinde farklı görüntü işleme teknikleri uygulanarak model başarıları iyileştirilmeye çalışılacaktır. Bunlara ek olarak, fidancıların ve üreticilerin kullanımına sunulmak üzere, en yüksek sonucu veren model için bir mobil uygulama yapılması somut bir çözüm olabilecektir.



KAYNAKLAR

- [1] K. Kulkarni, Z. Zhang, L. Chang, J. Yang, P. C. A. da Fonseca, and D. Barford, "Building a pseudo-atomic model of the anaphase-promoting complex," *Acta Crystallogr. Sect. D Biol. Crystallogr.*, vol. 69, no. 11, pp. 2236–2243, 2013.
- [2] T. Karadeniz and E. Güler, "CUMHURİYETİN İLK YILLARINDAN GÜNÜMÜZE CEVİZ YETİŞTİRİCİLİĞİ," *Bahçe*, 2017.
- [3] T. KARADENİZ, "Ordu Yöresinde yetiştirilen ceviz genotiplerinin (*Juglans regia* L.) seleksiyonu," *Ordu Üniversitesi Bilim ve Teknol. Derg.*, vol. 1, no. 1, pp. 65–74, 2011.
- [4] F. DOĞAN and İ. TÜRKOĞLU, "Derin öğrenme algoritmalarının yaprak sınıflandırma başarımlarının karşılaştırılması," *Sak. Univ. J. Comput. Inf. Sci.*, vol. 1, no. 1, pp. 10–21, 2018.
- [5] S. Solak and U. ALTINIŞIK, "Görüntü işleme teknikleri ve kümeleme yöntemleri kullanılarak fındık meyvesinin tespit ve sınıflandırılması," *Sak. Univ. J. Sci.*, vol. 22, no. 1, pp. 56–65, 2018.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [7] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant disease detection and classification by deep learning," *Plants*, vol. 8, no. 11, p. 468, 2019.
- [8] T. KARAHAN and V. Nabiyeve, "Plant identification with convolutional neural networks and transfer learning," *Pamukkale Üniversitesi Mühendislik Bilim. Derg.*, vol. 27, no. 5, pp. 638–645, 2021.
- [9] I. M. Dheir, A. Soliman, A. Mettleq, and A. A. Elsharif, "Nuts Types Classification Using Deep learning," *Int. J. Acad. Inf. Syst. Res.*, vol. 3, no. 12, pp. 12–17, 2019.
- [10] D. K. Nkemelu, D. Omeiza, and N. Lubalo, "Deep convolutional neural network for plant seedlings classification," *arXiv Prepr. arXiv1811.08404*, 2018.
- [11] Y. Sun, Y. Liu, G. Wang, and H. Zhang, "Deep learning for plant identification in natural environment," *Comput. Intell. Neurosci.*, vol. 2017, 2017.
- [12] J. Champ, T. Lorieul, M. Servajean, and A. Joly, "A comparative study of fine-grained classification methods in the context of the LifeCLEF plant identification challenge 2015," in *CLEF: Conference and Labs of the Evaluation forum*, 2015, no. 1391.

- [13] H. Zhu, Q. Liu, Y. Qi, X. Huang, F. Jiang, and S. Zhang, "Plant identification based on very deep convolutional neural networks," *Multimed. Tools Appl.*, vol. 77, no. 22, pp. 29779–29797, 2018.
- [14] M. Koklu, M. F. Unlarsen, I. A. Ozkan, M. F. Aslan, and K. Sabanci, "A CNN-SVM study based on selected deep features for grapevine leaves classification," *Measurement*, vol. 188, p. 110425, 2022.
- [15] A. Beikmohammadi, K. Faez, and A. Motallebi, "SWP-LeafNET: A novel multistage approach for plant leaf identification based on deep CNN," *Expert Syst. Appl.*, vol. 202, p. 117470, 2022.
- [16] J. Chaki, R. Parekh, and S. Bhattacharya, "Plant leaf classification using multiple descriptors: A hierarchical approach," *J. King Saud Univ. Inf. Sci.*, vol. 32, no. 10, pp. 1158–1172, 2020.
- [17] S. H. Lee, C. S. Chan, P. Wilkin, and P. Remagnino, "Deep-plant: Plant identification with convolutional neural networks," in *2015 IEEE international conference on image processing (ICIP)*, 2015, pp. 452–456.
- [18] X. Zhu, M. Zhu, and H. Ren, "Method of plant leaf recognition based on improved deep convolutional neural network," *Cogn. Syst. Res.*, vol. 52, pp. 223–233, 2018.
- [19] M. A. Alkahlout, S. S. Abu-Naser, A. H. Alsaqqa, and T. N. Abu-Jamie, "Classification of Fruits Using Deep Learning," 2022.
- [20] D. Bisen, "Deep convolutional neural network based plant species recognition through features of leaf," *Multimed. Tools Appl.*, vol. 80, no. 4, pp. 6443–6456, 2021.
- [21] A. Taslim, S. Saon, M. Muladi, and W. N. Hidayat, "Plant leaf identification system using convolutional neural network," *Bull. Electr. Eng. Informatics*, vol. 10, no. 6, pp. 3341–3352, 2021.
- [22] Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, "Plant leaf disease classification using EfficientNet deep learning model," *Ecol. Inform.*, vol. 61, p. 101182, 2021.
- [23] S. M. Hassan, A. K. Maji, M. Jasiński, Z. Leonowicz, and E. Jasińska, "Identification of plant-leaf diseases using CNN and transfer-learning approach," *Electronics*, vol. 10, no. 12, p. 1388, 2021.
- [24] R. C. Joshi, M. Kaushik, M. K. Dutta, A. Srivastava, and N. Choudhary, "VirLeafNet: Automatic analysis and viral disease diagnosis using deep-learning in Vigna mungo plant," *Ecol. Inform.*, vol. 61, p. 101197, 2021.
- [25] L. M. Tassis, J. E. T. de Souza, and R. A. Krohling, "A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images," *Comput. Electron. Agric.*, vol. 186, p. 106191, 2021.

- [26] B. Liu, Y. Zhang, D. He, and Y. Li, "Identification of apple leaf diseases based on deep convolutional neural networks," *Symmetry (Basel)*, vol. 10, no. 1, p. 11, 2017.
- [27] A. Sembiring, Y. Away, F. Arnia, and R. Muharar, "Development of concise convolutional neural network for tomato plant disease classification based on leaf images," in *Journal of Physics: Conference Series*, 2021, vol. 1845, no. 1, p. 12009.
- [28] G. Sachdeva, P. Singh, and P. Kaur, "Plant leaf disease classification using deep Convolutional neural network with Bayesian learning," *Mater. Today Proc.*, vol. 45, pp. 5584–5590, 2021.
- [29] J. A. Pandian, V. D. Kumar, O. Geman, M. Hnatiuc, M. Arif, and K. Kanchanadevi, "Plant Disease Detection Using Deep Convolutional Neural Network," *Appl. Sci.*, vol. 12, no. 14, p. 6982, 2022.
- [30] M. A. Khan *et al.*, "Machine Learning-based Detection and Classification of Walnut Fungi Diseases," *Intell. Autom. SOFT Comput.*, vol. 30, no. 3, pp. 771–785, 2021.
- [31] A. Anagnostis *et al.*, "A deep learning approach for anthracnose infected trees classification in walnut orchards," *Comput. Electron. Agric.*, vol. 182, p. 105998, 2021.
- [32] A. Anagnostis, G. Asiminari, E. Papageorgiou, and D. Bochtis, "A convolutional neural networks based method for anthracnose infected walnut tree leaves identification," *Appl. Sci.*, vol. 10, no. 2, p. 469, 2020.
- [33] S. Khalesi, A. Mahmoudi, A. Hosainpour, and A. Alipour, "Detection of walnut varieties using impact acoustics and artificial neural networks (ANNs)," *Mod. Appl. Sci.*, vol. 6, no. 1, p. 43, 2012.
- [34] M. Esteki *et al.*, "Classification and authentication of Iranian walnuts according to their geographical origin based on gas chromatographic fatty acid fingerprint analysis using pattern recognition methods," *Chemom. Intell. Lab. Syst.*, vol. 171, pp. 251–258, 2017.
- [35] L. U. Jun, X. Z. Han, and K. J. Wang, "Classification of Collection Walnut Based on GLCM and SVM," in *2nd International Conference on Test, Measurement and Computational Method (TMCM 2017)*, 2017, pp. 276–281.
- [36] K. Öztürk and M. E. Şahin, "Yapay sinir ağları ve yapay zekâ'ya genel bir bakış," *Tak. Vekayi*, vol. 6, no. 2, pp. 25–36, 2018.
- [37] A. Bhardwaj, W. Di, and J. Wei, *Deep Learning Essentials: Your hands-on guide to the fundamentals of deep learning and neural network modeling*. Packt Publishing Ltd, 2018.
- [38] A. Ajit, K. Acharya, and A. Samanta, "A review of convolutional neural networks," in *2020 international conference on emerging trends in information*

technology and engineering (ic-ETITE), 2020, pp. 1–5.

- [39] M. To and A. R. D. Tezi, “T b m r g k t i y b e s a m g,” 2021.
- [40] Z. Kuş, “Mikrokanonikal Optimizasyon Algoritması ile Konvolüsyonel Sinir Ağlarında Hiper Parametrelerin Optimize Edilmesi.” Fatih Sultan Mehmet Vakıf Üniversitesi, Lisansüstü Eğitim Enstitüsü, 2019.
- [41] R. ZahediNasab and H. Mohseni, “Neuroevolutionary based convolutional neural network with adaptive activation functions,” *Neurocomputing*, vol. 381, pp. 306–313, 2020.
- [42] I. S. Isa, Z. Saad, S. Omar, M. K. Osman, K. A. Ahmad, and H. A. M. Sakim, “Suitable MLP network activation functions for breast cancer and thyroid disease detection,” in *2010 second international conference on computational intelligence, modelling and simulation*, 2010, pp. 39–44.
- [43] A. Baştürk, M. E. Yükei, H. Badem, and A. Çalışkan, “Deep neural network based diagnosis system for melanoma skin cancer,” in *2017 25th Signal Processing and Communications Applications Conference (SIU)*, 2017, pp. 1–4.
- [44] L. Shen, Q. Zhang, G. Cao, and H. Xu, “Fall detection system based on deep learning and image processing in cloud environment,” in *Conference on Complex, Intelligent, and Software Intensive Systems*, 2018, pp. 590–598.
- [45] Y.-C. Kim, K. R. Kim, and Y. H. Choe, “Automatic myocardial segmentation in dynamic contrast enhanced perfusion MRI using Monte Carlo dropout in an encoder-decoder convolutional neural network,” *Comput. Methods Programs Biomed.*, vol. 185, p. 105150, 2020.
- [46] M. A. KIZRAK and B. BOLAT, “Uçak Motoru Sağlığı için Uzun-Kısa Süreli Bellek Yöntemi ile Öngörücü Bakım,” *Bilişim Teknol. Derg.*, vol. 12, no. 2, pp. 103–109, 2019.
- [47] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [48] S. J. Cho, S. Kang, and C. Yoo, “A Resizable Mini-batch Gradient Descent based on a Randomized Weighted Majority,” Nov. 2017.
- [49] F. Kurt, “Evrişimli Sinir Ağlarında Hiper Parametrelerin Etkisinin İncelenmesi,” 2018.
- [50] M. Kaya and H. Ş. Bilge, “Deep metric learning: A survey,” *Symmetry (Basel)*, vol. 11, no. 9, p. 1066, 2019.
- [51] Y. You, Z. Zhang, J. Demmel, K. Keutzer, and C.-J. Hsieh, “Imagenet training in 24 minutes,” *arXiv Prepr. arXiv1709.05011*, 2017.

- [52] S. Haykin, *Neural networks and learning machines*, 3/E. Pearson Education India, 2009.
- [53] Y. Çelik, İ. YILDIZ, and A. T. Karadeniz, “Son Üç Yılda Geliştirilen Metasezgisel Algoritmalar Hakkında Kısa Bir İnceleme,” *Avrupa Bilim ve Teknol. Derg.*, pp. 463–477, 2019.
- [54] E. SEYYARER, F. AYATA, T. UÇKAN, and A. KARCI, “Derin Öğrenmede Kullanılan Optimizasyon Algoritmalarının Uygulanması Ve Kiyaslanması,” *Comput. Sci.*, vol. 5, no. 2, pp. 90–98, 2020.
- [55] E. BAŞARAN, “TİMPANİK MEMBRAN GÖRÜNTÜ ANALİZİ VE YAPAY ZEKA KULLANILARAK SANAL OTİTİS MEDİA TANI SİSTEMİNİN GELİŞTİRİLMESİ.” 2020.
- [56] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, “Efficient backprop,” in *Neural networks: Tricks of the trade*, Springer, 2012, pp. 9–48.
- [57] M. Moreira and E. Fiesler, “Neural networks with adaptive learning rate and momentum terms,” *Idiap*, 1995.
- [58] M. Kouzehgar, Y. K. Tamilselvam, M. V. Heredia, and M. R. Elara, “Self-reconfigurable façade-cleaning robot equipped with deep-learning-based crack detection based on convolutional neural networks,” *Autom. Constr.*, vol. 108, p. 102959, 2019.
- [59] D. T. T. Do, J. Lee, and H. Nguyen-Xuan, “Fast evaluation of crack growth path using time series forecasting,” *Eng. Fract. Mech.*, vol. 218, p. 106567, 2019.
- [60] S. Ruder, “An overview of gradient descent optimization algorithms,” *arXiv Prepr. arXiv1609.04747*, 2016.
- [61] M. D. Zeiler, “Adadelta: an adaptive learning rate method,” *arXiv Prepr. arXiv1212.5701*, 2012.
- [62] G. Ser and C. T. Bati, “Determining the best model with deep neural networks: keras application on mushroom data,” *Yüzüncü Yıl Üniversitesi J. Agric. Sci.*, vol. 29, no. 3, pp. 406–417, 2019.
- [63] V.-H. Nhu *et al.*, “Effectiveness assessment of Keras based deep learning with different robust optimization algorithms for shallow landslide susceptibility mapping at tropical area,” *Catena*, vol. 188, p. 104458, 2020.
- [64] Z. Chang, Y. Zhang, and W. Chen, “Electricity price prediction based on hybrid model of adam optimized LSTM neural network and wavelet transform,” *Energy*, vol. 187, p. 115804, 2019.
- [65] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 1097–1105, 2012.

- [66] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," 2017.
- [67] N. Dong, L. Zhao, C.-H. Wu, and J.-F. Chang, "Inception v3 based cervical cell classification combined with artificially extracted features," *Appl. Soft Comput.*, vol. 93, p. 106311, 2020.
- [68] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv Prepr. arXiv1409.1556*, 2014.
- [69] T. Carvalho, E. R. S. De Rezende, M. T. P. Alves, F. K. C. Balieiro, and R. B. Sovat, "Exposing computer generated images by eye's region classification via transfer learning of VGG19 CNN," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2017, pp. 866–870.
- [70] Y. Zheng, C. Yang, and A. Merkulov, "Breast cancer screening using convolutional neural network and follow-up digital mammography," in *Computational Imaging III*, 2018, vol. 10669, p. 1066905.
- [71] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019, pp. 6105–6114.
- [72] H. Alhichri, A. S. Alswayed, Y. Bazi, N. Ammour, and N. A. Alajlan, "Classification of remote sensing images using EfficientNet-B3 CNN model with attention," *IEEE access*, vol. 9, pp. 14078–14094, 2021.
- [73] W. Setiawan and A. Purnama, "Tobacco Leaf Images Clustering using DarkNet19 and K-Means," in *2020 6th Information Technology International Seminar (ITIS)*, 2020, pp. 269–273.
- [74] W. Wu, L. Guo, H. Gao, Z. You, Y. Liu, and Z. Chen, "YOLO-SLAM: A semantic SLAM system towards dynamic environment with geometric constraint," *Neural Comput. Appl.*, vol. 34, no. 8, pp. 6011–6026, 2022.
- [75] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
- [76] Z. Guo, Q. Chen, G. Wu, Y. Xu, R. Shibasaki, and X. Shao, "Village building identification based on ensemble convolutional neural networks," *Sensors*, vol. 17, no. 11, p. 2487, 2017.
- [77] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [78] Q. A. Al-Haija, M. A. Smadi, and S. Zein-Sabatto, "Multi-class weather classification using ResNet-18 CNN for autonomous IoT and CPS applications," in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2020, pp. 1586–1591.

- [79] Y. Dong, H. Zhang, C. Wang, and Y. Wang, "Fine-grained ship classification based on deep residual learning for high-resolution SAR images," *Remote Sens. Lett.*, vol. 10, no. 11, pp. 1095–1104, 2019.
- [80] L. Ali, F. Alnajjar, H. Al Jassmi, M. Gocho, W. Khan, and M. A. Serhani, "Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures," *Sensors*, vol. 21, no. 5, p. 1688, 2021.
- [81] A. BOUGUETTAYA, A. KECHIDA, and A. M. TABERKIT, "A survey on lightweight CNN-based object detection algorithms for platforms with limited computational resources," *Int. J. Informatics Appl. Math.*, vol. 2, no. 2, pp. 28–44, 2019.
- [82] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
- [83] U. Seidaliyeva, D. Akhmetov, L. Ilipbayeva, and E. T. Matson, "Real-time and accurate drone detection in a video with a static background," *Sensors*, vol. 20, no. 14, p. 3856, 2020.
- [84] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size," *arXiv Prepr. arXiv1602.07360*, 2016.
- [85] F. Ucar and D. Korkmaz, "COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images," *Med. Hypotheses*, vol. 140, p. 109761, 2020.
- [86] T. H. B. Nguyen, E. Park, X. Cui, V. H. Nguyen, and H. Kim, "fPADnet: Small and efficient convolutional neural network for presentation attack detection," *Sensors*, vol. 18, no. 8, p. 2532, 2018.
- [87] D. Chudasama, T. Patel, S. Joshi, and G. I. Prajapati, "Image segmentation using morphological operations," *Int. J. Comput. Appl.*, vol. 117, no. 18, 2015.
- [88] M. Goyal, "Morphological image processing," *IJCST*, vol. 2, no. 4, p. 59, 2011.
- [89] A. Dobrescu, M. V. Giuffrida, and S. A. Tsaftaris, "Doing more with less: a multitask deep learning approach in plant phenotyping," *Front. Plant Sci.*, vol. 11, p. 141, 2020.
- [90] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.
- [91] H. Jiang *et al.*, "A multi-label deep learning model with interpretable grad-CAM for diabetic retinopathy classification," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*,

2020, pp. 1560–1563.

- [92] Y. Zhang, D. Hong, D. McClement, O. Oladosu, G. Pridham, and G. Slaney, “Grad-CAM helps interpret the deep learning models trained to classify multiple sclerosis types using clinical brain magnetic resonance imaging,” *J. Neurosci. Methods*, vol. 353, p. 109098, 2021.
- [93] C. Unsalan and A. Ercil, “Classification of Rust Grades on Steel Surfaces Part 1,” *Inst. Grad. Stud. Sci. Eng. Bogaziçi Univ. Bebek, Istanbul, Turkey*, 1997.
- [94] E. Başaran, A. Şengür, Z. Cömert, Ü. Budak, Y. Çelik, and S. Velappan, “Normal and acute tympanic membrane diagnosis based on gray level co-occurrence matrix and artificial neural networks,” in *2019 international artificial intelligence and data processing symposium (IDAP)*, 2019, pp. 1–6.
- [95] E. BAŞARAN, Z. CÖMERT, and Y. CELİK, “Timpanik Membran Görüntü Özellikleri Kullanılarak Sınıflandırılması,” *Fırat Üniversitesi Mühendislik Bilim. Derg.*, vol. 33, no. 2, pp. 441–453, 2021.
- [96] D. Kumar, “Feature extraction and selection of kidney ultrasound images using GLCM and PCA,” *Procedia Comput. Sci.*, vol. 167, pp. 1722–1731, 2020.
- [97] E. Erdal, M. Kayri, and S. Ekinçi, “Kısıtlı optimizasyon problemlerinin çözümü için atom arama optimizasyon algoritması,” *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Derg.*, vol. 10, no. 3, pp. 841–851, 2019.
- [98] W. Zhao, L. Wang, and Z. Zhang, “Atom search optimization and its application to solve a hydrogeologic parameter estimation problem,” *Knowledge-Based Syst.*, vol. 163, pp. 283–304, 2019.
- [99] W. Zhao, L. Wang, and Z. Zhang, “A novel atom search optimization for dispersion coefficient estimation in groundwater,” *Futur. Gener. Comput. Syst.*, vol. 91, pp. 601–610, 2019.
- [100] F. A. ÖZBAY and E. ÖZBAY, “DİYABETİK RETİNOPATİ TESPİTİ İÇİN ATOM ARAMA OPTİMİZASYONU İLE ÖZELLİK SEÇİMİ YÖNTEMİ,” *Adıyaman Üniversitesi Mühendislik Bilim. Derg.*, vol. 9, no. 16, pp. 88–104.
- [101] S. Mirjalili and A. Lewis, “The whale optimization algorithm,” *Adv. Eng. Softw.*, vol. 95, pp. 51–67, 2016.
- [102] A. KARADENİZ and Y. ÇELİK, “Whale optimization algorithm for numerical constrained optimization,” *Acad. Platform-Journal Eng. Sci.*, vol. 8, no. 3, pp. 547–554, 2020.
- [103] R. K. Saidala and N. R. Devarakonda, “Bubble-net hunting strategy of whales based optimized feature selection for e-mail classification,” in *2017 2nd international conference for convergence in technology (I2CT)*, 2017, pp. 626–631.
- [104] A. Mostafa, A. E. Hassanien, M. Houseni, and H. Hefny, “Liver segmentation

- in MRI images based on whale optimization algorithm,” *Multimed. Tools Appl.*, vol. 76, no. 23, pp. 24931–24954, 2017.
- [105] I. Aljarah, H. Faris, and S. Mirjalili, “Optimizing connection weights in neural networks using the whale optimization algorithm,” *Soft Comput.*, vol. 22, no. 1, pp. 1–15, 2018.
- [106] M. Güngör, “Ki-kare testi üzerine,” *Fırat Üniversitesi Doğu Araştırmaları Derg.*, vol. 7, no. 1, pp. 84–89, 2008.
- [107] H. Budak, “Özellik seçim yöntemleri ve yeni bir yaklaşım,” *Süleyman Demirel Üniversitesi Fen Bilim. Enstitüsü Derg.*, vol. 22, pp. 21–31, 2018.
- [108] S. Inc, “SPSS Clementine 12.0 Algorithms Guide,” *SPSS Inc, Chicago*, 2007.
- [109] B. ÖZDET and İ. Semra, “AKCİĞER BİLGİSAYARLI TOMOGRAFİ GÖRÜNTÜLERİNDE GÖRÜNTÜ İŞLEME UYGULAMALARI İLE TÜMÖRLERİNİN TESPİT EDİLMESİ,” *Uludağ Üniversitesi Mühendislik Fakültesi Derg.*, vol. 27, no. 1, pp. 135–150.
- [110] Y. Çelik, E. Başaran, and Y. Dilay, “Identification of durum wheat grains by using hybrid convolution neural network and deep features,” *Signal, Image Video Process.*, pp. 1–8, 2022.
- [111] E. Başaran, “Classification of white blood cells with SVM by selecting SqueezeNet and LIME properties by mRMR method,” *Signal, Image Video Process.*, pp. 1–9, 2022.
- [112] B. Yang, R. Gong, L. Wang, and S. Yang, “Support vector machine in image recognition of nursing methods for critically ill blood purification,” *Microprocess. Microsyst.*, p. 103398, 2020.
- [113] A. Patle and D. S. Chouhan, “SVM kernel functions for classification,” in *2013 International Conference on Advances in Technology and Engineering (ICATE)*, 2013, pp. 1–9.
- [114] E. Fix and J. L. Hodges, “Discriminatory analysis. Nonparametric discrimination: Consistency properties,” *Int. Stat. Rev. Int. Stat.*, vol. 57, no. 3, pp. 238–247, 1989.
- [115] W. C. Parr and W. R. Schucany, “Minimum distance and robust estimation,” *J. Am. Stat. Assoc.*, vol. 75, no. 371, pp. 616–624, 1980.
- [116] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, “KNN model-based approach in classification,” in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 2003, pp. 986–996.
- [117] H. Takcı, “Performance-enhanced KNN algorithm-based heart disease prediction with the help of optimum parameters,” *J. Fac. Eng. Archit. Gazi Univ.*, vol. 38, no. 1, pp. 451–460, 2023.

- [118] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, 2006, vol. 2, pp. 2126–2136.
- [119] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, 2007.
- [120] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009.
- [121] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, 1995, vol. 14, no. 2, pp. 1137–1145.
- [122] P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-validation.," *Encycl. database Syst.*, vol. 5, pp. 532–538, 2009.
- [123] J. Wei Tan, S.-W. Chang, S. Abdul-Kareem, H. J. Yap, and K.-T. Yong, "Deep learning for plant species classification using leaf vein morphometric," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 17, no. 1, pp. 82–90, 2018.
- [124] U. M. Khaire and R. Dhanalakshmi, "Stability of feature selection algorithm: A review," *J. King Saud Univ. Inf. Sci.*, 2019.
- [125] W. Q. Hammadi and O. S. Qasim, "Hybrid Binary Atom Search Optimization Approaches with Statistical Dependence for Feature Selection," in *2022 International Conference on Computer Science and Software Engineering (CSASE)*, 2022, pp. 218–223.
- [126] J. Yuan, Y. Zhang, Z. Shi, X. Geng, J. Fan, and Y. Rui, "Balanced masking strategy for multi-label image classification," *Neurocomputing*, vol. 522, pp. 64–72, 2023.
- [127] C. Hou, Y. Li, X. Chen, and J. Zhang, "Automatic modulation classification using KELM with joint features of CNN and LBP," *Phys. Commun.*, vol. 45, p. 101259, 2021.

ÖZGEÇMİŞ

Alper Talha KARADENİZ ilk ve orta öğretimini Ordu Altıncıdık İlköğretim Okulu'nda, lise öğretimini ise Ordu Özel Seçkin Kolej Fen Lisesi'nde tamamladı. 2011 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği'nden başarı ile mezun oldu. Ankara'da bir özel şirkette yazılımcı olarak çalıştıktan sonra 2014 yılında Ordu Üniversitesi Mesudiye Meslek Yüksekokulu'nda öğretim görevlisi olarak çalışmaya başladı. Ardından Bolu Abant Baysal Üniversitesi Teknik Bilimler Meslek Yüksekokulu ve Trabzon Üniversitesi'nde öğretim görevlisi olarak hizmet verdi. Halen Trabzon Üniversitesi Dijital Dönüşüm ve Yazılım Ofisi'nde öğretim görevlisi olarak çalışmaktadır. 2015 yılında, Karabük Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında yüksek lisans eğitimini tamamladıktan sonra, 2017 yılında aynı Anabilim Dalında Doktora eğitimine başlamıştır. İki çocuk babası olan Karadeniz, iyi düzeyde İngilizce bilmektedir.