



T.C.
NECMETTİN ERBAKAN
ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



**BALİNA OPTİMİZASYON ALGORİTMASI İLE DERİN ÖĞRENME
ALGORİTMALARININ PERFORMANSININ ARTIRILMASI**

Ömer ÖZCAN

DOKTORA TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Mart - 2026
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Ömer ÖZCAN tarafından hazırlanan "*Balina Optimizasyon Algoritması ile Derin Öğrenme Algoritmalarının Performansının Artırılması*" adlı tez çalışması 27/03/2026 tarihinde aşağıdaki jüri tarafından oy birliği ile Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda DOKTORA TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Doç. Dr. Onur İNAN

Danışman

Prof. Dr. Mehmet HACIBEYOĞLU

Üye

Doç. Dr. Muhammed KARAALTUN

Üye

Dr. Öğr. Üyesi Vahit TONGUR

Üye

Dr. Öğr. Üyesi Alperen EROĞLU

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun / .../20.. gün ve sayılı kararıyla onaylanmıştır.

Prof. Dr. Havvanur UÇBEYİAY
FBE Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

İmza

Ömer ÖZCAN

Tarih:

ÖZET

DOKTORA TEZİ

BALİNA OPTİMİZASYON ALGORİTMASI İLE DERİN ÖĞRENME ALGORİTMALARININ PERFORMANSININ ARTIRILMASI

Ömer ÖZCAN

NECMETTİN ERBAKAN ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Danışman: Prof. Dr. Mehmet HACİBEYOĞLU

2026, 93 Sayfa

Jüri

Prof. Dr. Mehmet HACİBEYOĞLU

Doç. Dr. Onur İNAN

Doç. Dr. Muhammed KARAALTUN

Dr. Öğr. Üyesi Vahit TONGUR

Dr. Öğr. Üyesi Alperen EROĞLU

Bu doktora tezinde, görüntü sınıflandırma ve nesne tespiti problemlerinde derin öğrenme modellerinin performansını artırmak amacıyla görüntü artırma, öznetelik çıkarımı ve model optimizasyonu bütünlük bir çerçevede ele alınmıştır. Bu kapsamda, kenar bilgisine dayalı Balina Optimizasyon Algoritması (BOA) tabanlı bir görüntü artırma yöntemi geliştirilmiştir. Önerilen yaklaşımda, görüntü içeriğine en uygun çekirdek filtre metasezgisel optimizasyonla belirlenmiş, bu filtre ile gerçekleştirilen kenar tespiti çıktıları kullanılarak sentetik örnekler üretilmiştir. Elde edilen sonuçlar, dengesiz veri setlerinde sınıflandırma doğruluğunun %74,6'dan %98'e yükseldiğini ve kenar tespiti performansının karşılaştırmalı yöntemlere kıyasla iyileştiğini göstermektedir. Ayrıca, popülasyon çeşitliliğini kosinüs benzerliği temelli bir ölçüm mekanizmasıyla koruyan Geliştirilmiş Balina Optimizasyon Algoritması (GBOA) önerilmiştir. GBOA, öznetelik çıkarımı ve sınıflandırma bileşenlerinin hiperparametre ve model parametrelerinin optimize edilmesinde kullanılarak bütünlük bir Çilek Tespit Sistemi'nin geliştirilmesini sağlamıştır. Önerilen sistem, çilek olgunluk tespitinde %98,23 doğruluk oranına ulaşarak yüksek performans sergilemiştir.

Anahtar Kelimeler: Balina Optimizasyon Algoritması, Derin Öğrenme, Görüntü Artırımı, Görüntü Sınıflandırma, Kenar Tespiti, Metasezgisel Optimizasyon, Nesne Tespiti, Veri Dengesizliği

ABSTRACT

Ph.D THESIS

ENHANCING THE PERFORMANCE OF DEEP LEARNING ALGORITHMS WITH THE WHALE OPTIMIZATION ALGORITHM

Ömer ÖZCAN

**THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF
NECMETTİN ERBAKAN UNIVERSITY
THE DEGREE OF DOCTOR OF PHILOSOPHY IN COMPUTER
ENGINEERING**

Advisor: Prof. Dr. Mehmet HACİBEYOĞLU

2026, 93 Pages

Jury

Advisor Prof. Dr. Mehmet HACİBEYOĞLU

Assoc. Prof. Dr. Onur İNAN

Assoc. Prof. Dr. Muhammed KARAALTUN

Assist. Prof. Dr. Vahit TONGUR

Assist. Prof. Dr. Alperen EROĞLU

This doctoral dissertation presents an integrated framework that combines image augmentation, feature extraction, and model optimization to improve the performance of deep learning models for image classification and object detection tasks. Within this framework, an edge-informed Whale Optimization Algorithm (WOA)-based image augmentation method is developed. In the proposed method, the kernel filter best suited to image content is identified via metaheuristic optimization, and the resulting edge information is used to generate synthetic samples. Experimental results demonstrate that classification accuracy on imbalanced datasets increases from 74.6% to 98%, while edge detection performance surpasses that of the compared baseline approaches. In addition, an Improved Whale Optimization Algorithm (IWOA), which preserves population diversity through a cosine similarity-based mechanism, is introduced. The proposed IWOA is employed to optimize both hyperparameters and model parameters of the feature extraction and classification modules, leading to the development of an integrated Strawberry Detection System. The resulting system achieves 98.23% accuracy in strawberry ripeness detection, indicating strong overall performance.

Keywords: Deep Learning, Edge Detection, Image Augmentation, Image Classification, Metaheuristic Optimization, Object Detection, Whale Optimization Algorithm, Data Imbalance

ÖNSÖZ

Tez çalışmam boyunca değerli bilgi ve deneyimleriyle bana yol gösteren, karşılaştığım zorluklarda destek ve anlayışını esirgemeyen danışmanım Sayın Prof. Dr. Mehmet HACIBEYOĞLU'na en içten teşekkürlerimi sunarım. Tez izleme komitemde yer alarak çalışmama değerli katkılarda bulunan Sayın Dr. Öğr. Üyesi Alperen EROĞLU'na ve Sayın Dr. Öğr. Üyesi Vahit TONGUR'a görüş ve önerileri için teşekkür ederim. Tez savunma jürimde yer alan Sayın Doç. Dr. Muhammed KARAALTUN'a ve Sayın Doç. Dr. Onur İNAN'a kıymetli değerlendirmeleri için teşekkür ederim. Bizleri yetiştiren, bu günlere getiren sevgili anneme ve babama sonsuz teşekkür ederim. Tez sürecinde sabır ve anlayışlarıyla yanımda olan, bu çalışmaya manen destek veren sevgili eşime ve çocuklarıma da içten teşekkürlerimi sunarım. Bu çalışmanın ortaya çıkmasında doğrudan veya dolaylı katkısı bulunan, adını sayamadığım tüm arkadaşlarıma ve meslektaşlarıma teşekkürü bir borç bilirim.

Bu tezin, ilgili alanda yapılacak sonraki araştırmalara ışık tutmasını ve bilim dünyasına küçük de olsa bir katkı sağlamasını dilerim.

Ömer ÖZCAN
KONYA-2026

İÇİNDEKİLER

ÖZET	iv
ABSTRACT	v
ÖNSÖZ	vi
ŞEKİLLER LİSTESİ	x
ÇİZELGELER LİSTESİ	xi
SİMGELER VE KISALTMALAR	xii
1. GİRİŞ	1
2. KAYNAK ARAŞTIRMASI	6
3. MATERYAL VE YÖNTEM	13
3.1. Balina Optimizasyon Algoritması	13
3.1.1. Balina optimizasyon algoritması biyolojik temelleri	13
3.1.2. Matematiksel formülasyon	14
3.1.2.1. Avlanma operatörü	14
3.1.2.2. Baloncuk-ağ saldırısı	15
3.1.2.3. Av arama	16
3.1.3. Algoritmanın sözde kodu	17
3.1.4. Algoritmanın özellikleri	18
3.1.5. Algoritmanın avantajları ve sınırlamaları	19
3.2. Balina Optimizasyon Algoritması Tabanlı Görüntü Artırımı Yöntemi	20
3.2.1. Balina optimizasyon algoritması tabanlı görüntü artırımı yönteminin genel mimarisi	21
3.2.2. Arama ajanı temsili	21
3.2.3. Amaç fonksiyonu	22
3.2.4. Kenar tespiti entegrasyonu	23
3.2.4.1. Çok katmanlı filtre yapısı	23
3.2.4.2. Optimum kernel filtresi	24
3.2.4.3. Filtre entegrasyonu ve kenar tespiti	25
3.2.5. Kenar tabanlı görüntü artırma	26
3.2.6. Balina optimizasyon algoritması tabanlı görüntü artırma yönteminin özellikleri	28
3.3. Geliştirilmiş Balina Optimizasyon Algoritması	29
3.3.1. Gboa'nın motivasyonu ve temel prensibi	29
3.3.2. Popülasyon çeşitliliği izleme	29
3.3.3. Geliştirilmiş spiral konum güncelleme mekanizması	29
3.3.4. Gboa'nın sözde kodu	30
3.3.5. Gboa'nın avantajları	31

3.4. S-Det: Çilek Tespit Sistemi	32
3.4.1. S-det sisteminin genel mimarisi	32
3.4.2. Segmentasyon aşaması	33
3.4.3. Gboa tabanlı otomatik kodlayıcı	33
3.4.3.1. Otokodlayıcı mimarisi	34
3.4.3.2. Aktivasyon fonksiyonu	35
3.4.3.3. Uygunluk fonksiyonu	35
3.4.4. Geliştirilmiş balina optimizasyon algoritması tabanlı dnn	35
3.4.4.1. Derin sinir ağı (deep neural network, dnn) mimarisi	36
3.4.4.2. Ağırlık optimizasyonu	37
3.4.5. Sınıf yoğunluğu hesaplama	37
3.4.6. S-det sisteminin özellikleri	37
3.5. Kullanılan Veri Setleri	38
3.5.1. Boa-ga ile birlikte kullanılan veri setleri	38
3.5.2. Gboa ve s-det ile birlikte kullanılan veri setleri	38
3.5.2.1. Sınıflandırma veri setleri	39
3.5.2.2. Strawtex veri seti	40
3.5.2.3. Görüntü veri setleri	40
4. DENEYSEL SONUÇLAR	41
4.1. BOA-GA Yönteminin DeneySEL Sonuçları	41
4.1.1. DeneySEL kurulum ve değerlendirme metrikleri	41
4.1.2. Boa tabanlı kenar tespiti yönteminin performans değerlendirmesi	42
4.1.3. DeneySEL kurulum	45
4.1.4. Balina optimizasyon algoritması tabanlı görüntü artırma yönteminin sınıflandırma sonuçları	46
4.1.5. Kenar tabanlı artırım yöntemleri ile karşılaştırma	46
4.1.6. Literatürdeki güncel yöntemler ile karşılaştırma	47
4.2. GBOA ve S-Det Sisteminin DeneySEL Sonuçları	48
4.2.1. Gboa'nın karşılaştırmalı optimizasyon testleri	49
4.2.1.1. Kullanılan benchmark sınıflandırma veri setleri	49
4.2.1.2. Optimizasyon algoritmalarının parametreleri	49
4.2.1.3. Gboa'nın benchmark veri setleri üzerindeki sonuçları	49
4.2.2. S-det sisteminin çilek uygunluk tespiti performansı	50
4.2.2.1. Strawtex veri seti	52
4.2.2.2. S-det sisteminin sınıflandırma sonuçları	52
4.2.3. S-det sisteminin diğer görüntü sınıflandırma problemlerindeki performansı	54
5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA	56
5.1. BOA-GA Yönteminin Sonuçlarının Yorumlanması ve Tartışılması	56
5.1.1. Boa tabanlı kenar tespiti yönteminin başarısının analizi	56
5.1.2. Boa-ga yönteminin dengesiz veri setlerindeki performans analizi	57
5.1.3. Literatür karşılaştırmalarının değerlendirilmesi	58
5.2. GBOA ve S-Det Sisteminin Sonuçlarının Yorumlanması ve Tartışılması ...	59
5.2.1. Gboa'nın optimizasyon performansının analizi	59

5.2.2. S-det sisteminin çilek olgunluk tespiti performansının analizi	60
5.2.3. S-det sisteminin benchmark görüntü veri setlerindeki performans analizi	62
5.3. Metodolojik Yaklaşımların Genel Değerlendirmesi	63
5.3.1. Güçlü yönler	63
5.3.2. Zayıf yönler ve sınırlamalar	64
5.3.3. Olası çözüm yolları	64
5.4. Çalışmanın Literatüre Katkıları	66
5.4.1. Birinci aşama katkıları	66
5.4.2. İkinci aşama katkıları	66
5.5. Pratik Uygulamalar	67
6. SONUÇ VE ÖNERİLER	68
6.1. Ana Bulguların Özeti	68
6.1.1. Birinci aşama bulguları	68
6.1.2. İkinci aşama bulguları	68
6.2. Bilimsel ve Pratik Anlamlar	68
6.2.1. Bilimsel katkılar	68
6.2.2. Pratik uygulamalar ve etkiler	69
6.3. Çalışmanın Sınırlılıkları	69
6.3.1. Metodolojik sınırlılıklar	69
6.3.2. Deneysel sınırlılıklar	69
6.3.3. Teorik sınırlılıklar	69
6.4. Gelecekteki Araştırmalar için Öneriler	69
6.4.1. Algoritmik ve metodolojik geliştirmeler	70
6.4.2. Uygulama ve teorik yönelimler	70
6.5. Genel Değerlendirme	70
7. KAYNAKLAR	71

ŞEKİLLER LİSTESİ

Şekil 3.1.	Kambur balinaların baloncuk-ağ ile avlanma davranışı.	14
Şekil 3.2.	Avlanma operatörü.	15
Şekil 3.3.	Spiral hareket mekanizması.	16
Şekil 3.4.	Keşif ve sömürü fazları.	17
Şekil 3.5.	BOA akış diyagramı.	18
Şekil 3.6.	$\vec{a}$ katsayısının değişimi ve keşif-sömürü dengesi.	19
Şekil 3.7.	BOA-GA yönteminin genel mimarisi.	21
Şekil 3.8.	Arama ajanı vektörünün yapısı.	22
Şekil 3.9.	Çok katmanlı filtre yapısının mimarisi.	24
Şekil 3.10.	OF yapısı ve uygulanması.	25
Şekil 3.11.	Kenar görüntüsünün sentetik görüntü üzerine açısal yerleştirilmesi.	27
Şekil 3.12.	Orijinal ve BOA-GA ile artırılmış görüntü örnekleri.	28
Şekil 3.13.	BOA ve GBOA'nın karşılaştırması.	30
Şekil 3.14.	S-Det sisteminin genel mimarisi.	32
Şekil 3.15.	Segmentasyon aşamasından elde edilen 32x32 piksel boyutundaki çilek dokuları.	33
Şekil 3.16.	GBOA-OK yönteminin genel mimarisi.	34
Şekil 3.17.	GBOA-DNN yönteminin genel mimarisi.	36
Şekil 4.1.	Kenar tespit edicilerin görsel karşılaştırması	44
Şekil 4.2.	Önerilen yöntem için 10 katlı çapraz doğrulama sonuçları	54

ÇİZELGELER LİSTESİ

Çizelge 3.1.	BOA-GA deneylerinde kullanılan veri setlerinin özellikleri	39
Çizelge 3.2.	Sınıflandırma veri setleri	39
Çizelge 3.3.	Görüntü veri setleri	40
Çizelge 4.1.	Sınıflandırma için karışıklık matrisi yapısı	42
Çizelge 4.2.	Kenar tespiti için piksel bazlı karşılaştırma karışıklık matrisi ..	44
Çizelge 4.3.	Önerilen yöntem ile kenar tespit edicilerin karşılaştırması	45
Çizelge 4.4.	BOA-GA yönteminin görüntü veri setleri üzerindeki performans değerlendirmesi	46
Çizelge 4.5.	HAM10000 veri seti üzerinde BOA-GA ve kenar tespit ediciler aşırı örnekleme değerlendirmesi	47
Çizelge 4.6.	HAM10000 veri seti aşırı örnekleme performans değerlendirmeleri	48
Çizelge 4.7.	Optimizasyon algoritmaları için parametreler	50
Çizelge 4.8.	GBOA'nın kapsamlı deneysel sonuçları	51
Çizelge 4.9.	StrawTex veri seti için karşılaştırmalı test sonuçları	53
Çizelge 4.10.	S-DET literatür karşılaştırması	55

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

SİMGELER

I	Görüntü
I_{filtered}	Filtrelenmiş görüntü
I_{edge}	Kenar görüntüsü
$G_{\sigma}(I)$	Standart sapması σ olan Gaussian filtresi ile filtrelenmiş görüntü
$OF(I)$	Optimum kernel filtresi ile filtrelenmiş görüntü
σ	Standart sapma parametresi
σ_i	i -inci katmanın standart sapma değeri
α	Karıştırma parametresi (döndürme açısı veya ağırlık katsayısı)
β	Karıştırma parametresi (benzerlik katsayısı veya ağırlık katsayısı)
λ_i	i -inci katmanın ağırlık katsayısı
λ_n	n -inci benzerlik metriğinin ağırlık katsayısı
θ	Açı parametresi
δ	Yerleştirme mesafesi
G_{width}	Görüntü genişliği
G_{height}	Görüntü yüksekliği
N	Örnek sayısı veya nöron sayısı
M_j	Çoğunluk sınıfındaki örnek sayısı
M_i	Azınlık sınıfındaki örnek sayısı
P_v	Yüzde değişimi
TP	Gerçek pozitif
TN	Gerçek negatif
FP	Yanlış pozitif
FN	Yanlış negatif
y_i	Gerçek çıktı
y'_i	Tahmin edilen çıktı
n	Eğitim veri setindeki örnek sayısı

cs	Kosinüs benzerliği değeri
r	Referans görüntü
t	Tespit edilen görüntü
(x, y)	Piksel koordinatları

KISALTMALAR

ABC	Yapay Arı Kolonisi
ACO	Karınca Kolonisi Optimizasyonu
ACU	Doğruluk
BOA	Balina Optimizasyon Algoritması
BOA-GA	Balina Optimizasyon Algoritması Tabanlı Görüntü Artırımı
BP	Geri Yayılım
CNN	Evrişimli Sinir Ağı
DE	Diferansiyel Evrim
DNN	Derin Sinir Ağı
ES	Evrin Stratejileri
F1	F1-Skoru
FN	Yanlış Negatif
FNR	Yanlış Negatif Oranı
FP	Yanlış Pozitif
FPR	Yanlış Pozitif Oranı
GA	Genetik Algoritma
GAN	Üretici Çekişmeli Ağ
GBOA	Geliştirilmiş Balina Optimizasyon Algoritması
GBOA-DNN	GBOA Tabanlı Derin Sinir Ağı
GBOA-OK	GBOA Tabanlı Optimum Kernel
GSKGM	Geliştirilmiş Spiral Konum Güncelleme Mekanizması
KNN	K-En Yakın Komşu
MSE	Ortalama Kare Hatası
OF	Optimum Kernel Filtresi
PBIL	Popülasyon Tabanlı Artırmalı Öğrenme

PRE	Kesinlik
PSO	Parçacık Sürü Optimizasyonu
REC	Duyarlılık
RF	Rastgele Orman
S-Det	Çilek Tespit Sistemi
SAM	Spektral Açı Haritası
SVM	Destek Vektör Makinesi
TN	Doğru Negatif (True Negative)
TP	Doğru Pozitif (True Positive)
VAE	Varyasyonel Otokodlayıcı
VIF	Görsel Bilgi Fidelity



1. GİRİŞ

Derin öğrenme teknolojilerinin, özellikle bilgisayarlı görü alanında son yıllarda gösterdiği olağanüstü gelişim, otomasyon ve karar destek sistemlerinin pek çok sektörde yaygınlaşmasına olanak sağlamıştır (Alzubaidi vd., 2021; Khan vd., 2022). Bu teknolojilerin laboratuvar ortamından gerçek dünya uygulamalarına başarılı bir şekilde aktarılması ise, temel araştırma problemlerinin çözümünü gerektirmektedir (Chen vd., 2023). Bu problemlerin başında, derin öğrenme modellerinin performansını doğrudan etkileyen iki kritik faktör gelmektedir: birinci faktör, eğitim verisinin kalite ve nicelik açısından yeterliliği; ikinci faktör ise model parametrelerinin etkin bir şekilde optimizasyonudur.

Gerçek dünya uygulamalarında sıklıkla karşılaşılan veri dengesizliği problemi, özellikle tıbbi teşhis, endüstriyel kalite kontrol ve tarımsal görüntü analizi gibi alanlarda kritik öneme sahiptir (Wang vd., 2021; Zhang vd., 2023). Bu tür uygulamalarda, azınlık sınıflarının (hastalıklı örnekler, kusurlu ürünler, belirli olgunluk aşamasındaki meyveler vb.) doğru tespiti büyük önem taşımakta, ancak mevcut veri setlerinde bu sınıflara ait örnek sayısı genellikle sınırlı kalmaktadır. Bu durum, geleneksel derin öğrenme modellerinin bu azınlık sınıflarını tespit etmede yetersiz kalmasına neden olmaktadır (Li vd., 2022; Zhang vd., 2023).

Diğer yandan, derin sinir ağlarının eğitiminde kullanılan gradyan tabanlı optimizasyon yöntemleri, karmaşık ve yüksek boyutlu parametre uzaylarında yerel minimumlara saplanma, yakınsama hızının yavaş olması ve hiperparametre seçimine aşırı bağımlılık gibi temel sınırlamalarla karşı karşıyadır (Reddi vd., 2021; Wang ve Li, 2023). Bu sınırlamalar, modellerin potansiyel performanslarına ulaşmalarını engelleyebilmekte ve eğitim süreçlerini verimsiz hale getirebilmektedir.

Bu bağlamda, bu tez çalışması, derin öğrenme tabanlı sistemlerin gerçek dünya uygulamalarındaki performansını artırmaya yönelik iki temel probleme odaklanmaktadır:

- Veri dengesizliği problemine etkin çözümlerin geliştirilmesi,
- Derin sinir ağlarının optimizasyon süreçlerinin iyileştirilmesi.

Bu problemlere getirilecek çözümlerin, özellikle tarımsal görüntü analizi gibi alanlarda otomasyon sistemlerinin geliştirilmesine önemli katkılar sağlaması beklenmektedir (Sun vd., 2022; Patel vd., 2023).

Birinci Aşama: Veri Dengesizliği Probleminin Optimizasyon Tabanlı Çözümü

Tez çalışmasının birinci aşaması, veri dengesizliği problemine odaklanarak, görüntü artırma süreçlerinin optimizasyon tabanlı bir yaklaşımla iyileştirilmesini hedeflemiştir. Bu aşamada, geleneksel veri artırma yöntemlerinin sabit parametre sınırlamasını aşmak için, görüntülerin yapısal bilgisini temel alan yenilikçi bir yaklaşım geliştirilmiştir (Cubuk vd., 2020; Hataya vd., 2020). Bu yaklaşımın temel prensibi, veri artırma sürecinin pasif bir ön işlem adımı olmaktan çıkarılıp, görüntü içeriğine göre dinamik olarak optimize edilen aktif bir süreç haline getirilmesidir (Zheng vd., 2021; Kim vd., 2022).

Bu aşamada, Balina Optimizasyon Algoritması (Whale Optimization Algorithm, BOA) kullanılarak görüntü artırma parametrelerinin optimizasyonu gerçekleştirilmiştir. BOA, doğadaki balinaların avlanma davranışlarından esinlenerek geliştirilmiş bir metasezgisel optimizasyon algoritmasıdır (Mirjalili ve Lewis, 2016) ve özellikle sürekli optimizasyon problemlerinde etkili sonuçlar vermektedir. Görüntü artırma sürecinde, her görüntü için en uygun dönüşüm parametrelerini (rotasyon açısı, ölçekleme faktörü, kesme miktarı vb.) belirlemek üzere BOA kullanılmıştır. Bu tercih, BOA'nın sürekli parametre uzaylarında etkin arama yapabilmesi sayesinde, veri artırma parametrelerinin otomatik ve içerik-duyarlı biçimde belirlenmesini hedefleyen çalışmalarla da örtüşmektedir (Singh vd., 2022; Kumar vd., 2023).

Bu aşamanın önemli bir bileşeni, görüntülerin yapısal bilgisinin korunması için kenar bilgisinin artırma sürecine entegre edilmesidir. Görüntü artırma işlemleri sırasında, özellikle dönüşüm ve ölçekleme gibi geometrik dönüşümler uygulanırken, görüntülerin kenar bilgisi gibi kritik yapısal özelliklerinin bozulmaması gerekmektedir (Soria vd., 2020). Bu nedenle, BOA'nın uygunluk fonksiyonu, hem sınıflandırma performansını hem de kenar korunumunu dikkate alacak şekilde tasarlanmıştır. Bu çalışmada kenarlar, BOA tarafından optimize edilen çok katmanlı filtre yapısı (Gaussian filtreleri ve optimum kernel filtresi) ile üretilmekte; Canny, Laplacian, Roberts ve Sobel gibi kenar tespiti algoritmaları ise referans kenar görüntülerinin oluşturulmasında ve karşılaştırmalı değerlendirmelerde kullanılmaktadır. Bu yaklaşım, veri artırma sürecinde yapısal bilginin korunmasına odaklanan çalışmalarla uyumlu olup, kenar bilgisinin artırma stratejisine bilinçli biçimde dâhil edilmesini gerektirmektedir (Zhang vd., 2022).

Birinci aşamada, geliştirilen Balina Optimizasyon Algoritması Tabanlı Görüntü

Artırımı (Whale Optimization Algorithm-based Image Augmentation, BOA-GA) yöntemi, dengesiz veri setleri üzerinde test edilmiş ve geleneksel veri artırma yöntemlerine kıyasla önemli performans iyileştirmeleri elde edilmiştir. Özellikle, azınlık sınıflarının sınıflandırma doğruluğunda belirgin artışlar gözlemlenmiştir. Bu sonuçlar, optimizasyon tabanlı veri artırma yaklaşımının etkinliğini göstermektedir (Chen vd., 2022; Li vd., 2023).

İkinci Aşama: Bütünleşik Derin Öğrenme Mimarisi ve Çok Aşamalı Optimizasyon Yaklaşımı

Birinci aşamada elde edilen başarılı sonuçlar, optimizasyon tabanlı yaklaşımların derin öğrenme sistemlerinde daha kapsamlı bir şekilde kullanılabileceği fikrini doğrulamıştır. İkinci aşamada, bu yaklaşım bir adım daha ileri götürülerek, aynı optimizasyon algoritmasının bir görüntü analizi boru hattının farklı aşamalarında koordineli bir şekilde kullanılması prensibine dayanan bütünleşik bir sistem tasarımı gerçekleştirilmiştir.

İkinci aşamanın temel motivasyonu, derin öğrenme sistemlerinin optimizasyon süreçlerinin iyileştirilmesi ve gerçek dünya uygulamalarına daha etkin bir şekilde adapte edilmesidir (Aljarah vd., 2022). Bu amaçla, standart BOA'nın popülasyon çeşitliliği kaybı sorununu ele alan Geliştirilmiş Balina Optimizasyon Algoritması (Improved Whale Optimization Algorithm, GBOA) önerilmiştir. GBOA, algoritmanın keşif yeteneğini koruyarak daha etkili optimizasyon performansı sağlamaktadır. Nitekim güncel çalışmalar, BOA tabanlı iyileştirmelerin çoğunlukla yakınsama davranışı ile popülasyon çeşitliliğinin korunması ekseninde şekillendiğini göstermekte; önerilen GBOA da bu motivasyonu doğrudan hedefleyerek çeşitlilik kaybını azaltmayı amaçlamaktadır (Kumar vd., 2023; Patel vd., 2024).

Geliştirilen Çilek Tespit Sistemi (Strawberry Detection System, S-Det), görüntü analizi sürecinin iki kritik aşamasında GBOA'nı kullanmaktadır: öznitelik çıkarımı ve sınıflandırma. Öznitelik çıkarımı aşamasında, otokodlayıcı (autoencoder) mimarisinin eğitim parametreleri GBOA ile optimize edilerek, daha anlamlı ve ayırt edici özniteliklerin çıkarılması sağlanmıştır (Zhao vd., 2021). Sınıflandırma aşamasında ise, derin sinir ağı sınıflandırıcısının ağırlık parametreleri GBOA ile optimize edilerek, daha yüksek sınıflandırma performansı elde edilmiştir. Bu iki alt problemin aynı optimizasyon çerçevesinde ele alınması, metasezgisel yöntemlerin temsil öğrenimi otokodlayıcı ile sınıflandırıcı optimizasyonunu birlikte iyileştirebilecek esneklikte

olduğunu göstermektedir (Zhang vd., 2023).

Tarımsal Görüntü Analizi Uygulaması: Çilek Olgunluk Tespiti

Geliştirilen metodolojilerin gerçek dünya uygulamalarındaki etkinliğini değerlendirmek amacıyla, tarımsal görüntü analizi alanında çilek olgunluk tespiti ve sınıflandırması problemi seçilmiştir. Bu seçimin temel nedeni, tarımsal uygulamaların hem veri dengesizliği hem de optimizasyon problemlerinin tipik örneklerini içermesi ve aynı zamanda ekonomik ve sosyal açıdan büyük öneme sahip olmasıdır (Fuentes vd., 2021). Bu nedenle çilek olgunluk tespiti, sahaya özgü değişken görüntü koşulları ile sınıf dengesizliğini birlikte barındırması sayesinde, önerilen optimizasyon tabanlı yaklaşımın gerçek dünya geçerliliğini sınamak için uygun bir vaka çalışması sunmaktadır (Patel vd., 2023; Kim vd., 2024).

Çilek olgunluk tespiti problemi, meyvelerin doğal ortamlarındaki görüntülerinden hem meyvenin tespit edilmesini hem de olgunluk seviyesinin belirlenmesini gerektirmektedir. Bu problem, doğası gereği dengesiz bir veri seti içermektedir çünkü bazı olgunluk aşamaları (örneğin, tam olgun meyveler) diğerlerine göre daha az sayıda örnek içerebilmektedir (Yu vd., 2019). Ayrıca, görüntülerin çekildiği koşullar (ışık, kamera açısı, arka plan vb.) değişkenlik gösterebilmekte, bu da sınıflandırma problemini daha da zorlaştırmaktadır (Wang vd., 2022; Chen vd., 2023).

Geliştirilen S-Det sistemi, çilek olgunluk tespiti problemi üzerinde test edilmiş ve mevcut yöntemlere kıyasla önemli performans iyileştirmeleri elde edilmiştir. Sistem, hem öznitelik çıkarımı hem de sınıflandırma aşamalarında GBOAnı kullanarak, daha yüksek doğruluk oranları ve daha dengeli sınıflandırma sonuçları üretmiştir. Özellikle, azınlık sınıflarının tespitinde belirgin iyileştirmeler gözlemlenmiştir. Bu sonuçlar, optimizasyon tabanlı yaklaşımların tarımsal görüntü işleme uygulamalarındaki etkinliğini göstermektedir (Patel vd., 2023; Kim vd., 2024).

Aşamaların Bütünleşik Yaklaşımı ve Metodolojik İlerleme

Bu tez çalışması, kademeli ve birbirini tamamlayan iki ana araştırma aşamasından oluşan bütünleşik bir yaklaşım benimsemektedir. Birinci aşamada, veri dengesizliği problemine yönelik optimizasyon tabanlı bir çözüm geliştirilmiş ve bu çözümün etkinliği deneysel olarak doğrulanmıştır. İkinci aşamada ise, bu metodolojik deneyim ve bilgi birikiminden yararlanılarak, daha karmaşık ve yüksek boyutlu optimizasyon problemlerine yönelik geliştirilmiş bir yaklaşım önerilmiştir.

Her iki çalışmada da deneysel doğrulama ve karşılaştırmalı analizlere büyük

önem verilmiştir. Performans değerlendirmesinde standart metrikler (doğruluk, kesinlik, duyarlılık, F1-skoru, yanlış pozitif/negatif oranları) kullanılmış ve istatistiksel analizlerle (çoklu çalıştırmalar, ortalama ve standart sapma, Friedman sıralama testi) sonuçların güvenilirliği sağlanmıştır.

Tezin Yapısı

Bu tez çalışması şu şekilde organize edilmiştir: Bölüm 2’de, tezin konusuyla ilgili mevcut literatür detaylı bir şekilde incelenecektir. İncelenecek konular arasında veri dengesizliği problemleri ve çözüm yaklaşımları, veri artırma teknikleri, kenar tespiti algoritmaları, metasezgisel optimizasyon algoritmaları ve bunların derin öğrenmede kullanımı, otokodlayıcılar ve tarımsal görüntü işleme uygulamaları bulunmaktadır.

Bölüm 3’te, öncelikle BOA’nın teorik temelleri ve algoritmik özellikleri detaylı bir şekilde açıklanacaktır. Ardından, birinci aşamada geliştirilen BOA-GA yönteminin matematiksel temelleri, kenar tespiti entegrasyonu ve uygulama adımları anlatılacaktır. Sonrasında, ikinci aşamada geliştirilen GBOA ve S-Det sisteminin mimarisi, bileşenleri ve çalışma prensipleri detaylandırılacaktır.

Bölüm 4’te, geliştirilen yöntemlerin deneysel sonuçları sunulacaktır. Öncelikle birinci aşamada geliştirilen BOA-GA yönteminin farklı dengesiz veri setlerindeki performansı ve kenar tespiti entegrasyonunun etkisi raporlanacaktır. Ardından, ikinci aşamada geliştirilen GBOA’nın karşılaştırmalı optimizasyon testleri ve S-Det sisteminin çilek olgunluk tespiti problemi ile diğer görüntü sınıflandırma problemlerindeki performansı detaylı bir şekilde sunulacaktır. Elde edilen sonuçlar, görselleştirmeler ve istatistiksel analizlerle desteklenecektir.

Bölüm 5’te, deneysel sonuçlar kapsamlı bir şekilde yorumlanmış ve tartışılmıştır. Metodolojik yaklaşımların güçlü ve zayıf yönleri, elde edilen başarıların nedenleri, karşılaşılan sınırlamalar ve bunların olası çözüm yolları ele alınmıştır. Ayrıca, çalışmanın literatüre katkıları ve pratik uygulamaları üzerine değerlendirmeler yapılmıştır.

Bölüm 6’da ise, tez çalışmasının ana bulguları özetlenmiş ve genel bir değerlendirme yapılmıştır. Elde edilen sonuçların bilimsel ve pratik anlamları vurgulanmış, çalışmanın sınırlılıkları belirtilmiş ve gelecekteki araştırmalar için öneriler sunulmuştur.

2. KAYNAK ARAŞTIRMASI

Bu bölümde, tez çalışmasının kapsadığı konularla ilgili mevcut literatür verilmiştir. İnceleme, birinci aşamada ele alınan problemlerden başlayarak, ikinci aşamada geliştirilen çözümlere doğru ilerleyecek şekilde organize edilmiştir. Bu yaklaşım, tez çalışmasının kademeli metodolojik ilerlemesini yansıtmakta ve literatürdeki mevcut çalışmaların tez kapsamındaki problemlere nasıl yaklaştığını sistematik bir şekilde ortaya koymaktadır.

Derin öğrenme modellerinin eğitiminde karşılaşılan en önemli problemlerden biri, veri setlerindeki sınıf dengesizliğidir (Buda vd., 2018; Wang vd., 2021). Bu problem, özellikle tıbbi teşhis, endüstriyel kalite kontrol ve tarımsal görüntü analizi gibi alanlarda kritik öneme sahiptir.

Veri dengesizliği probleminin çözümüne yönelik literatürde çeşitli yaklaşımlar önerilmiştir. Örnekleme temelli yöntemler arasında, Chawla vd. (2002) SMOTE algoritmasını önermiş ve azınlık sınıf örnekleri arasındaki yerel komşuluk ilişkilerini kullanarak sentetik örnek üretimini sistematik hale getirmiştir. SMOTE, azınlık sınıf örnekleri arasında enterpolasyon yaparak yeni örnekler üretmeyi amaçlar ve böylece modelin azınlık sınıfı daha iyi temsil etmesine katkı sağlar.

Han vd. (2005) Borderline-SMOTE yöntemini önermiş ve karar sınırına yakın (sınır bölgesindeki) örnekleri önceliklendirerek sentetik örneklerin daha kritik bölgelerde üretilmesini hedeflemiştir. He vd. (2008) ADASYN algoritmasını geliştirerek, öğrenme zorluğuna göre adaptif örnek üretimi yapmış; özellikle zor sınıflandırılan bölgelerde daha fazla sentetik örnek oluşturarak veri dağılımını dinamik biçimde dengelemeyi amaçlamıştır. Barua vd. (2014) MWMOTE yönteminde, azınlık sınıfı içindeki bilgilendirici/zor örnekleri ağırlıklandırarak örnek üretim sürecini bu örnekler etrafında yoğunlaştırmayı önermiştir.

Öğrenme algoritmasına dayalı yöntemlerde, Lin vd. (2017) odak kayıp (focal loss) fonksiyonunu önererek eğitim sırasında zor örneklerle daha fazla ağırlık verilmesini sağlamış ve baskın sınıfın öğrenmeyi domine etmesini azaltmayı hedeflemiştir. Cui vd. (2019) sınıf dengesizliğini kayıp fonksiyonu seviyesinde ele alan yeniden ağırlıklandırma yaklaşımları önermiş ve örnek sayısına bağlı yanlılığı azaltmayı amaçlamıştır. Krawczyk (2016), veri dengesizliği problemiindeki temel zorlukları (ör. sınıf örtüşmesi, gürültü ve maliyet duyarlılığı) tartışarak gelecek araştırma yönelimlerini

özetlemiştir. Haixiang vd. (2017) ise dengesiz veri setlerinde kullanılan makine öğrenmesi stratejilerini (örnekleme, maliyet duyarlı öğrenme, topluluk yöntemleri) kapsamlı biçimde derleyerek yöntemlerin güçlü/zayıf yönlerini karşılaştırmıştır.

Veri dengesizliği probleminin çözümünde, örnekleme temelli yöntemlerin en yaygın uygulaması veri artırmadır. Veri artırma, derin öğrenme modellerinin eğitiminde kullanılan en yaygın performans artırıcı tekniklerden biridir ve modelin genellenebilirliğini artırmayı hedefler.

Shorten ve Khoshgoftaar (2019), veri artırma tekniklerini kapsamlı bir şekilde inceleyerek geometrik (döndürme, kırpma, ölçekleme vb.) ve fotometrik (parlaklık/kontrast değişimi vb.) dönüşümlerin farklı görevlerdeki etkilerini tartışmıştır. Cubuk vd. (2019) AutoAugment ile veri setinden öğrenilen dönüşüm politikaları üzerinden otomatik strateji seçimini önermiş; böylece sabit/elle belirlenen artırma parametrelerinin sınırlılığını azaltmayı amaçlamıştır. Lim vd. (2019) Fast AutoAugment yaklaşımıyla arama sürecini hızlandırarak benzer artırma politikalarının daha düşük hesaplama maliyetiyle elde edilmesini hedeflemiştir.

Zhang vd. (2017) Mixup yönteminde iki örneğin piksel ve etiket düzeyinde karıştırılmasıyla daha yumuşak karar sınırları öğrenmeyi amaçlamış; Yun vd. (2019) CutMix ile görüntü yamalarının kes-yapıştır temelli birleştirilmesi üzerinden daha güçlü düzenleme etkisi elde etmeyi hedeflemiştir. Cubuk vd. (2020) RandAugment ile politika aramasını sadeleştirerek AutoAugment'in arama maliyetini azaltmayı önermiştir. Hataya vd. (2020) ise görüntü içeriğine bağlı olarak artırma parametrelerinin dinamik şekilde ayarlanmasının, artırma sürecinin görev ve veri seti özelliklerine daha iyi uyarlanmasına katkı sağlayabileceğini incelemiştir.

Geleneksel veri artırma yöntemlerinin yanı sıra, üretici modeller kullanılarak da veri artırma gerçekleştirilebilmektedir. Goodfellow vd. (2014) Üretici Çekişmeli Ağ (Generative Adversarial Network, GAN) çerçevesiyle, üretici ve ayırt edici ağların çekişmeli öğrenmesi üzerinden gerçekçi sentetik örnekler üretilebileceğini göstermiştir. Bu yaklaşım, özellikle azınlık sınıflarının örnek sayısının sınırlı olduğu senaryolarda veri çeşitliliğini artırmak için bir alternatif sunmaktadır. Karras vd. (2020) StyleGAN2 ile üretim kalitesini ve kararlılığını artıran mimari ve eğitim iyileştirmeleri önermiş, daha yüksek görsel kaliteye sahip sentetik görüntüler elde edilmesini sağlamıştır.

Veri artırma süreçlerinde, görüntülerin yapısal bilgisinin korunması kritik öneme sahiptir. Bu yapısal bilgiyi korumak ve değerlendirmek için, görüntülerdeki nesnelerin

sınırlarını ortaya çıkaran kenar tespiti gibi yöntemler sıklıkla kullanılmaktadır. Kenar tespiti, görüntülerdeki nesnelerin sınırlarını belirlemek için kullanılan temel tekniklerden biridir. Geleneksel yöntemler arasında, Canny (1986) gürültü azaltma, gradyan hesaplama ve çift eşikli izleme (histerezisli eşikleme) adımlarını birleştiren çok aşamalı bir yaklaşım önermiş ve uzun yıllar referans yöntem olarak kullanılmıştır.

Derin öğrenme tabanlı yaklaşımlarda, Xie ve Tu (2015) Bütüncül-İç İç Kenar Tespiti (Holistically-Nested Edge Detection, HED) yöntemiyle çok ölçekli ara katman çıktılarından kenar haritaları üreterek daha ayrıntılı ve tutarlı sınır tespiti hedeflemiştir. Liu vd. (2017) Zengin Evrişimsel Özellikler (Richer Convolutional Features, RCF) yaklaşımıyla farklı seviyelerdeki evrişimsel öznitelikleri birleştirerek ince ayrıntıların korunmasına odaklanmış; Wang vd. (2018) Kategori Farkındalıklı Anlamsal Kenar Tespiti Ağı (Category-Aware Semantic Edge Detection Network, CASENet) ile sınıftan bağımsız kenar tespitini semantik ipuçlarıyla güçlendirmeyi amaçlamıştır.

Deng ve Liu (2020), derin öğrenme tabanlı kenar tespiti yöntemlerini kapsamlı biçimde derleyerek mimari tercihlerin, kayıp fonksiyonlarının ve veri setlerinin performansa etkisini tartışmıştır. Soria vd. (2020) dikkat mekanizmalarını kenar tespitine entegre ederek, modelin kenar bölgelerine daha seçici odaklanmasını ve gürültülü bölgelerde yanlış tespitlerin azaltılmasını hedeflemiştir.

Kenar tespiti algoritmalarının veri artırma süreçlerine entegrasyonu, görüntülerin yapısal bilgisinin korunması açısından önemlidir. Bu entegrasyon, özellikle geometrik dönüşümler uygulanırken, görüntülerin kritik yapısal özelliklerinin bozulmamasını sağlamaktadır. Bu tür entegrasyon problemlerinin çözümünde, optimizasyon algoritmaları kritik bir rol oynamaktadır. Optimizasyon algoritmaları, veri artırma parametrelerinin seçiminden, kenar korunumunun değerlendirilmesine kadar birçok aşamada kullanılmaktadır.

Optimizasyon problemleri, mühendislik ve bilim alanlarında yaygın olarak karşılaşılan problemlerdir. Geleneksel optimizasyon yöntemleri, türevlenebilir ve dışbükey fonksiyonlar için etkili çözümler sunabilmektedir. Ancak, karmaşık, çok modlu ve yüksek boyutlu optimizasyon problemlerinde, bu yöntemler yetersiz kalmaktadır.

Metasezgisel optimizasyon algoritmaları, doğadaki biyolojik, fiziksel veya sosyal sistemlerden esinlenerek geliştirilmiş stokastik arama algoritmalarıdır (Yang, 2010). Kennedy ve Eberhart (1995) Parçacık Sürü Optimizasyonu (Particle Swarm

Optimization, PSO) algoritmasını önermiş, parçacık sürü davranışını modellemiştir. Holland (1992) Genetik Algoritma (Genetic Algorithm, GA)'nın temellerini atmış, doğal seçim ve genetik operatörleri kullanmıştır. Dorigo ve Stützle (2004) Karınca Kolonisi Optimizasyonu (Ant Colony Optimization, ACO) algoritmasını geliştirmiş, karınca kolonisi davranışını optimize etmiştir. Karaboga ve Basturk (2007) Yapay Arı Kolonisi (Artificial Bee Colony, ABC) algoritmasını önermiş, arı kolonisi davranışını modellemiştir. Mirjalili ve Lewis (2016) BOA'yı önermiş, balina avlanma davranışından esinlenmiştir. Bu çizgide, Yang ve Deb (2009) Guguk Kuşu Araması'nı (Cuckoo Search) önermiş; Mirjalili vd. (2014) ise Gri Kurt Optimizasyon Algoritması'nı (Grey Wolf Optimizer) geliştirerek metasezgisel algoritma ailesini önemli örneklerle zenginleştirmiştir. Bu algoritmaların temel avantajları türev bilgisine ihtiyaç duymamaları ve global optimum arama yetenekleridir, ancak yakınsama hızı ve popülasyon çeşitliliği kaybı gibi sınırlamaları bulunmaktadır (Faris vd., 2018).

BOA, Mirjalili ve Lewis (2016) tarafından önerilmiş, balinaların baloncuk-ağ ile avlanma davranışından esinlenmiştir. BOA, avlanma, baloncuk-ağ saldırısı ve av arama operatörlerini kullanarak keşif-sömürü dengesini sağlar. Standart BOA'nın popülasyon çeşitliliği kaybı sorununu ele almak için, Kaur ve Arora (2018) Kaotik BOA (Chaotic BOA)'yı önermiş, kaotik haritalar kullanarak çeşitliliği artırmıştır. Mafarja ve Mirjalili (2017) İkili BOA (Binary BOA)'yı geliştirmiş, ikili optimizasyon problemlerine uyarlamıştır. Gupta ve Deep (2019) iyileştirilmiş BOA ile yakınsama performansını iyileştirmiştir. Aljarah vd. (2020) BOA'nın çeşitli uygulama alanlarını incelemiş, Abdel-Basset vd. (2021) güncel BOA varyantlarını kapsamlı bir şekilde değerlendirmiştir. Metasezgisel optimizasyon algoritmalarının derin öğrenme alanındaki uygulamalarını anlamak için, öncelikle derin öğrenme modellerinin geleneksel optimizasyon yaklaşımlarını incelemek gerekmektedir. Bu inceleme, metasezgisel algoritmaların derin öğrenme sistemlerine nasıl entegre edilebileceğini ve hangi durumlarda daha etkili olabileceğini anlamak açısından önemlidir.

Geleneksel olarak, derin sinir ağlarının eğitimi, türevlenebilir kayıp fonksiyonları üzerinde gradyan tabanlı optimizasyon yöntemleri ile gerçekleştirilir. Bu sınıfta en temel yaklaşım Stokastik Gradyan İnişi (Stochastic Gradient Descent, SGD) ve onun momentum/Nesterov gibi ivmelendirilmiş varyantlarıdır. Uygulamada, uyarlamalı öğrenme oranı kullanan yöntemler de yaygın biçimde tercih edilmektedir; RMSprop (Tieleman ve Hinton, 2012) kare gradyanların üstel hareketli ortalamasını

izleyerek adım büyüklüğünü parametre bazında ayarlarken, Adam (Kingma ve Ba, 2014) birinci ve ikinci moment tahminlerini birleştirerek daha kararlı ve hızlı yakınsama sağlamayı hedefler. Bu çizgide Adagrad, Adadelta, Nadam, AMSGrad ve AdamW gibi türevler de, özellikle büyük veri setleri ve yüksek boyutlu parametre uzaylarında eğitim sürecinin daha dengeli yürütülmesine katkı sağlayabilmektedir.

Ancak, gradyan tabanlı yöntemlerin temel sınırlamaları, yerel minimumlara saplanma riski, hiperparametre seçimine aşırı bağımlılık ve yakınsama hızının yavaş olmasıdır. Bu sınırlamalar, özellikle küçük veri setleri veya dengesiz veri setlerinde daha belirgin hale gelmektedir (Reddi vd., 2021).

Metasezgisel optimizasyon algoritmalarının derin öğrenme modellerinin eğitiminde kullanımı, yıllardır popüler bir araştırma konusu olmuştur. Erken dönem çalışmalarda, PSO ve GA gibi popülasyon tabanlı yaklaşımlar sinir ağı eğitiminde ağırlık/parametre araması için alternatif bir optimizasyon yolu olarak ele alınmıştır (Whitley, 1994; Settles ve Soule, 2005).

Daha sonraki çalışmalarda, farklı metasezgisel algoritmaların sinir ağı eğitimi ve ayarı üzerindeki etkisi araştırılmıştır. Mirjalili vd. (2014) Gri Kurt Optimizasyon Algoritması yaklaşımını sinir ağı eğitimine uygulamış; Faris vd. (2018) ise BOA'yı derin öğrenme modellerinin optimizasyonunda kullanarak gradyan tabanlı yöntemlere alternatif bir arama mekanizması sunmuştur. Aljarah vd. (2018) PSO ile Evrişimli Sinir Ağı (Convolutional Neural Network, CNN) hiperparametre optimizasyonu gerçekleştirerek model seçimi/ayar sürecini otomatikleştirmeyi hedeflemiş; Elaziz vd. (2020) BOA ile derin sinir ağı ağırlık optimizasyonu yaparak öğrenme sürecinde daha iyi çözümlere ulaşmayı amaçlamıştır. Yıldız vd. (2021) bu çizgideki çalışmaları derleyerek metasezgisel algoritmaların derin öğrenmedeki kullanım alanlarını, avantajlarını ve sınırlılıklarını kapsamlı biçimde incelemiştir.

Bu yaklaşımların temel avantajları, global optimum arama yeteneği ve türev bilgisine ihtiyaç duymamalarıdır; ancak yüksek boyutlu parametre uzaylarında hesaplama karmaşıklığı artmaktadır. Bu nedenle, metasezgisel algoritmalar genellikle küçük veya orta ölçekli derin öğrenme modellerinde, belirli katmanların/alt bileşenlerin optimizasyonunda veya hiperparametre aramasında kullanılmaktadır. Bu bağlamda, girdi verisini daha düşük boyutlu bir temsile sıkıştırıp yeniden üretmeyi öğrenerek öznitelik çıkarımı/boyut azaltma amacıyla kullanılabilen otokodlayıcılar gibi eğitici-siz öğrenme mimarilerinin optimizasyonu, metasezgisel algoritmalar için önemli uygulama

alanlarından birini oluşturmaktadır.

Hinton ve Salakhutdinov (2006), derin otokodlayıcıların boyut azaltma probleminde etkili bir şekilde kullanılabileceğini; doğrusal yöntemlerin yakalayamadığı doğrusal olmayan yapıları daha kompakt bir temsilde koruyabildiğini göstermiştir. Çalışmada, katman-katman ön eğitim ve ardından ince ayar stratejisiyle, yüksek boyutlu girdilerden daha düşük boyutlu özniteliklerin öğrenilebildiği ve bu temsillerin yeniden yapılandırma kalitesini korurken anlamlı bir özet sunduğu raporlanmıştır. Bu sonuçlar, otokodlayıcıların öznitelik çıkarımı/boyut azaltma amacıyla kullanılmasına yönelik literatürdeki erken ve etkili örneklerden biridir.

Vincent vd. (2010) yığılmış gürültü giderici otokodlayıcı (stacked denoising autoencoder) yaklaşımıyla, gürültülü girdilerden daha dayanıklı öznitelikler çıkarmayı hedeflemiş ve yeniden yapılandırma sürecini düzenleştiren bir mekanizma olarak ele almıştır. Kingma ve Welling (2013) Varyasyonel Otokodlayıcı (Variational Autoencoder, VAE) modelini geliştirerek olasılıksal bir gizli özellik uzayı öğrenimi tanımlamış ve bu uzaydan örnekleme yaparak yeni veri üretimini mümkün kılmıştır. Masci vd. (2011) evrimsel otokodlayıcı (convolutional autoencoder) ile özellikle görüntülerde yerel örüntüleri daha iyi temsil eden hiyerarşik öznitelikler öğrenmeyi amaçlamıştır.

Chen vd. (2020) güncel otokodlayıcı mimarilerini derleyerek farklı varyantların hangi senaryolarda tercih edilebileceğini tartışmış; Zhao vd. (2021) ise otokodlayıcıların görüntü analizi uygulamalarındaki kullanımını değerlendirerek pratik kullanım alanlarını özetlemiştir. Otokodlayıcıların optimizasyonu, geleneksel gradyan tabanlı yöntemlerle gerçekleştirilebileceği gibi metasezgisel optimizasyon algoritmaları ile de yapılabilir. Metasezgisel algoritmalar, özellikle hiperparametre optimizasyonu veya belirli katmanların ağırlık parametrelerinin ayarlanmasında etkili olabilmekte; bu tür yaklaşımlar tarımsal görüntü işleme gibi gerçek dünya uygulamalarında öznitelik çıkarımını güçlendirmek için kullanılabilmektedir.

Tarımsal görüntü işleme, tarım sektöründe otomasyon ve karar destek sistemlerinin geliştirilmesinde kritik bir rol oynamaktadır. Kamilaris ve Prenafeta-Boldú (2018), tarımsal uygulamalarda derin öğrenmenin hangi görevlerde (sınıflandırma, tespit, segmentasyon) kullanıldığını sistematik biçimde incelemiş ve veri toplama/etiketleme, değişken çevresel koşullar ve sınırlı veri gibi pratik zorlukları vurgulamıştır.

Meyve tespiti ve sayımı problemlerinde, Sa vd. (2016) DeepFruits ile sahada meyve tespiti için bir sistem geliştirmiş; Koirala vd. (2019) MangoYOLO yaklaşımıyla nesne tespiti tabanlı bir yapı üzerinden mango tespiti ve sayımı gerçekleştirmiştir. Bargoti ve Underwood (2017) meyve sayımı probleminde CNN tabanlı yaklaşımların pratikte uygulanabilirliğini tartışmış; Rahnemoonfar ve Sheppard (2017) ise domates sayımı için derin öğrenme tabanlı bir çözüm önererek değişken görüntü koşullarında sayım performansını artırmayı hedeflemiştir.

Çilek olgunluk tespiti özelinde, Yu vd. (2019) olgunluk aşamalarını sınıflandırmaya odaklanan bir CNN mimarisi önermiş; Zhang vd. (2018) kalite değerlendirmesinde farklı türde öznelilikleri birleştiren hibrit bir yaklaşım geliştirmiştir. Liu vd. (2020) tarımsal görüntü işlemede veri artırma tekniklerinin rolünü derleyerek hangi dönüşümlerin hangi senaryolarda faydalı olabileceğini tartışmış; Fuentes vd. (2021) ise meyve olgunluk tespiti için güncel derin öğrenme yöntemlerinin güçlü ve sınırlı yönlerini karşılaştırmalı olarak değerlendirmiştir. Muresan ve Oltean (2020) tarımsal görüntü sınıflandırmasında CNN ve transfer öğrenme kullanarak sınırlı veri senaryolarında performans artışı elde edilebileceğini göstermiştir.

Bu bölümde özetlenen çalışmalar, veri dengesizliği, veri artırma, kenar tespiti, optimizasyon algoritmaları ve tarımsal görüntü işleme başlıklarında güçlü bir birikim ortaya koymaktadır. Ancak, bu alanların bütünleşik bir şekilde ele alınması ve optimizasyon tabanlı yaklaşımların görüntü analizi sistemlerinde sistematik olarak uygulanması konusunda hala boşluklar bulunmaktadır.

Bu tez çalışması, literatürde tespit edilen boşlukları doldurmak için Giriş bölümünde özetlenen kademeli ve bütünleşik yaklaşımı benimsemektedir; bu yaklaşım literatürdeki mevcut çalışmalardan ayrılmakta ve tezin temel katkısını oluşturmaktadır.

3. MATERYAL VE YÖNTEM

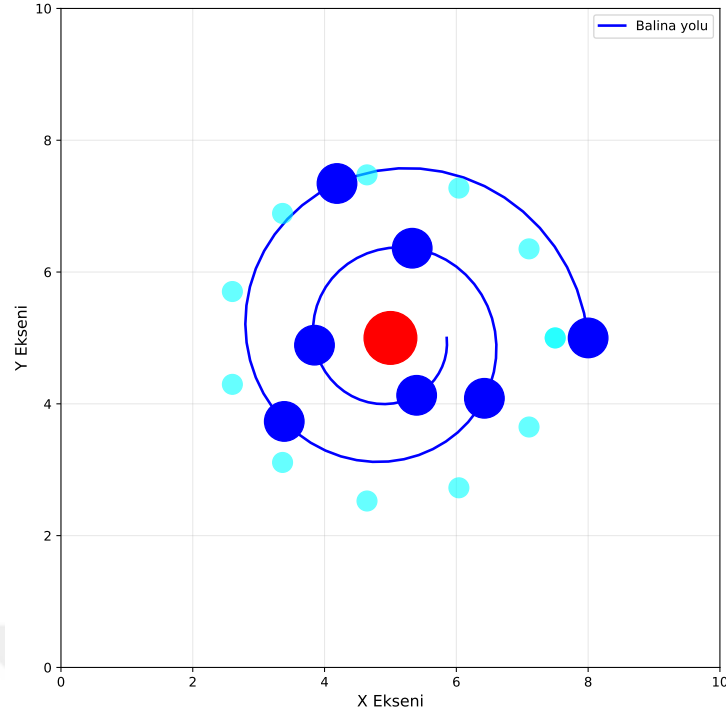
Bu tez çalışmasında, derin öğrenme tabanlı görüntü analizi problemlerinde performansı artırmaya yönelik iki aşamalı bir yaklaşım benimsenmiş ve bu yaklaşım kapsamında iki temel yöntem geliştirilmiştir. Birinci aşamada, sınıf dengesizliği problemini ele almak amacıyla BOA-GA önerilmiş; görüntülerin yapısal bilgisini koruyabilmek için BOA ile adaptif olarak öğrenilen optimum kernel filtresi üzerinden kenar bilgisinden yararlanan bir artırma stratejisi kurgulanmıştır. İkinci aşamada ise, BOA'nın arama dinamiklerini iyileştirmeyi hedefleyen GBOA önerilmiş ve bu algoritmanın hem öznitelik çıkarımı hem de sınıflandırma aşamalarında bütünlük biçimde kullanıldığı S-Det sistemi tasarlanmıştır. Bu bölümde, optimizasyon algoritmalarının teorik temelleri ile önerilen yöntemlerin matematiksel ve algoritmik ayrıntıları açıklanmakta; ayrıca deneysel çalışmalarda kullanılan veri setleri tanıtılmaktadır.

3.1. Balina Optimizasyon Algoritması

Balina Optimizasyon Algoritması, Mirjalili ve Lewis (2016) tarafından önerilmiş bir metasezgisel optimizasyon algoritmasıdır. BOA, doğadaki balinaların, özellikle kambur balinaların avlanma davranışlarından esinlenerek geliştirilmiştir. BOA, balinaların baloncuk-ağ ile avlanma mekanizmasını matematiksel olarak modelleyerek, optimizasyon problemlerinin çözümü için etkili bir arama stratejisi sunmaktadır.

3.1.1. Balina optimizasyon algoritması biyolojik temelleri

Kambur balinalar, avlarını yakalamak için baloncuk-ağ ile avlanma adı verilen özel bir avlanma stratejisi kullanırlar. Bu strateji, balinaların avlarının etrafında spiral bir yol izleyerek hava kabarcıkları oluşturması ve bu kabarcıkların avları yukarı doğru yönlendirmesi prensibine dayanmaktadır (Şekil 3.1). BOA, bu doğal davranışı üç temel operatör ile modellemektedir: avlanma (encircling prey), baloncuk-ağ saldırısı ve av arama (search for prey).



Şekil 3.1. Kambur balinaların baloncuk-ağ ile avlanma davranışı.

3.1.2. Matematiksel formülasyon

BOA, bir optimizasyon problemini çözmek için popülasyon tabanlı bir yaklaşım kullanır. BOA'nın temel bileşenleri ve matematiksel formülasyonları aşağıda detaylandırılmıştır.

3.1.2.1. Avlanma operatörü

Balinalar, avlarının konumunu tahmin ederek etraflarını sararlar. BOA'da, bu davranış Denklem 3.1 ve Denklem 3.2 ile modellenir:

$$\vec{D} = |\vec{C} \cdot X^*(t) - X(t)| \quad (3.1)$$

$$X(t+1) = X^*(t) - \vec{A} \cdot \vec{D} \quad (3.2)$$

Burada, $X^*(t)$ mevcut en iyi çözümü (avın konumunu), $X(t)$ mevcut balina konumunu, t iterasyon sayısını, $\vec{A}$ ve $\vec{C}$ katsayı vektörlerini temsil eder. $\vec{A}$ ve $\vec{C}$ katsayıları sırasıyla Denklem 3.3 ve Denklem 3.4'e göre hesaplanır:

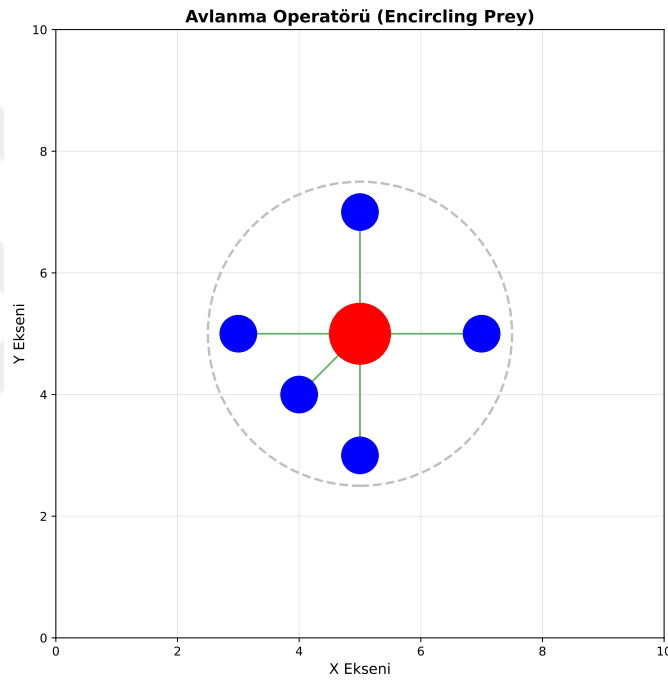
$$\vec{A} = 2\vec{a} \cdot \vec{r} - \vec{a} \quad (3.3)$$

$$\vec{C} = 2 \cdot \vec{r} \quad (3.4)$$

Burada, $\vec{a}$ iterasyonlar boyunca 2'den 0'a doğru lineer olarak azalan bir katsayıdır ve $\vec{r}$ [0,1] aralığında rastgele bir vektördür. $\vec{a}$ katsayısı, algoritmanın keşif ve sömürü dengesini kontrol eder ve Denklem 3.5'e göre güncellenir:

$$\vec{a} = 2 - \frac{2t}{T_{max}} \quad (3.5)$$

Burada, T_{max} maksimum iterasyon sayısını temsil eder. Avlanma operatörünün görsel temsili Şekil 3.2'de gösterilmektedir.



Şekil 3.2. Avlanma operatörü.

3.1.2.2. Baloncuk-ağ saldırısı

Baloncuk-ağ saldırısı, balinaların spiral bir yol izleyerek avlarını yakalaması davranışını modellemektedir. BOA'da, bu davranış iki mekanizma ile gerçekleştirilir: kuşatmayı daraltma mekanizması (shrinking encircling mechanism) ve spiral konum güncelleme mekanizması (spiral updating position mechanism). Kuşatmayı daraltma mekanizması, $\vec{A}$ katsayısının değerinin azalması ile gerçekleşir. $\vec{A}$ değeri [-1,1] aralığında değiştiğinde, balina mevcut en iyi çözümün etrafında rastgele bir konumda bulunabilir. Spiral konum güncelleme mekanizması, balina ile av arasındaki spiral hareketi modellemek için kullanılır ve Denklem 3.6'ya göre hesaplanır:

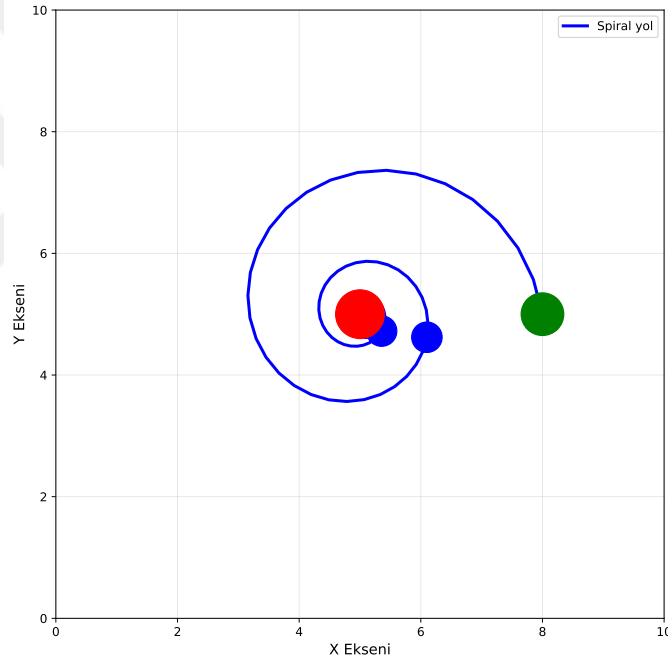
$$X(\vec{t} + 1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t) \quad (3.6)$$

Burada, $\vec{D}' = |X^*(t) - X(\vec{t})|$ balina ile av arasındaki mesafeyi, b logaritmik spiral şeklini tanımlayan sabit bir katsayıyı, l $[-1,1]$ aralığında rastgele bir sayıyı temsil eder.

BOA, kuşatmayı daraltma ve spiral konum güncelleme mekanizmalarını eşit olasılıkla Denklem 3.7'de gösterildiği şekilde kullanır:

$$X(\vec{t} + 1) = \begin{cases} X^*(t) - \vec{A} \cdot \vec{D} & \text{eğer } p < 0,5 \\ \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t) & \text{eğer } p \geq 0,5 \end{cases} \quad (3.7)$$

Burada, p $[0,1]$ aralığında rastgele bir sayıdır. Spiral hareket mekanizması Şekil 3.3'te gösterilmektedir.



Şekil 3.3. Spiral hareket mekanizması.

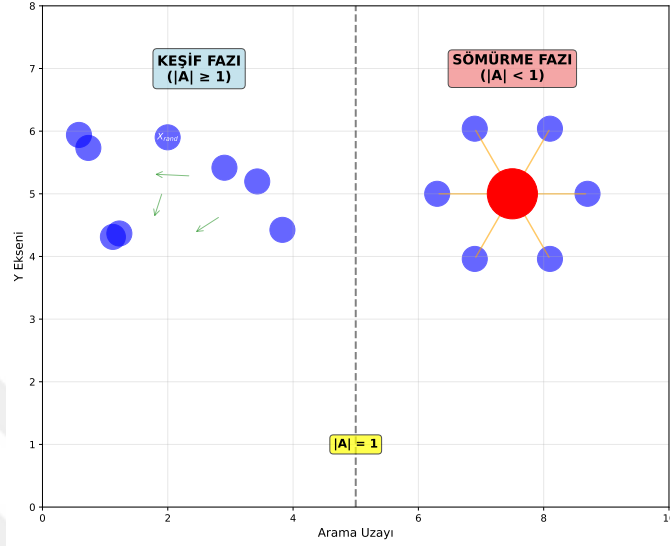
3.1.2.3. Av arama

Av arama mekanizması, algoritmanın keşif fazını temsil eder. Bu mekanizmada, balinalar rastgele seçilen bir balina konumunu referans alarak arama yaparlar. Bu davranış, $|\vec{A}| \geq 1$ olduğunda Denklem 3.8 ve Denklem 3.9 ile ifade edildiği şekilde gerçekleşir:

$$\vec{D} = |\vec{C} \cdot X_{rand} - X(\vec{t})| \quad (3.8)$$

$$X(t+1) = X_{rand} - \vec{A} \cdot \vec{D} \quad (3.9)$$

Burada, X_{rand} mevcut popülasyondan rastgele seçilen bir balina konumunu temsil eder. Keşif ve sömürü fazları Şekil 3.4'te gösterilmektedir.



Şekil 3.4. Keşif ve sömürü fazları.

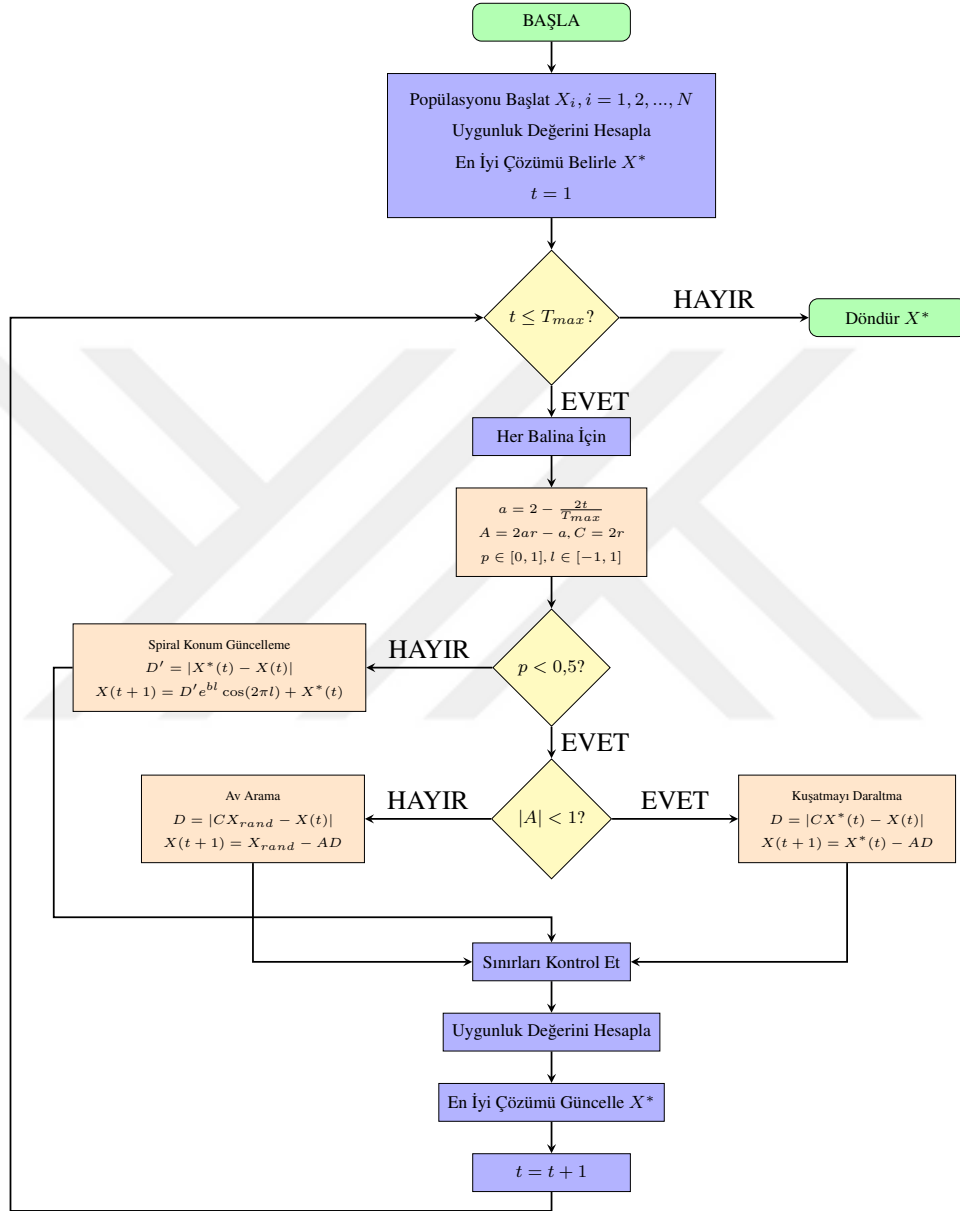
3.1.3. Algoritmanın sözde kodu

BOA'nın genel yapısı Şekil 3.5'te gösterilen akış diyagramı ile özetlenebilir. Algoritmanın detaylı sözde kodu aşağıdaki gibidir:

1. Popülasyonu başlat ($X_i, i = 1, 2, \dots, N$)
2. Her bir çözümün uygunluk değerini hesapla
3. En iyi çözümü belirle (X^*)
4. $t = 1$ 'den T_{max} 'a kadar:
 - (a) Her bir balina için:
 - i. a, A, C, l , ve p parametrelerini güncelle
 - ii. Eğer $p < 0,5$:
 - A. Eğer $|A| < 1$: Kuşatmayı daraltma mekanizması uygula
 - B. Eğer $|A| \geq 1$: Av arama mekanizması uygula
 - iii. Eğer $p \geq 0,5$: Spiral konum güncelleme mekanizması uygula

- (b) Sınırları kontrol et ve ihlal eden çözümleri düzelt
- (c) Her bir çözümün uygunluk değerini hesapla
- (d) En iyi çözümü güncelle (X^*)

5. En iyi çözümü döndür



Şekil 3.5. BOA akış diyagramı.

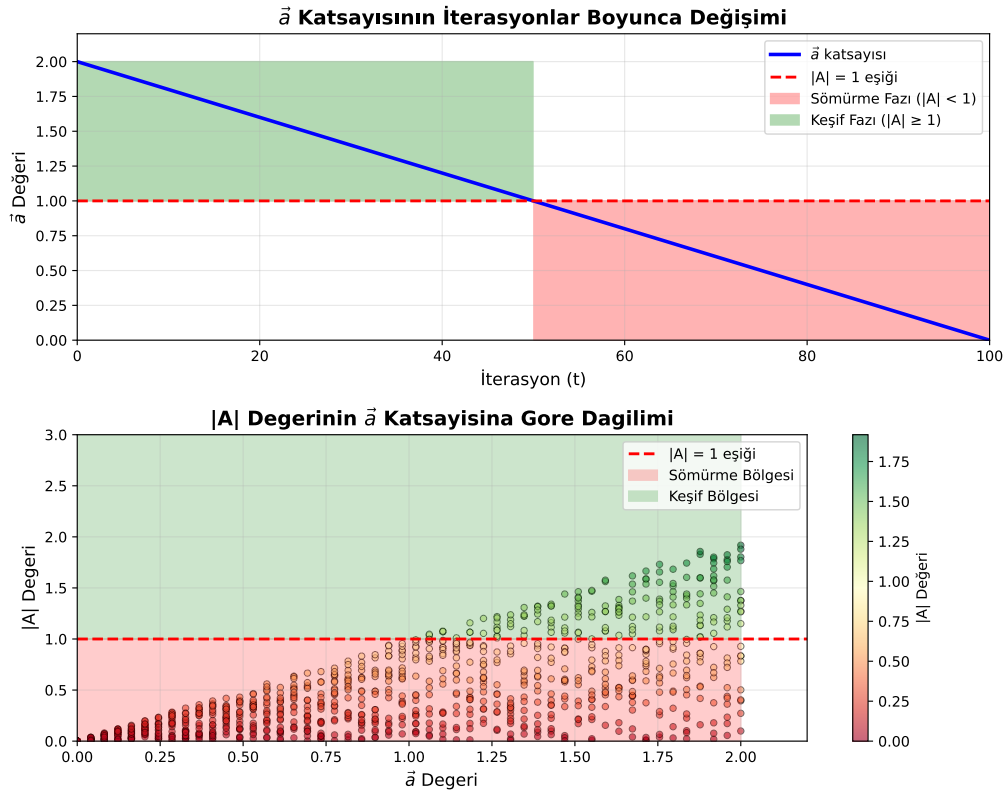
3.1.4. Algoritmanın özellikleri

BOA'nın temel özellikleri aşağıdaki gibi özetlenebilir:

Parametre Sayısı: BOA, diğer metasezgisel algoritmalara kıyasla daha az sayıda parametre gerektirir. Algoritmanın temel parametreleri, popülasyon boyutu (N),

maksimum iterasyon sayısı (T_{max}) ve logaritmik spiral katsayısı (b) olmak üzere sadece üç tanedir. Bu özellik, algoritmanın kullanımını kolaylaştırmakta ve parametre ayarlama gereksinimini minimize etmektedir.

Keşif-Sömürü Dengesi: BOA, $\vec{a}$ katsayısının iterasyonlar boyunca lineer olarak azalması sayesinde, başlangıçta keşif fazına, sonraki iterasyonlarda ise sömürü fazına geçiş yapmaktadır. Bu dinamik geçiş, algoritmanın hem global hem de yerel arama yeteneklerini dengeli bir şekilde kullanmasını sağlamaktadır. $\vec{a}$ katsayısının değişimi ve keşif-sömürü dengesi Şekil 3.6’da gösterilmektedir.



Şekil 3.6. $\vec{a}$ katsayısının değişimi ve keşif-sömürü dengesi.

Yakınsama Özellikleri: BOA, baloncuk-ağ saldırısı mekanizması sayesinde, mevcut en iyi çözümün etrafında yoğun bir arama gerçekleştirerek hızlı yakınsama sağlamaktadır. Ancak, bu özellik aynı zamanda algoritmanın yerel minimumlara saplanma riskini de artırmaktadır.

Hesaplama Karmaşıklığı: BOA'nın zaman karmaşıklığı $O(N \cdot T_{max} \cdot D)$ şeklindedir, burada N popülasyon boyutunu, T_{max} maksimum iterasyon sayısını ve D problem boyutunu temsil eder. Bu karmaşıklık, diğer metasezgisel algoritmalarla karşılaştırıldığında rekabetçi bir performans sunmaktadır.

3.1.5. Algoritmanın avantajları ve sınırlamaları

BOA'nın temel avantajları şunlardır (Mirjalili ve Lewis, 2016):

- **Basit Yapı:** Algoritmanın matematiksel formülasyonu basit ve anlaşılır bir yapıya sahiptir.
- **Az Parametre Gereksinimi:** Algoritma, sadece üç temel parametre ile çalışabilmektedir.
- **Etkili Performans:** Çeşitli optimizasyon problemlerinde etkili sonuçlar vermektedir.
- **Esnek Uygulanabilirlik:** Sürekli, ayrık ve karma optimizasyon problemlerine uyarlanabilmektedir.

BOA'nın temel sınırlamaları ise şunlardır:

- **Popülasyon Çeşitliliği Kaybı:** İterasyonlar ilerledikçe, popülasyon çeşitliliği azalmakta ve algoritma yerel minimumlara saplanma riski artmaktadır (Kaur ve Arora, 2018; Aljarah vd., 2020).
- **Yavaş Yakınsama:** Bazı problemlerde, algoritmanın yakınsama hızı yavaş olabilmektedir (Gupta ve Deep, 2019; Abdel-Basset vd., 2021).
- **Parametre Duyarlılığı:** Logaritmik spiral katsayısı (b) gibi bazı parametreler, problemden probleme değişiklik gösterebilmektedir (Mirjalili ve Lewis, 2016; Aljarah vd., 2020).
- **Yüksek Boyutlu Problemler:** Yüksek boyutlu optimizasyon problemlerinde, algoritmanın performansı düşebilmektedir (Abdel-Basset vd., 2021).

Bu sınırlamalar, literatürde çeşitli BOA varyantlarının geliştirilmesine yol açmıştır.

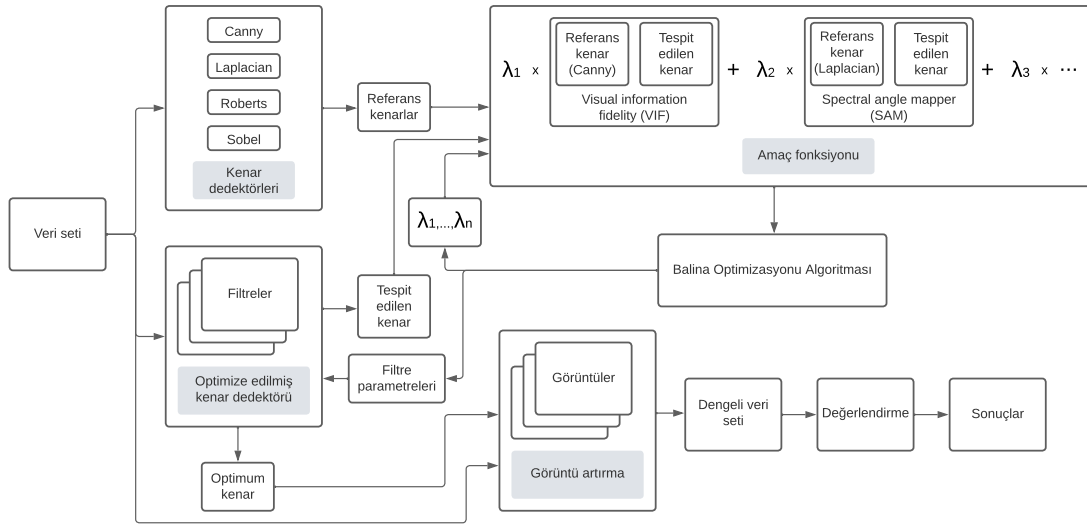
3.2. Balina Optimizasyon Algoritması Tabanlı Görüntü Artırımı Yöntemi

Bu bölümde, birinci aşamada geliştirilen BOA-GA yönteminin matematiksel temelleri, kenar tespiti entegrasyonu ve uygulama adımları detaylı bir şekilde açıklanmaktadır. BOA-GA yöntemi, dengesiz görüntü veri setlerindeki sınıf

dengesizliği problemini çözmek için optimizasyon tabanlı bir görüntü artırımı yaklaşımı sunmaktadır.

3.2.1. Balina optimizasyon algoritması tabanlı görüntü artırımı yönteminin genel mimarisi

BOA-GA yöntemi, iki temel aşamadan oluşmaktadır: (i) çok katmanlı filtre yapısı kullanılarak optimize edilmiş kenar tespiti ve (ii) dengesiz veri setleri için kenar tabanlı görüntü artırımı. Yöntemin genel mimarisi Şekil 3.7’de gösterilmektedir.



Şekil 3.7. BOA-GA yönteminin genel mimarisi.

Şekil 3.7’de görüldüğü gibi, çok katmanlı kenar tespitçisinin parametreleri ve amaç fonksiyonundaki benzerlik metriklerinin ağırlık katsayıları, BOA kullanılarak optimize edilmektedir. Optimizasyon süreci, tespit edilen kenar görüntüleri ile kenar tespiti yöntemlerinden elde edilen referans kenar görüntüleri arasındaki benzerliği maksimize etmeyi amaçlamaktadır. Optimal parametreler belirlendikten sonra, elde edilen kenar görüntüleri açısıl yerleştirme tabanlı bir görüntü artırma stratejisi kullanılarak azınlık sınıfları için sentetik örnekler üretmek amacıyla kullanılmaktadır.

3.2.2. Arama ajanı temsili

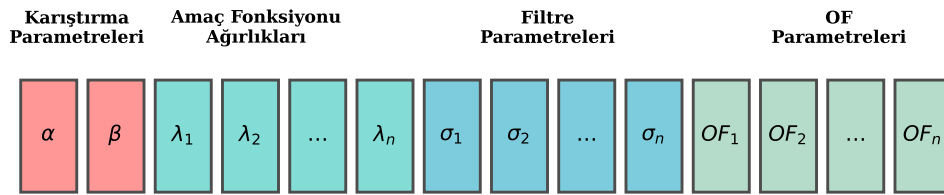
BOA-GA yönteminde, her bir arama ajanı optimize edilecek tüm parametreleri temsil eden tek boyutlu bir vektör olarak kodlanmaktadır. Arama ajanı vektörü dört ana bileşenden oluşmaktadır:

1. **Karıştırma Parametreleri** (α, β): Filtrelenmiş görüntülerin entegrasyonunu kontrol eden parametreler.
2. **Amaç Fonksiyonu Ağırlıkları** ($\lambda_1, \lambda_2, \dots, \lambda_8$): Benzerlik metriklerinin katkısını belirleyen ağırlık katsayıları.
3. **Çok Katmanlı Filtre Parametreleri** ($\sigma_1, \sigma_2, \dots, \sigma_n$): Gaussian filtresi parametreleri dahil olmak üzere filtre parametreleri.
4. **Optimum Kernel Filtresi (OF) Parametreleri**: Optimum kernel filtresinin genişlik ve yükseklik parametreleri.

Arama ajanı vektörünün toplam uzunluğu Denklem 3.10'a göre belirlenmektedir:

$$V_{length} = Numberof(\lambda) + Numberof(\sigma) + OF_{width} \times OF_{height} + 2 \quad (3.10)$$

Arama ajanı vektörünün yapısı Şekil 3.8'de gösterilmektedir.



Şekil 3.8. Arama ajanı vektörünün yapısı.

Bu temsil yapısı, BOA'nın kenar tespiti ve artırma çerçevesinin tüm bileşenlerini aynı anda optimize etmesine olanak sağlamaktadır.

3.2.3. Amaç fonksiyonu

Amaç fonksiyonu, optimizasyon sürecini yönlendirmede kritik bir rol oynamaktadır. Bu çalışmada, tespit edilen kenar görüntülerinin kalitesini değerlendirmek için benzerlik tabanlı metrikler kullanılmaktadır. Özellikle, Görsel Bilgi Doğruluğu (Visual Information Fidelity, VIF) ve Spektral Açılı Haritalayıcı (Spectral Angle Mapper, SAM) metrikleri, tespit edilen kenar görüntüsü ile kenar tespiti yöntemlerinden elde edilen referans kenar görüntüleri arasındaki benzerliği ölçmek için kullanılmaktadır.

VIF metriği, iki görüntü arasındaki paylaşılan görsel bilgi miktarını ölçmektedir. SAM metriği ise, görüntü vektörleri arasındaki açısal mesafeyi hesaplayarak spektral benzerliği değerlendirmektedir. Bu tamamlayıcı metrikler, kenar kalitesinin sağlam bir değerlendirmesini sağlamaktadır. VIF ve SAM metrikleri sırasıyla Denklem 3.11 ve Denklem 3.12'e göre hesaplanmaktadır:

$$VIF = \frac{I(C, F|S)}{I(C, E|S)} \quad (3.11)$$

$$SAM = \cos^{-1} \left(\frac{\sum XY}{\sqrt{\sum X^2 \sum Y^2}} \right) \quad (3.12)$$

burada C referans görüntüyü, F tespit edilen kenar görüntüsünü, E referans kenar görüntüsünü, S görsel bilgiyi, X ve Y ise görüntü vektörlerini temsil etmektedir. Amaç fonksiyonu, VIF ve SAM benzerlik skorlarının ağırlıklı bir kombinasyonu olarak tanımlanmaktadır:

$$Fitness = \sum_{n=1}^4 \lambda_n \times VIF(r, t) + \sum_{n=5}^8 \lambda_n \times SAM(r, t) \quad (3.13)$$

burada r referans kenar görüntüsünü, t tespit edilen kenar görüntüsünü, λ_n ise benzerlik metriklerinin ağırlık katsayılarını temsil etmektedir. Denklem 3.13'te tanımlanan amaç fonksiyonu, VIF ve SAM skorlarının ağırlıklı toplamından oluşmaktadır. Benzerlik metriklerinin ağırlıkları da BOA tarafından optimize edilmektedir, bu da algoritmanın etkilerini adaptif olarak dengelemesine olanak sağlamaktadır. Optimizasyon süreci, amaç fonksiyonu değerini maksimize etmeyi amaçlamakta, böylece referans kenarlara yakın kenar görüntüleri üretilmektedir.

3.2.4. Kenar tespiti entegrasyonu

BOA-GA yönteminde, kenar tespiti çok katmanlı bir filtre yapısı kullanılarak gerçekleştirilmektedir. Bu yapı, Gaussian filtreleri ve OF içermektedir. Filtre parametreleri, BOA tarafından optimize edilmekte ve referans kenar görüntüleri ile maksimum benzerlik sağlayacak şekilde ayarlanmaktadır.

3.2.4.1. Çok katmanlı filtre yapısı

Çok katmanlı filtre yapısı, görüntülerdeki kenarları tespit etmek için hiyerarşik bir yaklaşım sunmaktadır. Bu yapı, farklı ölçeklerdeki kenar bilgisini yakalamak için

birden fazla filtre katmanından oluşmaktadır. Her katman, farklı standart sapma değerlerine $(\sigma_1, \sigma_2, \dots, \sigma_n)$ sahip Gaussian filtreleri kullanmaktadır.

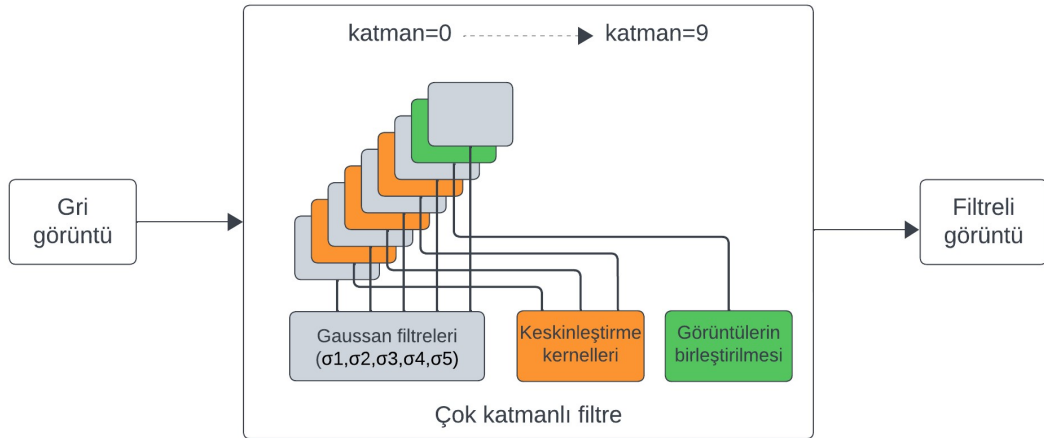
Gaussian filtreleri, görüntüdeki gürültüyü azaltmak ve kenar tespiti için uygun bir temel sağlamak amacıyla kullanılmaktadır. İki boyutlu Gaussian filtresi, aşağıdaki denklem ile tanımlanmaktadır:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (3.14)$$

Denklem 3.14'te gösterildiği gibi, Gaussian filtresi iki boyutlu bir dağılım fonksiyonu kullanarak görüntüyü yumuşatmaktadır. Burada σ standart sapma parametresini, (x, y) piksel koordinatlarını temsil etmektedir. Farklı σ değerleri, farklı ölçeklerdeki kenar bilgisini yakalamaya olanak sağlamaktadır. Küçük σ değerleri, ince detayları ve keskin kenarları vurgularken, büyük σ değerleri daha geniş alanları ve yumuşak geçişleri tespit etmektedir.

Çok katmanlı yapıda, her katmanın çıktısı bir sonraki katmana girdi olarak verilmekte ve bu şekilde hiyerarşik bir kenar tespiti gerçekleştirilmektedir. Bu yaklaşım, hem yerel hem de global kenar bilgisini koruyarak daha kapsamlı bir kenar haritası üretmektedir.

Çok katmanlı filtre yapısının genel mimarisi Şekil 3.9'da gösterilmektedir.



Şekil 3.9. Çok katmanlı filtre yapısının mimarisi.

3.2.4.2. Optimum kernel filtresi

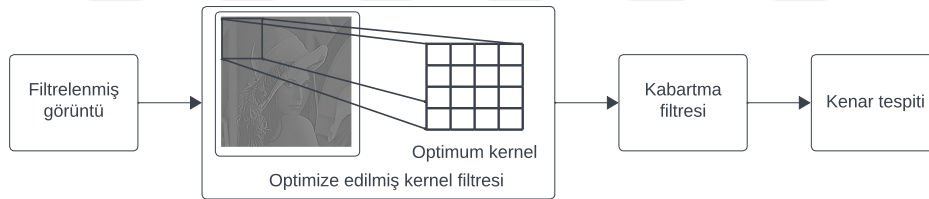
Optimum Kernel Filtresi, kenar tespiti için özel olarak tasarlanmış ve BOA tarafından optimize edilen bir filtre yapısıdır. OF, sabit boyutlu bir kernel matrisi olarak temsil edilmekte ve her bir elemanı optimizasyon sürecinde belirlenmektedir. OF'nin

boyutu, $OF_{width} \times OF_{height}$ olarak tanımlanmakta ve bu boyutlar da optimizasyon sürecinin bir parçasını oluşturmaktadır. Filtre elemanları, arama ajanı vektörünün bir bileşeni olarak kodlanmakta ve BOA tarafından optimize edilmektedir. OF'nin uygulanması, Denklem 3.15'te verilen konvolüsyon işlemi ile gerçekleştirilmektedir:

$$I_{filtered}(x, y) = \sum_{i=0}^{OF_{width}-1} \sum_{j=0}^{OF_{height}-1} OF(i, j) \times I(x+i-\lfloor OF_{width}/2 \rfloor, y+j-\lfloor OF_{height}/2 \rfloor) \quad (3.15)$$

Denklem 3.15'te tanımlanan konvolüsyon işlemi, optimum kernel filtresinin görüntü üzerinde kaydırılması ve her konumda çarpımların toplanması prensibine dayanmaktadır. Burada I orijinal görüntüyü, $I_{filtered}$ filtrelenmiş görüntüyü, $OF(i, j)$ optimum kernel filtresinin (i, j) konumundaki elemanını temsil etmektedir.

OF'nin avantajı, görüntü içeriğine ve referans kenar görüntülerine göre adaptif olarak öğrenilebilmesidir. Bu sayede, kenar tespiti yöntemlerinin sabit kernel yapılarından farklı olarak, her veri seti için özelleştirilmiş bir kenar tespiti gerçekleştirilebilmektedir. OF'nin yapısı ve uygulanması Şekil 3.10'da gösterilmektedir.



Şekil 3.10. OF yapısı ve uygulanması.

3.2.4.3. Filtre entegrasyonu ve kenar tespiti

Çok katmanlı filtre yapısı ve OF'nin entegrasyonu, aşağıdaki adımlarla gerçekleştirilmektedir:

1. Orijinal görüntü, farklı σ değerlerine sahip Gaussian filtreleri ile yumuşatılmaktadır. Her Gaussian filtresi çıktısı, bir filtre katmanını oluşturmaktadır.
2. Her filtre katmanına, OF uygulanmaktadır. Bu işlem, her katmandaki kenar bilgisini güçlendirmektedir.

3. Filtrelenmiş görüntüler, karıştırma parametreleri (α , β) kullanılarak birleştirilmektedir. Bu parametreler, farklı katmanların katkısını kontrol etmektedir:

$$I_{edge} = \alpha \times \sum_{i=1}^n \lambda_i \times G_{\sigma_i}(I) + \beta \times OF(I) \quad (3.16)$$

Denklem 3.16'da gösterildiği gibi, nihai kenar görüntüsü, Gaussian filtrelerinin ağırlıklı toplamı ile optimum kernel filtresinin çıktısının birleştirilmesiyle elde edilmektedir. Burada $G_{\sigma_i}(I)$ σ_i standart sapmasına sahip Gaussian filtresi ile filtrelenmiş görüntüyü, $OF(I)$ optimum kernel filtresi ile filtrelenmiş görüntüyü, λ_i katman ağırlıklarını temsil etmektedir.

4. Nihai kenar görüntüsü, eşikleme ve morfolojik işlemler ile iyileştirilmektedir.

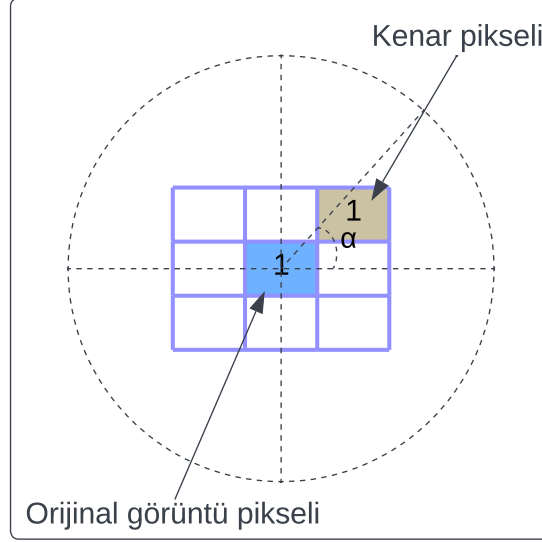
Referans kenar görüntüleri, kenar tespiti yöntemlerinin (Canny, Laplacian, Roberts, Sobel) çıktılarının ortalaması alınarak elde edilmektedir. Bu yaklaşım, farklı kenar tespiti yöntemlerinin güçlü yönlerini birleştirerek daha dengeli ve güvenilir referans kenar görüntüleri üretmektedir. Optimizasyon süreci, tespit edilen kenar görüntülerinin bu referans görüntülerle maksimum benzerliğe sahip olmasını sağlamaktadır.

3.2.5. Kenar tabanlı görüntü artırma

Biyolojik ve tıbbi görüntüler içeren uygulamalarda, sınırlı veri bulunabilirliği nedeniyle dengesiz veri setleri yaygın olarak karşılaşılan bir durumdur. Bu problemi ele almak için, BOA-GA yöntemi kenar tabanlı bir görüntü artırımı stratejisi içermektedir. Bu yaklaşımda, sentetik örnekler optimize edilmiş kenar görüntüsünün karşılık gelen orijinal görüntü ile açısal yerleştirme kullanılarak birleştirilmesiyle üretilmektedir. Artırma süreci, Şekil 3.11'de gösterildiği gibi, kenar görüntüsünün rastgele seçilen bir açı ile döndürülmesi ve önceden tanımlanmış bir katsayı aralığı kullanılarak orijinal görüntü ile karıştırılması şeklinde matematiksel olarak tanımlanmaktadır. Açısal yerleştirme işlemi Denklem 3.17'de ifade edilmektedir:

$$Edge(x, y) = \begin{cases} x = \frac{G_{width}}{2} + \delta \times \cos(\alpha) \\ y = \frac{G_{height}}{2} + \delta \times \sin(\alpha) \end{cases} \quad (3.17)$$

Burada G_{width} ve G_{height} görüntünün genişlik ve yüksekliğini, δ yerleştirme mesafesini, α ise döndürme açısını temsil etmektedir.



Şekil 3.11. Kenar görüntüsünün sentetik görüntü üzerine açısal yerleştirilmesi.

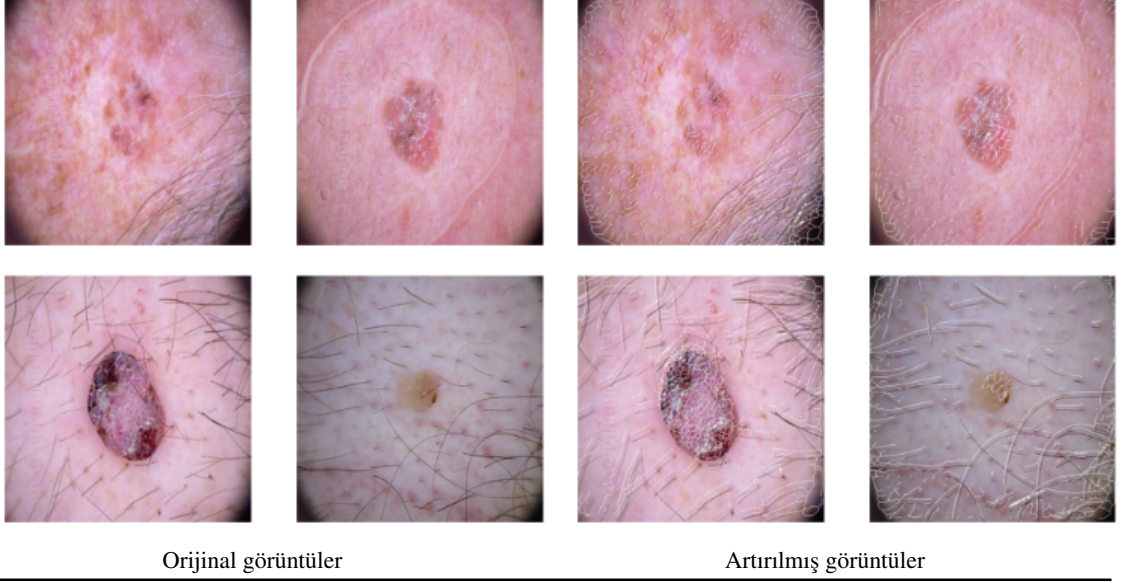
Her bir azınlık sınıfı için üretilecek sentetik örnek sayısı, veri setinin dengesizlik oranına göre belirlenmektedir. Yüzde değişimi Denklem 3.18'e göre, üretilecek örnek sayısı ise Denklem 3.19'e göre hesaplanmaktadır:

$$P_v = \left(\frac{M_j}{M_i} - 1 \right) \times 100 \quad (3.18)$$

$$N = \frac{P_v \times M_i}{100} \quad (3.19)$$

Burada M_j çoğunluk sınıfındaki örnek sayısını, M_i azınlık sınıfındaki örnek sayısını, P_v yüzde değişimini ve N üretilecek sentetik örnek sayısını temsil etmektedir. Bu adaptif aşırı örnekleme stratejisi, gereğinden fazla çoğaltmayı önlerken dengeli bir sınıf dağılımı elde etmektedir.

BOA-GA yöntemi ile üretilen orijinal ve artırılmış görüntü örnekleri Şekil 3.12'de gösterilmektedir.



Şekil 3.12. Orijinal ve BOA-GA ile artırılmış görüntü örnekleri.

3.2.6. Balina optimizasyon algoritması tabanlı görüntü artırma yönteminin özellikleri

BOA-GA, geleneksel veri artırma yöntemlerinden birkaç önemli özellik ile ayrılmaktadır:

- **Optimizasyon Tabanlı Yaklaşım:** Yöntem, sabit parametreler yerine görüntü içeriğine göre dinamik olarak optimize edilen parametreler kullanmaktadır. Bu yaklaşım, her görüntü için en uygun kenar tespiti parametrelerinin belirlenmesini sağlamaktadır.
- **Yapısal Bilgi Korunumu:** Kenar tabanlı artırma stratejisi, görüntülerin kritik yapısal özelliklerini koruyarak sınıf-spesifik özelliklerin bozulmamasını garanti etmektedir.
- **Adaptif Örnek Üretimi:** BOA-GA, veri setinin dengesizlik oranına göre adaptif olarak sentetik örnek sayısını belirlemektedir. Bu yaklaşım, aşırı çoğaltmayı önlerken dengeli bir sınıf dağılımı elde etmektedir.
- **Çok Metrik Değerlendirme:** VIF ve SAM gibi tamamlayıcı metriklerin kullanılması, kenar kalitesinin daha sağlam bir şekilde değerlendirilmesini sağlamaktadır.

- **Referans Tabanlı Optimizasyon:** Standart kenar tespiti yöntemlerinin birleşiminden elde edilen referans kenar görüntüleri, optimizasyon sürecini yönlendirerek daha güvenilir sonuçlar üretmektedir.

3.3. Geliştirilmiş Balina Optimizasyon Algoritması

BOA'nın popülasyon çeşitliliği kaybı sorununu ele almak için, bu tez çalışmasında GBOA önerilmiştir. GBOA, algoritmanın keşif yeteneğini koruyarak daha etkili optimizasyon performansı sağlamaktadır.

3.3.1. Gboa'nın motivasyonu ve temel prensibi

Metasezgisel optimizasyon algoritmaları, temel olarak iki modda çalışır: keşif ve sömürü. Standart BOA'da, sömürü modunda algoritma başlangıç iterasyonlarında hızlı bir şekilde uygunluk fonksiyonunu günceller ve çözümü arama uzayındaki umut verici bölgeye yaklaştırmaya çalışır. Ancak, sonraki iterasyonlarda popülasyon çeşitliliği azaldığında, algoritmanın umut verici bölgeyi yanlış tahmin etme olasılığı bulunmaktadır. Bu sorunu çözmek için, GBOA'da popülasyon çeşitliliği izlenmekte ve çeşitlilik eşik değerinin altına düştüğünde, geliştirilmiş sömürü modu ajanları daha umut verici bir bölgeye yönlendirmektedir.

3.3.2. Popülasyon çeşitliliği izleme

GBOA'da, popülasyondaki ajanların çeşitliliği, kosinüs benzerliği (cosine similarity) metriği kullanılarak hesaplanmaktadır. Kosinüs benzerliği, iki vektör arasındaki açısal mesafeyi ölçen bir metrik olup, aşağıdaki denkleme göre hesaplanmaktadır:

$$\text{cosine}(\vec{M}, \vec{N}) = \frac{\vec{M} \cdot \vec{N}}{\|\vec{M}\| \times \|\vec{N}\|} \quad (3.20)$$

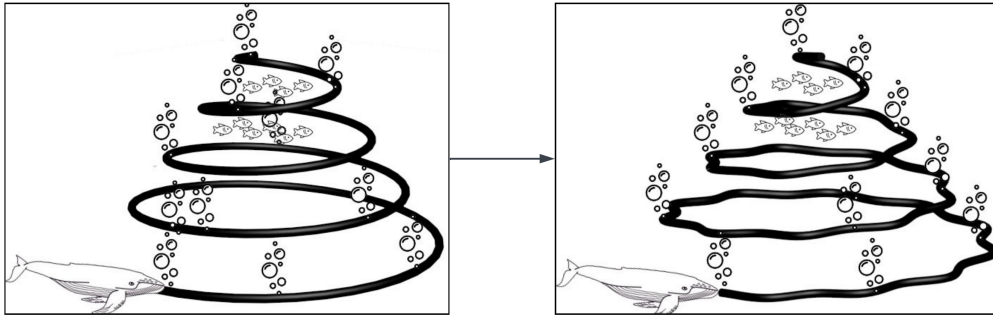
Denklem 3.20'de tanımlanan kosinüs benzerliği, iki vektör arasındaki açısal benzerliği ölçmektedir. Burada, $\vec{M}$ en iyi ajanın vektörünü, $\vec{N}$ ise popülasyondaki ilgili ajanın vektörünü temsil etmektedir. Kosinüs benzerliği değeri $[-1, 1]$ aralığında değişmekte olup, 1'e yakın değerler iki vektörün birbirine çok benzer olduğunu, -1'e yakın değerler ise birbirinden çok farklı olduğunu göstermektedir.

3.3.3. Geliştirilmiş spiral konum güncelleme mekanizması

İlgili ajanın kosinüs benzerliği değeri, benzerlik katsayısı $\beta = 0.9$ 'un üzerinde olduğunda, GBOA ajanları daha umut verici bir bölgeye yönlendirmek için Geliştirilmiş Spiral Konum Güncelleme Mekanizması (GSKGM) kullanmaktadır. GSKGM, aşağıdaki denklem ile tanımlanmaktadır:

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) \cdot (\sin(t \cdot \vec{a}) + \vec{A}) + \vec{X}^*(t) \quad (3.21)$$

Burada kullanılan $\vec{D}'$, b , l , t , $\vec{a}$ ve $\vec{A}$ sembolleri, standart BOA'daki spiral konum güncelleme mekanizmasında daha önce verilmiştir. Denklem 3.21'de, $\sin(t \cdot \vec{a}) + \vec{A}$ terimi, orijinal spiral konum güncelleme mekanizmasına eklenmiştir. Bu terim, t , $\vec{a}$ ve $\vec{A}$ parametrelerine bağlı olarak sinüzoidal bir çizgi boyunca ilerlemeyi sağlamaktadır. GBOA'da, ajanlar küçülen spiral yol üzerinde asimetrik sinüzoidal bir yörünge boyunca hareket etmektedir. Bu yaklaşım, ajanların arama uzayını etkin bir şekilde keşfetmesini ve hızlı bir şekilde optimal çözümlere ulaşmasını sağlamaktadır.



Şekil 3.13. BOA ve GBOA'nın karşılaştırması.

Şekil 3.13'de görüldüğü üzere, önerilen yönlendirilmiş spiral güncelleme mekanizması, klasik BOA'daki logaritmik spiral yapıyı korumakta; ancak popülasyon çeşitliliğine bağlı olarak spiral merkezini daha umut verici bölgelere doğru uyarlamaktadır. BOA'da, ajanlar sabit bir spiral yörünge boyunca hareket ederken, GBOA'da sinüzoidal bir modülasyon eklenerek arama davranışı zenginleştirilmektedir. Bu sayede, GBOA hem yerel optima çevresinde daha hassas bir arama yapabilmekte hem de geniş arama uzayında daha etkin bir keşif gerçekleştirebilmektedir. Geliştirilmiş mekanizma, özellikle kosinüs benzerliği yüksek olan ajanlar için spiral merkezini dinamik olarak ayarlayarak, algoritmanın yakınsama hızını artırmakta ve çözüm kalitesini iyileştirmektedir.

3.3.4. Gboa'nın sözde kodu

GBOAnın genel yapısı aşağıdaki sözde kod ile özetlenebilir:

1. Popülasyonu başlat ($X_i, i = 1, 2, \dots, N$)
2. Her bir ajanın uygunluk değerini hesapla
3. En iyi ajanı belirle (X^*)
4. $t = 1$ 'den T_{max} 'a kadar:
 - (a) Her bir ajan için:
 - i. a, A, C, l , ve p parametrelerini güncelle
 - ii. Eğer $p < 0,5$:
 - A. Eğer $|A| < 1$: Kuşatmayı daraltma mekanizması uygula
 - B. Eğer $|A| \geq 1$: Av arama mekanizması uygula
 - iii. Eğer $p \geq 0,5$: Spiral konum güncelleme mekanizması uygula
 - iv. **Kosinüs benzerliğini hesapla:** $cs = cosine(X^*, X(t + 1))$
 - v. **Eğer $cs > \beta$: GSKGM uygula (Denklem 3.21)**
 - (b) Sınırları kontrol et ve ihlal eden çözümleri düzelt
 - (c) Her bir ajanın uygunluk değerini hesapla
 - (d) En iyi ajanı güncelle (X^*)
5. En iyi çözümü döndür

3.3.5. Gboa'nın avantajları

GBOA'nın standart BOA'ya göre temel avantajları şunlardır:

- **Popülasyon Çeşitliliği Korunumu:** Kosinüs benzerliği metriği kullanılarak popülasyon çeşitliliği izlenmekte ve düşük çeşitlilik durumunda GSKGM mekanizması devreye girmektedir. Bu sayede, algoritma yerel minimumlara saplanma riskini azaltmaktadır.
- **Geliştirilmiş Keşif Yeteneği:** GSKGM mekanizması, ajanların sinüzoidal bir yörünge boyunca hareket etmesini sağlayarak arama uzayının daha etkin bir şekilde keşfedilmesine olanak sağlamaktadır.

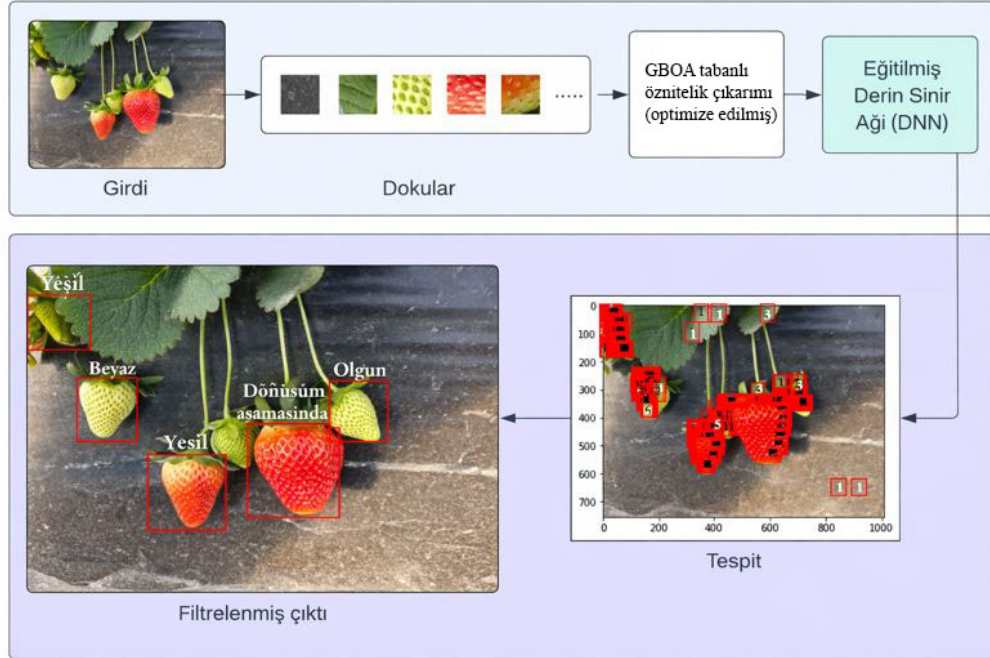
- **Hızlı Yakınsama:** Geliştirilmiş mekanizmalar sayesinde, GBOA standart BOA'ya kıyasla daha hızlı yakınsama göstermektedir.
- **Tutarlı Performans:** Popülasyon çeşitliliğinin korunması sayesinde, algoritmanın farklı çalıştırmalardaki performansı daha tutarlı hale gelmektedir.

3.4. S-Det: Çilek Tespit Sistemi

Bu bölümde, tezin ikinci aşamasında geliştirilen S-Det sisteminin mimarisi, bileşenleri ve çalışma prensipleri detaylı bir şekilde açıklanmaktadır. S-Det sistemi, GBOA'ı hem öznetelik çıkarımı hem de sınıflandırma aşamalarında kullanarak, çilek olgunluk tespiti ve sınıflandırması problemi için bütünlük bir çözüm sunmaktadır.

3.4.1. S-det sisteminin genel mimarisi

S-Det sistemi, üç temel aşamadan oluşmaktadır: (i) segmentasyon, (ii) öznetelik çıkarımı ve (iii) sınıflandırma. Sistemin genel akış diyagramı Şekil 3.14'te gösterilmektedir.



Şekil 3.14. S-Det sisteminin genel mimarisi.

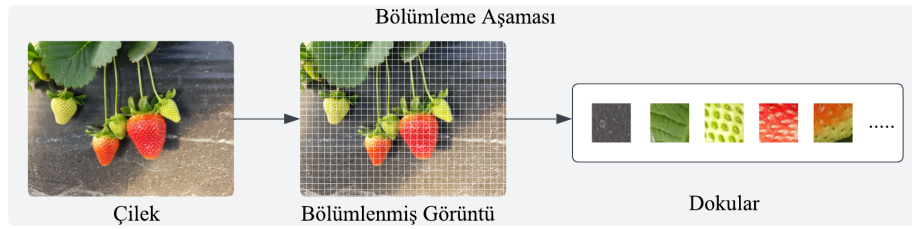
Şekil 3.14'te görüldüğü gibi, S-Det sistemi öncelikle giriş görüntüsünü segmentasyon işlemine tabi tutarak 32x32 piksel boyutunda dokular elde etmektedir. Ardından, GBOA tabanlı otokodlayıcı (GBOA-OK) kullanılarak bu dokulardan öznetelik vektörleri

çıkarılmaktadır. Son olarak, GBOA tabanlı derin sinir ağı (GBOA-DNN) kullanılarak öznitelik vektörleri sınıflandırılmakta ve çileklerin olgunluk aşamaları belirlenmektedir.

3.4.2. Segmentasyon aşaması

Görüntü segmentasyonu, bilgisayarlı görü ve görüntü analizi alanında kritik bir tekniktir ve bir görüntüyü anlamlı bölgelere ayırma işlemini içermektedir. Bu süreç, görüntü içindeki çeşitli nesnelerin, dokuların veya sınırların tanımlanmasını ve çizgilenmesini sağlayarak daha hassas ve bilgilendirici bir analiz yapılmasına olanak sağlamaktadır.

S-Det sisteminin segmentasyon aşamasında, çilek görüntüsünün piksel verileri daha yüksek seviyeli temsillere dönüştürülerek çilek dokuları elde edilmektedir. Bu işlem, görüntüdeki çilek meyvelerinin tespit edilmesi ve 32x32 piksel boyutunda doku parçalarının çıkarılması şeklinde gerçekleştirilmektedir. Segmentasyon aşamasından elde edilen örnek dokular Şekil 3.15'te gösterilmektedir.

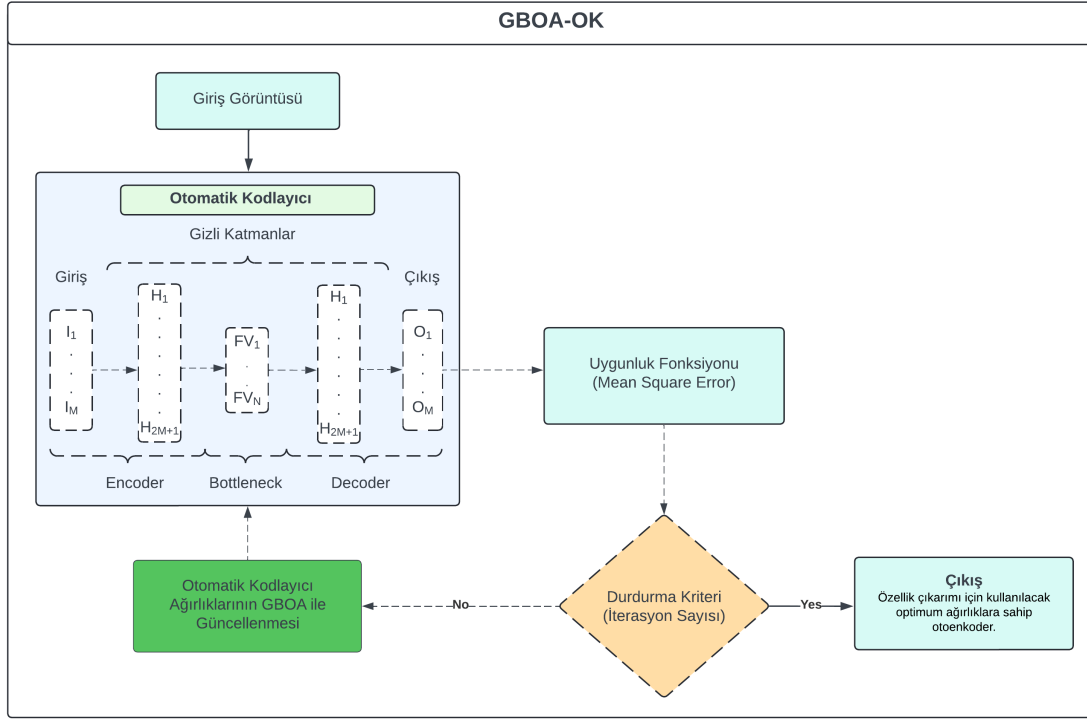


Şekil 3.15. Segmentasyon aşamasından elde edilen 32x32 piksel boyutundaki çilek dokuları.

3.4.3. Gboa tabanlı otomatik kodlayıcı

Derin sinir ağlarının performansı, girdi katmanındaki öznitelik sayısından doğrudan etkilenmektedir. Girdi katmanındaki öznitelik sayısının azaltılması, sınıflandırma sürecinin verimliliğini ve etkinliğini artırmaktadır. Bu nedenle, sınıflandırılacak görüntüdeki öznitelik sayısı, öznitelik çıkarımı yöntemleri kullanılarak azaltılmaktadır.

S-Det sisteminde, çilek görüntülerinden öznitelik çıkarımı için otokodlayıcı kullanılmaktadır. Genel olarak, otokodlayıcıların ağırlıkları geri yayılım (backpropagation) algoritması kullanılarak eğitilmektedir. Bu çalışmada ise, otokodlayıcının optimal ağırlıkları, önerilen GBOA kullanılarak belirlenmekte ve bu sayede girdi görüntülerinden ayırt edici öznitelikler çıkarılmaktadır. Önerilen GBOA-OK yönteminin akış diyagramı Şekil 3.16'te gösterilmektedir.



Şekil 3.16. GBOA-OK yönteminin genel mimarisi.

3.4.3.1. Otokodlayıcı mimarisi

GBOA-OK yönteminde, M girdiye sahip otokodlayıcı, gizli katmanlarında $2M + 1$ nöron ve çıktı katmanında M nöron içermektedir. M değeri Denklem 3.22'e göre belirlenmektedir:

$$M = (\text{Piksel Sayısı}) \times NC \quad (3.22)$$

Burada, NC girdi görüntüsündeki renk kanalı sayısını, Piksel Sayısı ise girdi görüntüsünün çözünürlüğünü temsil etmektedir. Örneğin, 32x32 piksel boyutunda ve üç renk kanalına (kırmızı, yeşil, mavi) sahip bir görüntü için $M = 32 \times 32 \times 3 = 3072$ olarak hesaplanmaktadır.

Otokodlayıcının darboğaz (bottleneck) katmanındaki nöron sayısı N , öznitelik vektörünün boyutunu belirlemektedir ve Denklem 3.23'e göre hesaplanmaktadır:

$$N = \sqrt{\text{Piksel Sayısı} \times NC} \quad (3.23)$$

Örneğin, 32x32 piksel boyutunda ve üç renk kanalına sahip bir görüntü için $N = \sqrt{32 \times 32 \times 3} = 32 \times 3 = 96$ olarak hesaplanmaktadır. Bu değer, aynı zamanda öznitelik

vektörünün boyutunu da temsil etmektedir.

3.4.3.2. Aktivasyon fonksiyonu

GBOA-OK yönteminde, otokodlayıcının tüm katmanlarında softplus aktivasyon fonksiyonu kullanılmaktadır. Softplus aktivasyon fonksiyonu, her zaman sıfırdan büyük bir çıktı değeri üretme özelliğine sahip olup, bu sayede inaktif nöronların oluşmasını etkin bir şekilde önlemektedir. Softplus aktivasyon fonksiyonu Denklem 3.24'e göre hesaplanmaktadır:

$$F(x) = \log(1 + \exp(x)) \quad (3.24)$$

3.4.3.3. Uygunluk fonksiyonu

GBOA-OK yönteminde, tahmin edilen çıktılardaki kaybı hesaplamak için MSE fonksiyonu kullanılmakta ve bu fonksiyon uygunluk fonksiyonu olarak görev yapmaktadır. MSE fonksiyonu Denklem 3.25'e göre hesaplanmaktadır:

$$MSE = \frac{\sum_{i=1}^n (y_i - y'_i)^2}{n} \quad (3.25)$$

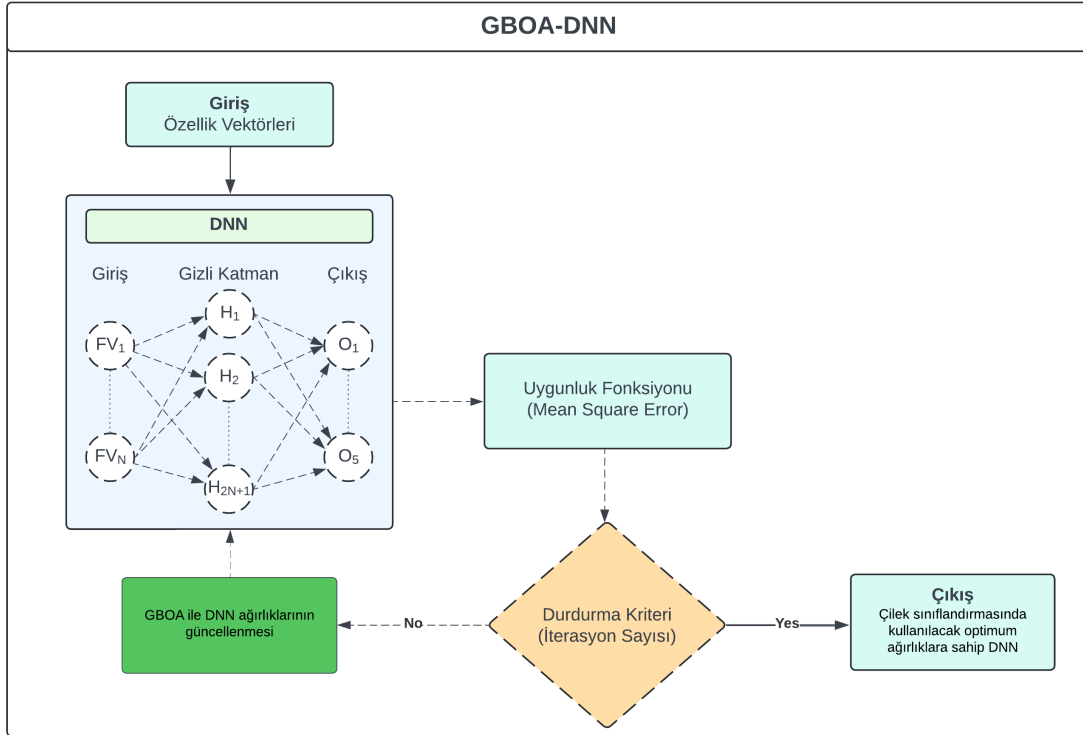
Burada, y_i gerçek çıktıyı, y'_i tahmin edilen çıktıyı, n ise eğitim veri setindeki örnek sayısını temsil etmektedir. GBOA, MSE değerini minimize edecek şekilde otokodlayıcının ağırlık parametrelerini optimize etmektedir.

GBOA-OK yöntemi ile eğitilen otokodlayıcı, girdi dokularını etkin bir şekilde öznitelik vektörlerine dönüştürmektedir. Elde edilen düşük boyutlu öznitelik vektörleri, sonraki sınıflandırma aşamasında derin sinir ağının verimliliğini ve etkinliğini artırmaktadır.

3.4.4. Geliştirilmiş balina optimizasyon algoritması tabanlı dnn

Bu aşamada, önceki aşamadan elde edilen öznitelik vektörleri, çilek tespiti için derin sinir ağı kullanılarak sınıflandırılmaktadır. Derin sinir ağlarının eğitimi sırasında, gradyan tabanlı optimizasyon algoritmaları bazen yerel minimumlara saplanmakta ve bu durum ideal ağırlıklara ulaşılmasını engelleyerek uzun çalışma sürelerine neden olmaktadır. Bu nedenle, S-Det sistemi bu sınırlamaları aşmak için GBOA ile eğitilmiş derin sinir ağı kullanmaktadır.

GBOA kullanarak derin sinir ağı eğitimi, hesaplama verimliliği ve gradyan tabanlı optimizasyon yöntemlerinin sınırlarının ötesinde arama uzayını etkin bir şekilde keşfederek optimal ağırlıklara ulaşma açısından avantajlar sağlamaktadır. GBOA-DNN yönteminin akış diyagramı Şekil 3.17’de gösterilmektedir.



Şekil 3.17. GBOA-DNN yönteminin genel mimarisi.

3.4.4.1. Derin sinir ağı (deep neural network, dnn) mimarisi

GBOA-DNN yönteminde, öznitelik vektörü derin sinir ağının yapısını belirlemektedir. Girdi katmanındaki nöron sayısı N , öznitelik vektörünün boyutuna eşittir. Gizli katmandaki nöron sayısı ise $2N + 1$ olarak belirlenmektedir (Patel vd., 2024). Çıktı katmanı, çilek olgunluk aşamalarını temsil eden beş nöron içermektedir: beyaz, yeşil, dönüşüm aşaması, olgun ve arkaplan.

Optimizasyon boyutu olan ağırlık sayısı, Denklem 3.26’e göre hesaplanmaktadır:

$$N_{\text{ağırlık}} = (N_{\text{girdi}} \times N_{\text{gizli}}) + N_{\text{gizli bias}} + (N_{\text{gizli}} \times N_{\text{çıkı}}) + N_{\text{çıkı bias}} \quad (3.26)$$

Burada, N_{girdi} girdi katmanındaki nöron sayısını, N_{gizli} gizli katmandaki nöron

sayısını, $N_{\text{çıkıtı}}$ çıktı katmanındaki nöron sayısını, $N_{\text{gizli bias}}$ ve $N_{\text{çıkıtı bias}}$ ise sırasıyla gizli ve çıktı katmanlarındaki bias değerlerinin sayısını temsil etmektedir.

3.4.4.2. Ağırlık optimizasyonu

GBOA-DNN yönteminin başlangıcında, başlangıç ağırlıkları $[-1, 1]$ aralığında rastgele olarak başlatılmaktadır. Ardından, durdurma kriteri karşılanana kadar GBOA kullanılarak ağırlıklar güncellenmektedir. Bu süreç sonunda, çilek sınıflandırması için kullanılacak derin sinir ağı elde edilmektedir.

Derin sinir ağının eğitimi sırasında, önceki aşamada olduğu gibi, tüm katmanlarda softmax aktivasyon fonksiyonu ve MSE kayıp fonksiyonu kullanılmaktadır. GBOA, sınıflandırma doğruluğunu maksimize edecek şekilde derin sinir ağının ağırlık parametrelerini optimize etmektedir.

3.4.5. Sınıf yoğunluğu hesaplama

S-Det sisteminde, sınıflandırılan çilekler, ilgili sınıfın yoğunluğuna (d) göre çıktı görüntüsünde etiketlenmektedir. Sınıf yoğunluğu, Denklem 3.27'e göre hesaplanmaktadır:

$$d_i = \frac{m_i}{v}, \quad i = 0 \text{ to sınıf sayısı} \quad (3.27)$$

Burada, m_i ilgili sınıf için tespit edilen doku sayısını, v ise toplam doku sayısını temsil etmektedir. Bu yoğunluk değeri, çıktı görüntüsünde her bir sınıfın görsel temsilini belirlemektedir.

3.4.6. S-det sisteminin özellikleri

S-Det sisteminin temel özellikleri şu şekilde özetlenebilir:

- **Bütünleşik Optimizasyon Yaklaşımı:** Sistem, aynı optimizasyon algoritması olan GBOA'yı hem öznitelik çıkarımı hem de sınıflandırma aşamalarında kullanarak koordineli bir optimizasyon gerçekleştirmektedir.
- **Adaptif Öznitelik Çıkarımı:** GBOA-OK yöntemi, görüntü içeriğine göre adaptif olarak öğrenilen ağırlıklarla ayırt edici öznitelikler çıkarmaktadır.
- **Etkin Sınıflandırma:** GBOA-DNN yöntemi, gradyan tabanlı yöntemlerin yerel

minimum sorununu aşarak daha yüksek sınıflandırma performansı sağlamaktadır.

- **Ölçeklenebilir Mimari:** Sistem, farklı görüntü boyutları ve renk kanalı sayılarına uyum sağlayabilen esnek bir mimariye sahiptir.
- **Gerçek Zamanlı Uygulanabilirlik:** Öznitelik çıkarımı sayesinde düşük boyutlu öznitelik vektörleri kullanıldığından, sistem gerçek zamanlı uygulamalara uygun hale gelmektedir.

3.5. Kullanılan Veri Setleri

Bu tez çalışmasının deneysel değerlendirmelerinde, birinci ve ikinci aşama için farklı veri setleri kullanılmıştır. Bu bölümde kullanılan veri setleri kısaca tanıtılmış ve temel özellikleri verilmiştir.

3.5.1. Boa-ga ile birlikte kullanılan veri setleri

Önerilen BOA-GA, yaygın olarak kullanılan ve halka açık beş dengesiz veri seti üzerinde değerlendirilmiştir: HAM10000, Plant Village, Cassava, Plant Leaves ve Flowers102. Bu veri setleri, tıbbi ve tarımsal uygulamaları kapsamakta ve farklı sayıda sınıf ve dengesizlik oranları sergilemektedir.

Dengesizlik oranı (Denklem 3.28), veri setindeki en fazla örneğe sahip sınıfın örnek sayısının ($n_{\max}$), en az örneğe sahip sınıfın örnek sayısına ($n_{\min}$) oranı olarak hesaplanmaktadır:

$$\text{Dengesizlik Oranı} = \frac{n_{\max}}{n_{\min}} \quad (3.28)$$

Burada $n_{\max}$ çoğunluk sınıfındaki örnek sayısını, $n_{\min}$ ise azınlık sınıfındaki örnek sayısını temsil etmektedir. Oran 1'e ne kadar uzaksa, sınıf dengesizliği o kadar belirgindir.

Çizelge 3.1 veri setlerinin temel özelliklerini, sınıf sayısı, örnek sayısı ve dengesizlik oranlarını özetlemektedir. Veri setleri arasında, HAM10000 en yüksek dengesizlik oranını sergilemekte ve bu nedenle aşırı örnekleme stratejilerini değerlendirmek için özellikle uygundur.

Çizelge 3.1. BOA-GA deneylerinde kullanılan veri setlerinin özellikleri

Veri Seti	Sınıf Sayısı	Örnek Sayısı	Dengesizlik Oranı
Plant Leaves	22	4502	4,48
Flowers102	102	8189	6,45
Cassava	5	9430	8,40
Plant Village	38	54305	36,23
HAM10000	7	10015	58,30

3.5.2. Gboa ve s-det ile birlikte kullanılan veri setleri

GBOA optimizasyon performansı için çeşitli sınıflandırma veri setleri, S-Det sistemi için ise hem hedef uygulama alanını temsil eden bir görüntü veri seti hem de genel amaçlı sınıflandırma veri setleri kullanılmıştır. Aşağıda bu veri setleri kısaca tanıtılmaktadır.

3.5.2.1. Sınıflandırma veri setleri

GBOA'nın performansını değerlendirmek için, California Üniversitesi Irvine (UCI) Makine Öğrenmesi Deposu ve DELVE Deposu'ndan 15 farklı sınıflandırma veri seti kullanılmıştır (Dua ve Graff, 2019; DELVE, t.y.). Bu veri setleri, GBOA'nın diğer optimizasyon algoritmaları ile karşılaştırılması amacıyla benchmark veri seti olarak kullanılmaktadır. Veri setlerinin özellikleri Çizelge 3.2'te verilmiştir.

Çizelge 3.2. Sınıflandırma veri setleri

Veri Seti	Sınıflar	Öznitelikler	Eğitim Örnekleri	Test Örnekleri
Australian	2	14	455	235
Blood	2	4	493	255
Breast cancer	2	8	461	238
Chess	2	36	2109	1087
Credit	2	61	670	330
Diabetes	2	8	506	262
Hepatitis	2	10	102	53
Ionosphere	2	33	231	120
Liver	2	6	79	41
Parkinson	2	22	128	67
Ring	2	20	4884	2516
Seed	3	7	138	72
Tic-tac-toe	2	9	632	326
Titanic	2	3	1452	749
Wine	3	13	117	61

3.5.2.2. Strawtex veri seti

StrawTex veri seti, çilek tarlalarından elde edilen 8000 renkli görüntüden oluşan Strawberry Digital Images (StrawDI) veri setinden elde edilmiştir (Pérez-Borrero vd., 2020). StrawTex veri seti, StrawDI üzerinden 2000 adet doku örneği kesilerek elde edilmiş olup 5 sınıftan oluşmaktadır: beyaz, yeşil, dönüşüm, olgun ve arkaplan. Her sınıf 32x32 piksel boyutlarında görüntülerle temsil edilmektedir. Elde edilen StrawTex veri seti, görüntülerden çilek meyvelerini doğru bir şekilde tespit etmek için DNN eğitimi için kullanılmıştır.

Doğal ortamında, çilek bitkisinin meyveleri, dalları, çiçekleri ve yaprakları yön ve boyut parametreleri açısından farklılıklar göstermektedir. DNN'nin bu farklılıkları etkili bir şekilde öğrenmesi ve sınıflandırma performansını artırması için önemli miktarda örneğe ihtiyaç duymaktadır. DNN'nin öğrenme yeteneklerini geliştirmek için, StrawTex veri setine döndürme, öteleme ve yakınlaştırma veri artırma teknikleri uygulanmış ve doku örneklerinin sayısı 6000'e çıkarılmıştır.

3.5.2.3. Görüntü veri setleri

S-Det sisteminin öznetelik çıkarımı ve sınıflandırma performansını farklı problem türlerinde değerlendirmek için, Cassava, CIFAR-10 (Kanada İleri Araştırmalar Enstitüsü; Canadian Institute For Advanced Research) ve MNIST (Değiştirilmiş Ulusal Standartlar ve Teknoloji Enstitüsü; Modified National Institute of Standards and Technology) görüntü veri setleri kullanılmıştır. Cassava, CIFAR-10 ve MNIST veri setlerinin temel özellikleri Çizelge 3.3'te özetlenmiştir.

Çizelge 3.3. Görüntü veri setleri

Veri Seti	Sınıflar	Örnekler
Cassava	5	9430
CIFAR-10	10	60000
MNIST	10	70000

4. DENEYSEL SONUÇLAR

Bu bölümde, geliştirilen yöntemlerin deneysel sonuçları detaylı bir şekilde sunulmaktadır. Bölüm, birinci aşamada geliştirilen BOA-GA yönteminin performans değerlendirmesi ile başlamakta, ardından ikinci aşamada geliştirilen GBOA ve S-Det sisteminin kapsamlı test sonuçları ile devam etmektedir. Elde edilen sonuçlar, görselleştirmeler ve istatistiksel analizlerle desteklenmektedir.

4.1. BOA-GA Yönteminin Deneysel Sonuçları

Bu bölümde, birinci aşamada geliştirilen BOA-GA'nın deneysel sonuçları sunulmaktadır. Öncelikle, BOA tabanlı kenar tespiti yönteminin performansı değerlendirilmiş, ardından BOA-GA yönteminin farklı dengesiz veri setlerindeki sınıflandırma performansı ve kenar tespiti entegrasyonunun etkisi detaylı bir şekilde raporlanmıştır.

4.1.1. Deneysel kurulum ve değerlendirme metrikleri

Bu bölümde, birinci aşama deneylerinin hangi donanımda yürütüldüğü ve sonuçların hangi metriklerle ölçüldüğü tanıtılmaktadır. Tüm deneyler, Intel Core i7 işlemci (2.9 GHz), 32 GB RAM ve Windows 10 işletim sistemi kullanan bir masaüstü bilgisayarda yapılmıştır. BOA-GA değerlendirmesi iki ayrı başlıkta ele alınmaktadır: (i) önerilen kenar tespiti yönteminin performansı, (ii) kenar tabanlı görüntü artırımı uygulandıktan sonra sınıflandırma performansı. Dengesiz veri setlerinde model performansını ölçmek için literatürde yaygın kullanılan altı metrik kullanılmıştır. Bu metrikler, ikili (ve çok sınıflı) sınıflandırmada tahminlerin gerçek etiketlerle karşılaştırılmasıyla elde edilen karışıklık matrisinden türetilir. Karışıklık matrisinde dört bileşen vardır: doğru pozitif (True Positive, TP), doğru negatif (True Negative, TN), yanlış pozitif (False Positive, FP) ve yanlış negatif (False Negative, FN). İkili sınıflandırma için karışıklık matrisi yapısı Çizelge 4.1'de verilmiştir.

Kullanılan metrikler ve anlamları kısaca şöyledir: Doğruluk (Accuracy, ACU), doğru sınıflandırılan örneklerin tüm örneklerle oranıdır. Kesinlik (Precision, PRE), modelin “pozitif” dediği örneklerin gerçekten kaçının pozitif olduğunu gösterir. Duyarlılık (Recall, REC), doğru pozitif örneklerin kaçının doğru tahmin edildiğini ölçer.

Çizelge 4.1. Sınıflandırma için karışıklık matrisi yapısı

Tahmin	Gerçek Etiket	
	Pozitif	Negatif
Pozitif	TP (Doğru Pozitif)	FP (Yanlış Pozitif)
Negatif	FN (Yanlış Negatif)	TN (Doğru Negatif)

Yanlış pozitif oranı (False Positive Rate, FPR) ve yanlış negatif oranı (False Negative Rate, FNR), sırasıyla negatiflerin yanlışlıkla pozitif sayılma ve pozitiflerin kaçırılma oranlarıdır. F1-skoru (F1-Score, F1) ise kesinlik ile duyarlılığın harmonik ortalamasıdır ve dengesiz veri setlerinde tek başına doğruluktan daha bilgilendiricidir. Tüm metrikler, TP, TN, FP ve FN cinsinden Denklem 4.1, 4.2, 4.3, 4.4, 4.5 ve 4.6'ya göre hesaplanır:

$$ACU = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.1)$$

$$PRE = \frac{TP}{TP + FP} \quad (4.2)$$

$$REC = \frac{TP}{TP + FN} \quad (4.3)$$

$$FPR = \frac{FP}{FP + TN} \quad (4.4)$$

$$FNR = \frac{FN}{FN + TP} \quad (4.5)$$

$$F1 = \frac{2 \cdot PRE \cdot REC}{PRE + REC} \quad (4.6)$$

4.1.2. Boa tabanlı kenar tespiti yönteminin performans değerlendirmesi

Önerilen BOA tabanlı kenar tespiti yönteminin etkinliğini değerlendirmek için, yaygın olarak kullanılan Lena benchmark görüntüsü (512×512 piksel) üzerinde kenar tespiti gerçekleştirilmiştir. Tespit edilen kenarlar, Canny, Laplacian, Roberts ve Sobel operatörleri gibi kenar tespiti yöntemlerinin çıktılarının ortalaması alınarak elde edilen referans kenar görüntüleri ile karşılaştırılmıştır. Karşılaştırmada kullanılan kenar dedektörlerinin matematiksel temelleri aşağıda aşamalar halinde adım adım verilmektedir.

Sobel operatörü (birinci türev tabanlı gradyan operatörü):

Adım 1: Yatay gradyan bileşeni G_x hesaplanır: $G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I$

Adım 2: Dikey gradyan bileşeni G_y hesaplanır: $G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I$

Adım 3: Kenar büyüklüğü elde edilir: $|G| = \sqrt{G_x^2 + G_y^2}$

Roberts operatörü (2×2 köşegen gradyan operatörü):

Adım 1: Köşegen gradyan bileşeni G_x hesaplanır: $G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} * I$

Adım 2: Köşegen gradyan bileşeni G_y hesaplanır: $G_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} * I$

Adım 3: Kenar büyüklüğü elde edilir: $|G| = \sqrt{G_x^2 + G_y^2}$

Laplacian operatörü (ikinci türev tabanlı, sıfır geçişi tespiti):

Adım 1: Ayırık Laplacian konvolüsyonu uygulanır: $\nabla^2 I = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} * I =$
 $I(x+1, y) + I(x-1, y) + I(x, y+1) + I(x, y-1) - 4I(x, y)$

Adım 2: Sıfır geçişleri veya eşikleme ile kenar pikselleri belirlenir.

Canny kenar dedektörü (çok aşamalı süreç):

Adım 1 — Gauss yumuşatma: Görüntü I Gauss filtresi ile yumuşatılır: $I_s = G_\sigma * I$

Adım 2 — Gradyan hesaplama: Genellikle Sobel operatörü ile büyüklük $M = \sqrt{G_x^2 + G_y^2}$ ve yön $\theta = \arctan(G_y/G_x)$ hesaplanır.

Adım 3 — Maksimum olmayan bastırma: Kenar kalınlığı tek piksele indirgenir; gradyan yönünde maksimum olmayan değerler elenir.

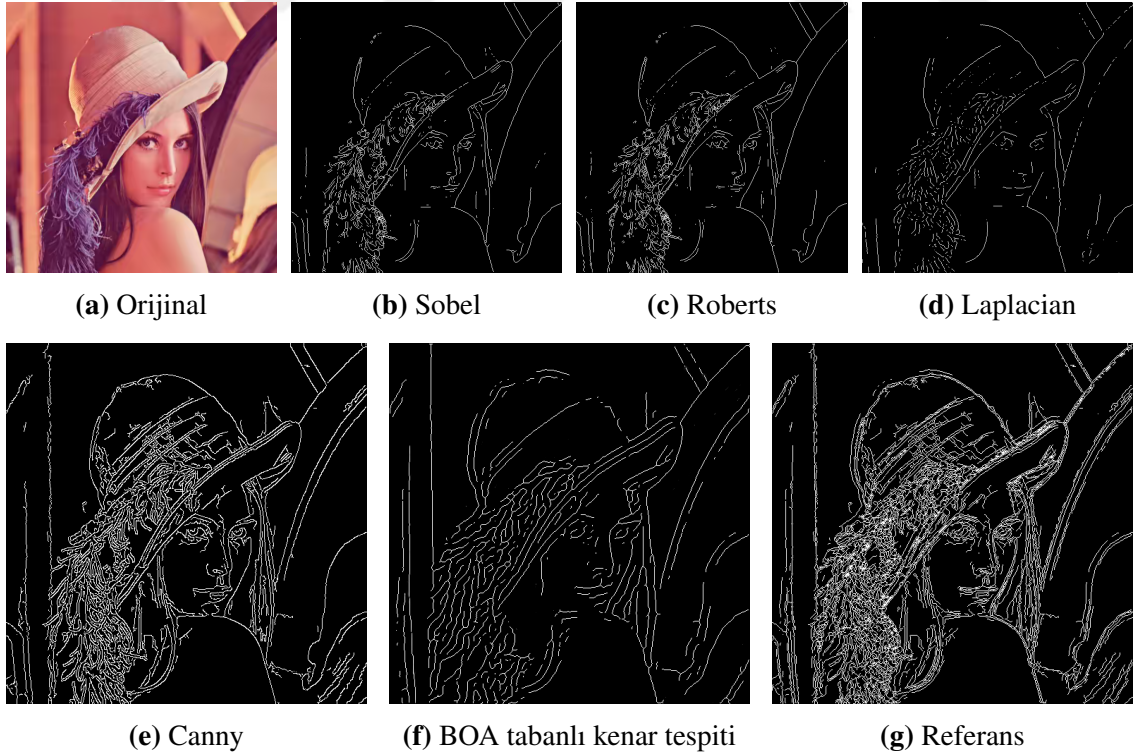
Adım 4 — Çift eşikli izleme (histerezisli eşikleme): Üst ve alt eşikler ile zayıf ve güçlü kenarlar ayrıştırılır; güçlü kenarlara bağlı zayıf kenarlar korunur.

Piksel bazlı karşılaştırma, Çizelge 4.2’de gösterilen karışıklık matrisi kullanılarak gerçekleştirilmiştir. TP, FP, FN ve TN değerleri Denklem 4.1–4.6 ile ACU, PRE, REC, FPR, FNR ve F1 metriklerine dönüştürülmüştür.

Çizelge 4.2. Kenar tespiti için piksel bazlı karşılaştırma karışıklık matrisi

Kenar Tespit Edici	Gerçek Değer	
	Kenar pikseli	Kenar olmayan piksel
Kenar pikseli	TP	FP
Kenar olmayan piksel	FN	TN

Şekil 4.1 kenar tespit ediciler ile önerilen yöntem arasındaki görsel karşılaştırmayı sunmaktadır. Şekil 4.1’de (a) orijinal görüntü, (b)–(e) kenar tespit edicilerin (Sobel, Roberts, Laplacian, Canny) çıktıları, (f) önerilen BOA tabanlı yaklaşımın ürettiği kenar görüntüsü ve (g) yöntemlerin ortalaması ile oluşturulan referans kenar görüntüsü birlikte gösterilmektedir. Görsel karşılaştırma, BOA tabanlı kenar tespiti yönteminin gürültüyü bastırırken daha sürekli ve belirgin kenarlar üretebildiğini göstermektedir.



Şekil 4.1. Kenar tespit edicilerin görsel karşılaştırması

Çizelge 4.3. Önerilen yöntem ile kenar tespit edicilerin karşılaştırması

Yöntem	ACU	PRE	REC	FPR	FNR	F1
Canny	0,935	0,776	0,995	0,081	0,004	0,872
Laplacian	0,951	0,957	0,710	0,005	0,289	0,815
Roberts	0,950	0,891	0,786	0,018	0,213	0,835
Sobel	0,944	0,853	0,789	0,026	0,210	0,820
BOA tabanlı kenar tespiti	0,968	0,925	0,916	0,018	0,083	0,920

Performans metriklerine ilişkin nicel sonuçlar Çizelge 4.3'te verilmiştir; çizelge, ACU, PRE, REC ve F1 sütunlarında en yüksek, FPR ve FNR sütunlarında ise en düşük değerleri koyu vurgulayarak sunmaktadır. Önerilen yöntem, değerlendirilen tüm yöntemler arasında en yüksek doğruluk (ACU=0,968, %96,8) ve F1-skoru (0,920) değerlerini elde etmiştir. Laplacian operatörü en yüksek kesinliği (PRE=0,957) ve en düşük yanlış pozitif oranını (FPR=0,005) sağlamasına rağmen, yüksek yanlış negatif oranı (FNR=0,289) nedeniyle kenar piksellerinin %29'unu kaçırmaktadır. Canny operatörü en yüksek duyarlılığı (REC=0,995) ve en düşük FNR'yi (0,004) gösterirken, yanlış pozitif oranı (FPR=0,081) yöntemler arasında en yüksek değeri almaktadır. Roberts ve Sobel sırasıyla ACU=0,950 ve 0,944, F1=0,835 ve 0,820 ile ara değerler sunmaktadır. Önerilen yöntem, REC=0,916 ve PRE=0,925 ile kesinlik-duyarlılık dengesinde üstünlük sağlayarak, tek bir metrikte en iyi olan standart yöntemlerin sınırlılıklarını aşmaktadır.

4.1.3. Deneysel kurulum

Kullanılan veri setleri Bölüm 3.5'te tanıtılmıştır. BOA-GA yönteminin sınıflandırma performansına etkisini değerlendirmek için, her veri setinin hem orijinal (dengesiz) hali hem de BOA-GA ile aşırı örnekleme uygulanmış (dengeli) hali sınıflandırma testlerine tabi tutulmuştur. Aşırı örneklemede, tüm sınıflar çoğunluk sınıfındaki örnek sayısına eşitlenecek şekilde dengelenmiştir; yani her azınlık sınıfı için Denklem 3.19'e göre hesaplanan $N = (M_j - M_i)$ kadar sentetik örnek üretilerek sınıf içi örnek sayısı çoğunluk sınıfına (M_j) getirilmiştir.

ImageNet ağırlıkları ile başlatılan ve her veri seti üzerinde ince ayar yapılan önceden eğitilmiş evrişimli sinir ağı modelleri DenseNet201 (Huang vd., 2017), MobileNet (Howard vd., 2017) ve VGG16 (Simonyan ve Zisserman, 2014) sınıflandırıcı olarak kullanılmıştır. SGD, 0,001 öğrenme oranı ile optimizasyon algoritması olarak

seçilmiş; havuzlama stratejisi olarak ortalama havuzlama uygulanmıştır. Raporlanan tüm sonuçlar, çok sınıflı veri setleri için sınıflar arası makro ortalama (macro-average) performansı temsil etmektedir; böylece her sınıf eşit ağırlıkla değerlendirilerek sınıf dengesizliğinden kaynaklanan yanıltıcı metrikler önlenmektedir.

4.1.4. Balina optimizasyon algoritması tabanlı görüntü artırma yönteminin sınıflandırma sonuçları

Dengesiz ve BOA-GA ile dengelenmiş veri setleri üzerinde DenseNet201, MobileNet ve VGG16 ile elde edilen makro ortalama ACU, PRE ve F1 değerleri Çizelge 4.4'te verilmiştir. Bu bölümde raporlanan tüm sınıflandırma testlerinde veri setleri %80 eğitim ve %20 test olacak şekilde ayrılmıştır. Bulgular, aşırı örnekleme beş veri setinin tamamında ve üç sınıflandırıcıda da performansı artırdığını; artışın büyüklüğünün ise dengesizlik oranı ile birlikte değerlendirildiğini göstermektedir.

Nicel etkiyi somutlaştırmak amacıyla DenseNet201 sonuçları esas alındığında, dengesizlik oranı 58,30 olan HAM10000 veri setinde dengesiz sürüm için ACU=0,746 ve F1=0,470 iken, BOA-GA sonrasında ACU=0,986 ve F1=0,986 elde edilmiştir. Benzer biçimde Cassava veri setinde ACU 0,674'ten 0,891'e, Plant Leaves veri setinde 0,779'dan 0,907'ye, Flowers102 veri setinde 0,786'dan 0,942'ye yükselmiş; Plant Village veri setinde ise 0,974'ten 0,982'ye çıkmıştır. Bu sonuçlar, BOA-GA tabanlı aşırı örnekleme özelliğiyle yüksek dengesizlik oranına sahip veri setlerinde belirgin kazanımlar ürettiğini, düşük/orta dengesizlik seviyelerinde ise daha sınırlı fakat tutarlı iyileşmeler sağladığını ortaya koymaktadır.

Çizelge 4.4. BOA-GA yönteminin görüntü veri setleri üzerindeki performans değerlendirmesi

Veri Seti	Versiyon	DenseNet201			MobileNet			VGG16		
		ACU	PRE	F1	ACU	PRE	F1	ACU	PRE	F1
HAM10000	Dengesiz	0,746	0,627	0,470	0,784	0,647	0,590	0,811	0,705	0,674
	Dengeli	0,986	0,986	0,986	0,980	0,980	0,980	0,981	0,981	0,982
Cassava	Dengesiz	0,674	0,600	0,496	0,652	0,562	0,496	0,616	0,476	0,444
	Dengeli	0,891	0,892	0,892	0,912	0,912	0,912	0,870	0,870	0,870
Plant Leaves	Dengesiz	0,779	0,837	0,741	0,925	0,921	0,913	0,869	0,842	0,838
	Dengeli	0,907	0,929	0,906	0,984	0,985	0,984	0,953	0,955	0,955
Plant Village	Dengesiz	0,974	0,969	0,958	0,930	0,918	0,883	0,896	0,942	0,900
	Dengeli	0,982	0,982	0,982	0,959	0,959	0,958	0,976	0,976	0,976
Flowers102	Dengesiz	0,786	0,780	0,761	0,623	0,591	0,577	0,783	0,782	0,766
	Dengeli	0,942	0,941	0,941	0,848	0,845	0,845	0,947	0,947	0,947

4.1.5. Kenar tabanlı artırım yöntemleri ile karşılaştırma

Önerilen yöntemin etkinliğini daha ayrıntılı biçimde değerlendirmek için, BOA-GA tabanlı aşırı örnekleme stratejisi, Canny, Laplacian, Roberts ve Sobel kenar tespit edicilerine dayalı artırım yaklaşımları ile karşılaştırılmıştır. Bu karşılaştırma testlerinde veri seti %80 eğitim ve %20 test olacak şekilde ayrılmıştır. HAM10000 veri seti üzerinde elde edilen sınıflandırma sonuçları Çizelge 4.5'te verilmiştir.

DenseNet201 sonuçları dikkate alındığında, dengesiz sürüm için ACU=0,746 iken BOA-GA ile dengelenmiş sürümde ACU=0,986 elde edilmiştir. Kenar tespit edicilerine dayalı artırım yöntemleri arasında DenseNet201 için en yüksek performans Laplacian ile ACU=0,976 (F1=0,976) değerleriyle sağlanmış; BOA-GA aynı sınıflandırıcıda ACU=0,986 (F1=0,986) ile bu değer üzerine çıkmıştır. Dolayısıyla BOA-GA'nın katkısı, yalnızca sınıf dağılımını dengelemekle sınırlı kalmayıp kenar temsillerinin sınıflandırma açısından ayırt ediciliğini artıran bir üretim mekanizması olarak değerlendirilebilir; bu üstünlüğün MobileNet ve VGG16 sonuçlarında da korunduğu Çizelge 4.5'te görülmektedir.

Çizelge 4.5. HAM10000 veri seti üzerinde BOA-GA ve kenar tespit ediciler aşırı örnekleme değerlendirmesi

HAM10000	Versiyon	DenseNet201			MobileNet			VGG16		
		ACU	PRE	F1	ACU	PRE	F1	ACU	PRE	F1
Orijinal	Dengesiz	0,746	0,627	0,470	0,784	0,647	0,590	0,811	0,705	0,674
BOA-GA ile artırılmış	Dengeli	0,986	0,986	0,986	0,980	0,980	0,980	0,981	0,981	0,982
Canny ile artırılmış	Dengeli	0,959	0,959	0,959	0,936	0,936	0,936	0,926	0,929	0,926
Laplacian ile artırılmış	Dengeli	0,976	0,976	0,976	0,933	0,933	0,933	0,794	0,871	0,785
Roberts ile artırılmış	Dengeli	0,974	0,974	0,974	0,932	0,931	0,931	0,860	0,881	0,859
Sobel ile artırılmış	Dengeli	0,976	0,976	0,976	0,927	0,926	0,926	0,955	0,958	0,955

4.1.6. Literatürdeki güncel yöntemler ile karşılaştırma

Önerilen BOA-GA yönteminin sınıflandırma performansı, HAM10000 veri seti üzerinde literatürde rapor edilen güncel aşırı örnekleme/veri artırma yaklaşımları ile karşılaştırılmış; elde edilen nicel sonuçlar Çizelge 4.6'da verilmiştir. Çizelge 4.6 incelendiğinde, BOA-GA'nın ACU=0,98, PRE=0,986 ve F1=0,986 değerleri ile karşılaştırmaya dahil edilen çalışmalar arasında en yüksek performansı sağladığı görülmektedir. Literatürde en yüksek doğruluk ve F1-skoru, Sandeep Kumar vd. (2023) tarafından ACU=0,97 ve F1=0,971 olarak raporlanmış; BOA-GA bu değerlerin üzerinde

sonuç üretmiştir. Kesinlik açısından en yüksek literatür değeri Dakhli ve Barhoumi (2023) tarafından $PRE=0,972$ olarak bildirilmiş; BOA-GA aynı metrikte $PRE=0,986$ elde etmiştir. Bu bulgular, BOA-GA'nın kenar bilgisine dayalı üretim mekanizmasıyla hem doğruluk hem de sınıf-temelli ölçütlerde (PRE, F1) eşzamanlı kazanım ürettiğini ve HAM10000 gibi yüksek dengesizlik oranına sahip senaryolarda literatürdeki yaklaşımlara kıyasla daha güçlü bir dengeleme etkisi sunduğunu göstermektedir.

Çizelge 4.6. HAM10000 veri seti aşırı örnekleme performans değerlendirmeleri

Yöntem	Sınıflandırıcı	Aşırı Örneklemeye Yöntemi	ACU	PRE	F1
(Singh vd., 2023)	VGG19	Flipping, cropping	0,91	0,900	0,890
(Ahmed vd., 2023)	AlexNet	Flipping, cropping, translation, color jitting	0,95	0,840	0,810
(Khan vd., 2021)	Softmax, Naive bayes, M-SVM, F-KNN	Masking	0,83	0,843	0,835
(Sandeep Kumar vd., 2023)	Modified SqueezeNet	Flipping, rotation, zoom, shifting	0,97	0,971	0,971
(Kondaveeti ve Edupuganti, 2020)	Resnet50, MobileNet, Xception, InceptionV3	Flipping, cropping, rotating	0,90	0,900	0,894
(Dakhli ve Barhoumi, 2023)	ResNet-V2	Segmentation	0,95	0,972	0,963
BOA-GA	VGG16, MobileNet, DenseNet201	Kenar tespiti	0,98	0,986	0,986

4.2. GBOA ve S-Det Sisteminin Deneysel Sonuçları

Bu bölümde, tezin ikinci aşamasında geliştirilen GBOA ve S-Det sistemine ilişkin deneysel bulgular sistematik biçimde sunulmaktadır. Deney tasarımı üç tamamlayıcı bileşenden oluşmaktadır: (i) GBOA'nın optimizasyon kabiliyeti, 15 farklı benchmark sınıflandırma veri seti üzerinde karşılaştırmalı olarak değerlendirilmiştir; (ii) önerilen S-Det sisteminin hedef uygulama senaryosundaki başarımı StrawTex veri seti üzerinde farklı yöntem grupları ile kıyaslanmıştır; (iii) sınıflandırma bileşeni olarak kullanılan GBOA-DNN'nin genellenebilirliği, Cassava, CIFAR-10 ve MNIST benchmark görüntü veri setlerinde test edilerek literatürde raporlanan sonuçlarla ilişkilendirilmiştir. Böylece hem optimizasyon düzeyindeki kazanımların hem de uçtan

uca sistem performansının tutarlı biçimde analiz edilmesi amaçlanmıştır.

4.2.1. Gboa'nın karşılaştırmalı optimizasyon testleri

GBOA'nın optimizasyon performansını değerlendirmek amacıyla, GBOA ile eğitilmiş DNN modeli; temel yaklaşım olarak BP ve metasezgisel tabanlı alternatifler olarak BOA, GA, PSO, ACO, DE, ES ve PBIL ile eğitilmiş modellerle karşılaştırılmıştır. Bu karşılaştırma, önerilen arama mekanizmasının farklı problem türlerinde yakınsama kararlılığı ve çözüm kalitesi üzerindeki etkisini ortaya koymayı hedeflemektedir.

4.2.1.1. Kullanılan benchmark sınıflandırma veri setleri

Bu bölümde kullanılan benchmark sınıflandırma veri setleri, Bölüm 3.5'te verilen veri setleri olup temel özellikleri Çizelge 3.2'te verilmiştir. Sınıflandırma deneylerinde, veri setleri sınıf dağılımları mümkün olduğunca korunacak şekilde %66 eğitim ve %34 test alt kümelerine ayrılmıştır (Singh vd., 2022). Ön işleme aşamasında, tüm veri setlerine min-maks normalleştirme uygulanarak özniteliklerin ortak bir ölçeğe taşınması sağlanmış ve yöntemlerin karşılaştırılabilirliği artırılmıştır. Min-maks normalleştirme, her bir öznitelik için Denklem 4.7'de tanımlandığı şekilde hesaplanır:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (4.7)$$

Burada x ham öznitelik değerini, $x_{\min}$ ve $x_{\max}$ ise ilgili öz niteliğin veri setindeki minimum ve maksimum değerlerini temsil etmektedir.

4.2.1.2. Optimizasyon algoritmalarının parametreleri

Karşılaştırmaya dahil edilen tüm algoritmalar için kullanılan parametreler Çizelge 4.7'de verilmiştir. Sonuçların rassallık etkilerinden arındırılarak raporlanabilmesi için her algoritma 250 iterasyon boyunca 10 bağımsız çalıştırma ile değerlendirilmiştir.

4.2.1.3. Gboa'nın benchmark veri setleri üzerindeki sonuçları

Benchmark veri setleri üzerindeki doğruluk sonuçları, her algoritma için 10 bağımsız çalıştırmadan türetilen ortalama (AVG), standart sapma (STD) ve en iyi (Best)

Çizelge 4.7. Optimizasyon algoritmaları için parametreler

Algoritma	Parametre	Değer
BP	Öğrenme oranı, Momentum	0,001, 0,9
GA	Çaprazlama olasılığı, Mutasyon olasılığı, Seçim mekanizması	0,9, 0,1, Rulet tekerleği
DE	Çaprazlama Olasılığı, Diferansiyel ağırlık	0,9, 0,5
PSO	İvme sabitleri, Atalet ağırlıkları	2,1, 2,1, 0,9, 0,6
GBOA	Kosinüs benzerliği	0,9
ACO	Başlangıç feromonu, Feromon güncelleme sabiti (Q), Feromon sabiti (q), Global feromon bozunma oranı (Pg), Yerel feromon bozunma oranı (pt), Feromon duyarlılığı (α), Görünürlük duyarlılığı (β)	1e-06, 20, 1, 0,9, 0,5, 1, 5
ES	λ, σ	10, 1
PBIL	Öğrenme oranı, İyi popülasyon üyesi, Kötü popülasyon üyesi, Elitizm parametresi, Mutasyon olasılığı	0,05, 1, 0, 1, 0,1

değerler şeklinde Çizelge 4.8’de verilmiştir. Nicel bulgular, GBOA ile eğitilmiş ve ayrıntıları bölüm 3.4.4’de verilen DNN’nin 15 veri setinin tamamında hem AVG hem de Best ölçütlerinde en yüksek değerleri ürettiğini göstermektedir. Kararlılık açısından değerlendirildiğinde, GBOA 15 veri setinin 12’sinde en düşük STD değerine sahip olup, sonuçların çalıştırmalar arasında daha az dalgalandığını ortaya koymaktadır.

Performans farkı özellikle bazı veri setlerinde belirginleşmektedir: Wine veri setinde GBOA için AVG=0,9948 elde edilirken en yakın yöntem için AVG=0,8894 raporlanmıştır; Chess veri setinde GBOA AVG değeri 0,9128 iken en yakın yöntem 0,8088’de kalmıştır; Parkinson veri setinde ise GBOA AVG=0,9546 ve en yakın yöntem 0,8507 olarak ölçülmüştür. Friedman sıralama testi sonuçları da bu eğilimi doğrulamakta; GBOA ortalama sıralama değeri 1,0 ile birinci sırada yer alırken, ikinci sıradaki yöntem BOA için ortalama sıralama değeri 3,07 olarak verilmiştir. Bu nicel çizelge, kosinüs benzerliği tabanlı arama mekanizmasının aynı iterasyon sayısı altında daha yüksek çözüm kalitesi ve daha tutarlı sonuçlar ürettiğini göstermektedir.

4.2.2. S-det sisteminin çilek olgunluk tespiti performansı

Benchmark veri setlerinde başarılı sınıflandırma sonuçları elde edildikten sonra, GBOA-DNN yöntemi ikinci aşama deneysel çalışmada StrawTex veri seti kullanılarak

Çizelge 4.8. GBOA'nın kapsamlı deneysel sonuçları

Veri Seti	Metrik	BOA	ACO	BP	DE	ES	GA	PBIL	PSO	GBOA
Australian	AVG	0,8535	0,7724	0,7794	0,8232	0,8096	0,8289	0,8355	0,8241	0,9093
	STD	0,0159	0,0474	0,0528	0,0311	0,0297	0,0228	0,0166	0,0255	0,0040
	Best	0,8772	0,8421	0,8553	0,8816	0,8509	0,8553	0,8553	0,8596	0,9106
Blood	AVG	0,7867	0,7651	0,6349	0,7718	0,7835	0,7827	0,7812	0,7792	0,8117
	STD	0,0059	0,0119	0,2165	0,0045	0,0090	0,0089	0,0058	0,0096	0,0028
	Best	0,7961	0,7765	0,7804	0,7765	0,7922	0,7961	0,7882	0,7961	0,8133
Breast cancer	AVG	0,9731	0,9206	0,8500	0,9605	0,9605	0,9706	0,9702	0,9685	0,9888
	STD	0,0063	0,0391	0,1020	0,0095	0,0103	0,0079	0,0085	0,0057	0,0000
	Best	0,9832	0,9622	0,9706	0,9706	0,9748	0,9832	0,9832	0,9748	0,9888
Chess	AVG	0,7283	0,6128	0,6713	0,6695	0,6896	0,8088	0,7733	0,7160	0,9128
	STD	0,0512	0,0231	0,0471	0,0289	0,0183	0,0238	0,0196	0,0218	0,0212
	Best	0,8068	0,6440	0,7259	0,7259	0,7167	0,8500	0,8040	0,7479	0,9331
Credit	AVG	0,6991	0,6918	0,6933	0,7018	0,7061	0,7133	0,7124	0,7100	0,7799
	STD	0,0212	0,0219	0,0196	0,0261	0,0263	0,0200	0,0270	0,0132	0,0274
	Best	0,7364	0,7121	0,7273	0,7303	0,7884	0,7545	0,7333	0,7364	0,7966
Diabetes	AVG	0,7584	0,6679	0,5660	0,7115	0,7156	0,7504	0,7366	0,7481	0,8150
	STD	0,0139	0,0385	0,1469	0,0290	0,0233	0,0169	0,0208	0,0307	0,0065
	Best	0,7786	0,7557	0,6908	0,7519	0,7519	0,7748	0,7634	0,7977	0,8241
Hepatitis	AVG	0,8717	0,8472	0,7509	0,8528	0,8453	0,8623	0,8491	0,8434	0,9663
	STD	0,0318	0,0392	0,1996	0,0318	0,0306	0,0252	0,0267	0,0378	0,0083
	Best	0,9057	0,8868	0,8491	0,9057	0,8868	0,9057	0,8868	0,8868	0,9702
Ionosphere	AVG	0,7942	0,7067	0,7367	0,7767	0,7358	0,8025	0,7825	0,7600	0,8816
	STD	0,0429	0,0484	0,0385	0,0340	0,0281	0,0157	0,0240	0,0242	0,0241
	Best	0,8667	0,8000	0,7833	0,8417	0,7917	0,8250	0,8333	0,7917	0,9500
Liver	AVG	0,6958	0,5525	0,5407	0,5653	0,6347	0,6780	0,6636	0,6703	0,7666
	STD	0,0284	0,0639	0,0573	0,0727	0,0535	0,0524	0,0461	0,0263	0,0041
	Best	0,7373	0,6356	0,6525	0,6695	0,7458	0,7373	0,7203	0,6356	0,7692
Parkinson	AVG	0,8358	0,7642	0,7582	0,8239	0,8254	0,8507	0,8388	0,8463	0,9546
	STD	0,0392	0,0421	0,1556	0,0384	0,0264	0,0233	0,0336	0,0345	0,0121
	Best	0,8955	0,8209	0,8657	0,8806	0,8657	0,8806	0,8806	0,8955	0,9687
Ring	AVG	0,7729	0,6233	0,5502	0,6719	0,6998	0,7211	0,7393	0,7091	0,8270
	STD	0,0084	0,0338	0,0513	0,0230	0,0248	0,0328	0,0176	0,0118	0,0106
	Best	0,7830	0,6717	0,6526	0,7075	0,7333	0,7738	0,7655	0,7266	0,8304
Seed	AVG	0,8986	0,5444	0,7986	0,6347	0,7930	0,8931	0,8583	0,7903	0,9554
	STD	0,0208	0,1387	0,1277	0,0747	0,0619	0,0208	0,0375	0,0580	0,0296
	Best	0,9306	0,7500	0,9167	0,7361	0,8889	0,9167	0,9028	0,8889	0,9708
Tic-tac-toe	AVG	0,6733	0,6077	0,5666	0,6215	0,6383	0,6353	0,6626	0,6377	0,7647
	STD	0,0112	0,0347	0,0441	0,0250	0,0313	0,0271	0,0206	0,0209	0,0067
	Best	0,6840	0,6810	0,6258	0,6564	0,6748	0,6687	0,6902	0,6595	0,7668
Titanic	AVG	0,7617	0,7610	0,7377	0,7622	0,7656	0,7625	0,7676	0,7605	0,7882
	STD	0,0026	0,0086	0,0387	0,0072	0,0075	0,0029	0,0047	0,0049	0,0006
	Best	0,7690	0,7677	0,7690	0,7690	0,7717	0,7677	0,7770	0,7677	0,7887
Wine	AVG	0,8894	0,6803	0,7697	0,7576	0,7515	0,8894	0,8667	0,8227	0,9948
	STD	0,0335	0,1098	0,1153	0,0763	0,0436	0,0580	0,0557	0,0474	0,0060
	Best	0,9545	0,8181	0,9091	0,8788	0,8333	0,9394	0,9091	0,8939	1,0000
Friedman Ortalama Sıralama		3,07	8,2	8,13	6,53	5,8	3,13	3,8	5,33	1,0
Sıralama		2	9	8	7	6	3	4	5	1

çilek tespiti için kullanılmıştır. Veri ön işleme aşamasında, StrawTex veri setinin öznitelik vektörleri önerilen GBOA-OK öznitelik çıkarımı yaklaşımı kullanılarak elde edilmiştir. Sonuç olarak, girdi miktarındaki azalma, sınıflandırma süreci için daha hızlı eğitime neden olmaktadır.

4.2.2.1. Strawtex veri seti

Bu bölümde kullanılan StrawTex veri seti ve veri setine uygulanan ön işleme ile veri artırma adımları Bölüm 3.5’de anlatılmıştır.

4.2.2.2. S-det sisteminin sınıflandırma sonuçları

S-Det sisteminin StrawTex veri seti üzerindeki sınıflandırma performansı, üç yöntem grubu altında değerlendirilmiştir: (i) klasik sınıflandırıcılar (KNN, SVM, RF), (ii) metasezgisel algoritmalar ile eğitilmiş DNN tabanlı yaklaşımlar (BP-DNN, ABC-DNN, GA-DNN, PSO-DNN, BOA-DNN ve GBOA-DNN) ve (iii) önceden eğitilmiş derin öğrenme modelleri (VGG16, DenseNet201 ve ResNet50). Çizelge 4.9’da 10 katlı çapraz doğrulama ile elde edilen sonuçlar verilmiştir. Nicel olarak, GBOA-DNN en yüksek doğruluk ($ACU=0,9823$) ve $F1=0,9824$ ile birinci sırada yer almakta; aynı zamanda $FPR=0,0$ ve $FNR=0,0008$ değerleriyle en düşük hata oranlarını üretmektedir. İkinci en iyi sonuç ResNet50 ile $ACU=0,9773$ ve $F1=0,9770$ olarak elde edilirken, BOA-DNN $ACU=0,9772$ ve $F1=0,9771$ değerleriyle yakın bir performans göstermektedir. Klasik sınıflandırıcılar ise $ACU=0,9206-0,9412$ ve $F1=0,9206-0,9411$ aralığında kalarak DNN tabanlı yöntemlerin gerisinde kalmıştır.

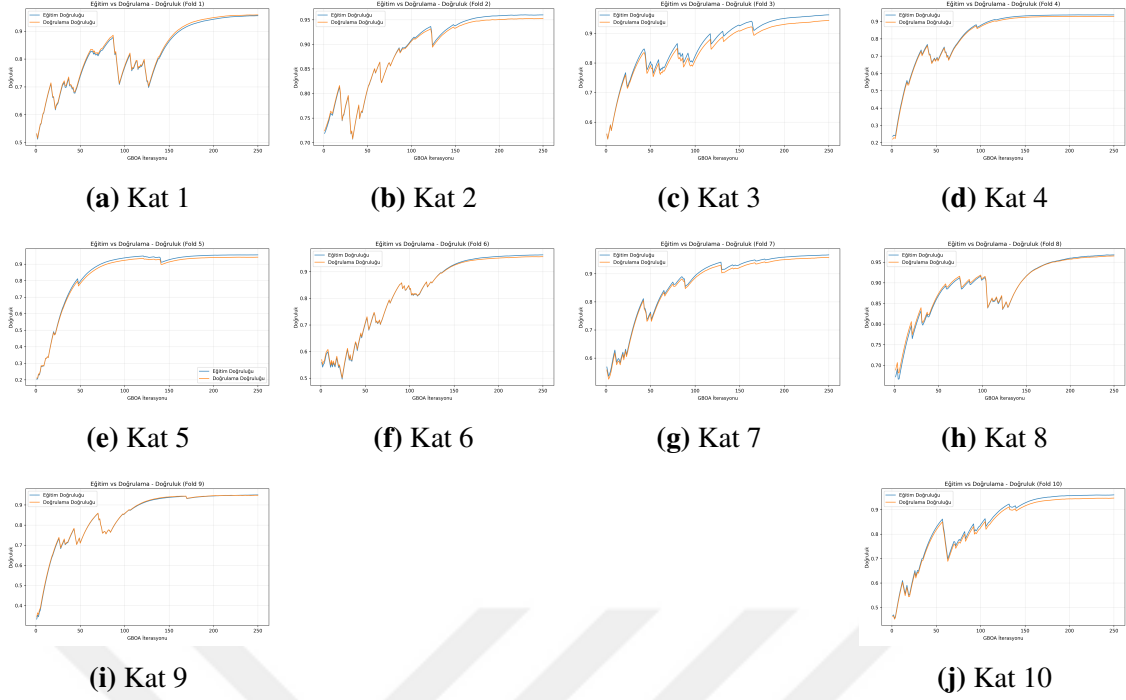
Çizelge 4.9. StrawTex veri seti için karşılaştırmalı test sonuçları

Sınıflandırıcı	Parametreler	Performans Değerlendirme Metrikleri						
		ACU	PRE	REC	FPR	FNR	F1	Sıra
KNN	k: 3	0,9217	0,9235	0,9217	0,0196	0,0786	0,9215	11
SVM	Kernel: linear	0,9206	0,9213	0,9206	0,0199	0,0788	0,9206	12
RF	Estimators: 300	0,9412	0,9420	0,9412	0,0146	0,0587	0,9411	10
BP-DNN	Epok: 250, Aktivasyon: Softplus, Öğrenme Oranı: 0,001, Kayıp: SCCE, Optimizasyon Algoritması: Adam	0,9506	0,9515	0,9506	0,0123	0,0497	0,9504	9
ABC-DNN	Epok: 250, Popülasyon: 10, n_limits:5	0,9633	0,9644	0,9641	0,0149	0,0103	0,9635	7
GA-DNN	Epok: 250, Popülasyon: 10, Seçim: Turnuva, Çaprazlama Olasılığı: 0,9, Mutasyon Olasılığı: 0,05	0,9636	0,9644	0,9640	0,0142	0,0095	0,9635	6
PSO-DNN	Epok: 250, Popülasyon: 10, c1: 2,1, c2: 2,1, w_min: 0,4, w_max: 0,9	0,9762	0,9766	0,9764	0,0	0,0017	0,9762	4
BOA-DNN	Epok: 250, Popülasyon: 10	0,9772	0,9774	0,9773	0,0	0,0012	0,9771	3
VGG16	Epok: 250, Aktivasyon: Sigmoid, Öğrenme Oranı: 0,001, Ağırlıklar: ImageNet, Kayıp: SCCE, Optimizasyon Algoritması: Adam	0,9528	0,9351	0,9409	0,0149	0,0601	0,0337	8
DenseNet201	Epok: 250, Aktivasyon: Softplus, Öğrenme Oranı: 0,001, Ağırlıklar: ImageNet, Kayıp: SCCE, Optimizasyon Algoritması: Adam	0,9757	0,9778	0,9757	0,0061	0,0243	0,9749	5
ResNet50	Epok: 250, Aktivasyon: Softplus, Öğrenme Oranı: 0,001, Ağırlıklar: ImageNet, Kayıp: SCCE, Optimizasyon Algoritması: Adam	0,9773	0,9793	0,9773	0,0057	0,0225	0,9770	2
GBOA-DNN	Epok: 250, Popülasyon: 10, Kosinüs benzerliği: 0,9	0,9823	0,9825	0,9826	0,0	0,0008	0,9824	1

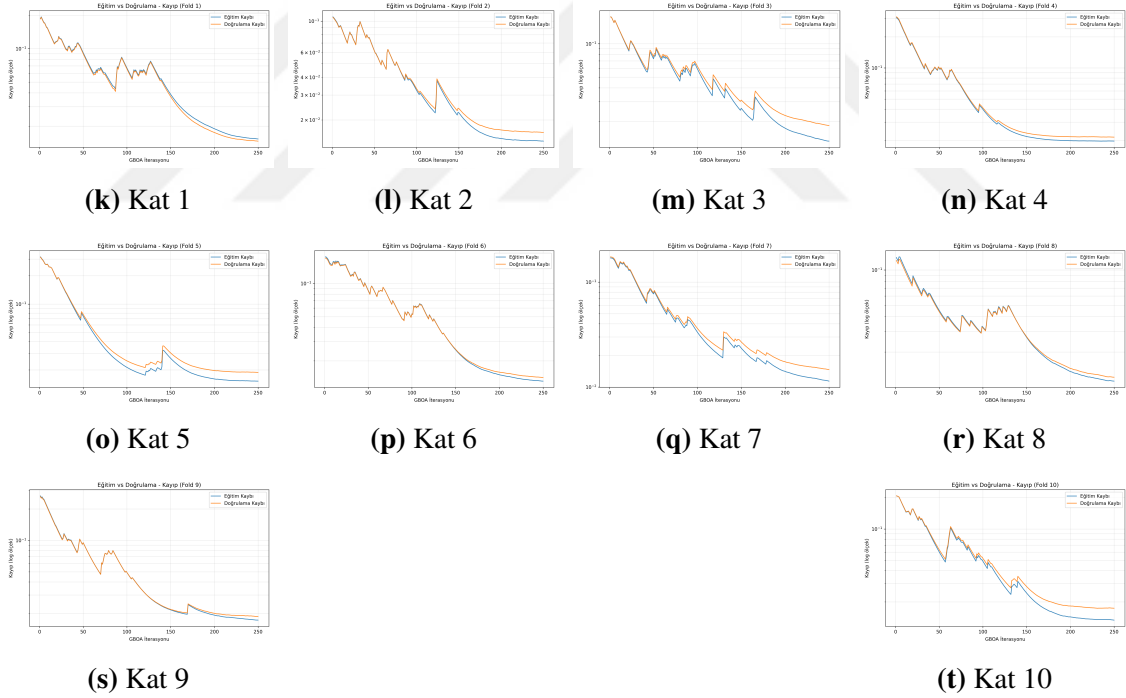
KNN, SVM ve RF için parametre seçimi yapılmış ve en iyi sonuçları üreten ayarlar Çizelge 4.9’da verilmiştir. DNN tabanlı yaklaşımlarda girdi katmanı 96 nöron, gizli katman 193 nöron ve çıktı katmanı 5 nöron içerecek şekilde yapılandırılmıştır. Çizelge 4.9’daki sonuçlar birlikte değerlendirildiğinde, GBOA ile ağırlık optimizasyonu yapılan DNN’nin hem doğruluk hem de hata oranları açısından tutarlı bir üstünlük sağladığı; metasezgisel tabanlı diğer DNN yaklaşımlarının ve önceden eğitilmiş modellerin ise birbirine yakın, ancak daha düşük performans değerlerinde kümelendiği görülmektedir. Bu bulgu, GBOA tabanlı optimizasyonun sınıflandırma performansını artırırken yanlış sınıflandırma oranlarını da belirgin biçimde düşürdüğünü göstermektedir.

Şekil 4.2 önerilen GBOA-DNN’nin performansını kapsamlı bir şekilde analiz eden yakınsama grafiklerini göstermekte ve GBOA tabanlı yaklaşımda aşırı uyum veya yetersiz uyum olmadığını açıkça belirtmektedir.

Doğruluk Grafikleri



Kayıp Grafikleri



Şekil 4.2. Önerilen yöntem için 10 katlı çapraz doğrulama sonuçları

4.2.3. S-det sisteminin diğer görüntü sınıflandırma problemlerindeki performansı

Bu aşamada kullanılan benchmark görüntü veri setleri (Cassava, CIFAR-10 ve MNIST) Bölüm 3.5’de tanıtılmış; temel özellikleri Çizelge 3.3’te verilmiştir. Çizelge 4.10 önerilen S-DET sistemi ile literatürdeki önceki çalışmalar arasındaki

karşılaştırmalı sonuçları sunmaktadır.

Çizelge 4.10. S-DET literatür karşılaştırması

Veri Seti	Sınıflandırıcı	ACU	PRE	REC	F1
Cassava	Hybrid dual-path Network (Karthik vd., 2023)	0,9066	0,8865	0,8450	0,8645
	Deep Gaussian CNN (Ahishakiye vd., 2024)	0,9010	0,8830	0,8960	0,8790
	Önerilen S-DET	0,9183	0,8855	0,8731	0,8794
CIFAR-10	LSTM (Pereira-Ferrero vd., 2023)	0,9202	0,9171	0,8946	0,9154
	H-CapsNet (Noor ve Robles-Kelly, 2024)	0,9141	0,9139	0,9141	0,9138
	Önerilen S-DET	0,9395	0,9151	0,8931	0,9039
MNIST	MQCNN (Hassan vd., 2024)	0,9909	0,9868	0,9887	0,9901
	DNN (Kabir vd., 2024)	0,9964	0,9966	0,9965	0,9965
	Önerilen S-DET	0,9872	0,9861	0,9774	0,9817

Çizelge 4.10 sonuçları, önerilen S-DET yaklaşımının farklı zorluk düzeylerine sahip benchmark veri setlerinde farklılaşan bir performans profiline sahip olduğunu göstermektedir. Cassava veri setinde S-DET, ACU=0,9183 ve F1=0,8794 ile karşılaştırmaya dahil edilen yöntemler arasında en yüksek doğruluk ve F1-skorunu üretmiştir; aynı deneyde PRE=0,8855 ve REC=0,8731 değerleri elde edilmiş olup, kesinlik bakımından en yüksek literatür değeri 0,8865 ve duyarlılık bakımından en yüksek literatür değeri 0,8960 olarak raporlanmıştır. CIFAR-10 veri setinde S-DET, ACU=0,9395 ile en yüksek doğruluğu sağlamış; buna karşılık PRE=0,9151, REC=0,8931 ve F1=0,9039 değerleri ile sınıf-temelli ölçütlerde LSTM (F1=0,9154) ve H-CapsNet (F1=0,9138) sonuçlarının gerisinde kalmıştır. MNIST veri setinde ise S-DET, ACU=0,9872 ve F1=0,9817 değerleri elde etmiş; aynı çizelgede DNN (Kabir vd., 2024) tarafından ACU=0,9964 ve F1=0,9965 değerleri raporlanmıştır. Bu bulgular birlikte değerlendirildiğinde, S-DET'nin özellikle karmaşık ve sınıf içi varyansı yüksek doğal görüntü veri setlerinde rekabetçi sonuçlar üretebildiği; ancak MNIST gibi daha basit ve yüksek doğruluk tavanına sahip görevlerde literatürdeki özel amaçlı yaklaşımlara göre daha düşük değerler verdiği görülmektedir.

5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Bu bölümde, deneysel sonuçlar kapsamlı bir şekilde yorumlanmış ve tartışılmıştır. Metodolojik yaklaşımların güçlü ve zayıf yönleri, elde edilen başarıların nedenleri, karşılaşılan sınırlamalar ve bunların olası çözüm yolları ele alınmıştır. Ayrıca, çalışmanın literatüre katkıları ve pratik uygulamaları üzerine değerlendirmeler yapılmıştır.

5.1. BOA-GA Yönteminin Sonuçlarının Yorumlanması ve Tartışılması

5.1.1. Boa tabanlı kenar tespiti yönteminin başarısının analizi

BOA tabanlı kenar tespiti yöntemi, kenar tespit edicilere (Canny, Laplacian, Roberts, Sobel) kıyasla daha dengeli ve üstün performans sergilemiştir. Çizelge 4.3'te görüldüğü üzere, önerilen yöntem %96,8 doğruluk ve 0,920 F1-skoru ile en yüksek performansı elde etmiştir. Bu başarının temel nedenleri şunlardır:

Güçlü Yönler:

- **Adaptif Kernel Öğrenimi:** BOA, görüntü içeriğine ve referans kenar görüntülerine göre adaptif olarak optimum kernel filtresini öğrenmektedir. Bu yaklaşım, sabit kernel yapılarına sahip yöntemlerin aksine, her veri seti için özelleştirilmiş kenar tespiti gerçekleştirmektedir.
- **Çok Katmanlı Filtre Entegrasyonu:** Farklı σ değerlerine sahip Gaussian filtrelerinin çok katmanlı yapısı ve optimum kernel filtresinin entegrasyonu, hem düşük hem de yüksek frekanslı kenar bilgisini etkili bir şekilde yakalamaktadır. Bu sayede, gürültü bastırılırken sürekli ve belirgin kenarlar üretilmektedir.
- **Referans Tabanlı Optimizasyon:** Standart kenar tespiti yöntemlerinin çıktılarının ortalaması alınarak elde edilen referans kenar görüntüleri, farklı yöntemlerin güçlü yönlerini birleştirerek daha dengeli bir referans sağlamaktadır.

Zayıf Yönler ve Sınırlamalar:

- **Hesaplama Karmaşıklığı:** BOA optimizasyon süreci, kenar tespiti yöntemlerine kıyasla daha fazla hesaplama kaynağı gerektirmektedir. Özellikle büyük görüntüler için optimizasyon süresi artmaktadır.

- **Parametre Hassasiyeti:** Algoritmanın performansı, popülasyon boyutu, iterasyon sayısı ve kosinüs benzerliği gibi parametrelere bağlıdır. Bu parametrelerin uygun seçilmemesi durumunda, optimizasyon süreci yerel minimumlarda takılabilir.
- **Referans Görüntü Bağımlılığı:** Yöntemin başarısı, referans kenar görüntülerinin kalitesine bağlıdır. Düşük kaliteli referans görüntüler, optimizasyon sürecini olumsuz etkileyebilir.

Olası Çözüm Yolları:

- Hesaplama karmaşıklığını azaltmak için, paralel hesaplama teknikleri ve GPU hızlandırması kullanılabilir. Ayrıca, görüntü ön işleme aşamasında görüntü boyutunun uygun şekilde ölçeklendirilmesi optimizasyon süresini kısaltabilir.
- Parametre hassasiyetini azaltmak için, adaptif parametre ayarlama mekanizmaları veya meta-optimizasyon teknikleri geliştirilebilir. Ayrıca, farklı veri setleri için önceden optimize edilmiş parametre setleri oluşturulabilir.
- Referans görüntü kalitesini artırmak için, daha gelişmiş kenar tespiti yöntemlerinin kombinasyonu veya derin öğrenme tabanlı kenar tespiti yöntemleri kullanılabilir.

5.1.2. Boa-ga yönteminin dengesiz veri setlerindeki performans analizi

BOA-GA yöntemi, beş farklı dengesiz veri seti üzerinde test edilmiş ve tüm veri setlerinde tutarlı performans iyileştirmeleri sağlamıştır. Özellikle HAM10000 veri setinde, dengesiz sürümde doğruluk %74,6 iken, önerilen artırım yöntemi uygulandıktan sonra doğruluk %98'e yükselmiştir.

Güçlü Yönler:

- **Kenar Tabanlı Bilgilendirici Artırım:** BOA-GA yöntemi, optimize edilmiş kenar görüntülerini kullanarak sentetik örnekler üretmektedir. Bu yaklaşım, sadece geometrik dönüşümler (döndürme, öteleme, yakınlaştırma) kullanan artırım yöntemlerinden farklı olarak, görüntünün yapısal özelliklerini (kenar bilgisi) koruyarak daha anlamlı ve ayırt edici sentetik örnekler üretmektedir.
- **Adaptif Aşırı Örnekleme Stratejisi:** Yöntem, veri setinin dengesizlik oranına göre adaptif olarak üretilecek sentetik örnek sayısını belirlemektedir. Bu sayede, aşırı çoğaltma önlenirken dengeli bir sınıf dağılımı elde edilmektedir.

- **Açısal Yerleştirme Mekanizması:** Kenar görüntüsünün rastgele seçilen bir açı ile döndürülmesi ve önceden tanımlanmış bir katsayı aralığı kullanılarak orijinal görüntü ile karıştırılması, çeşitliliği artırarak modelin genelleme yeteneğini güçlendirmektedir.

Zayıf Yönler ve Sınırlamalar:

- **Düşük Dengesizlik Oranlarında Sınırlı İyileştirme:** Plant Village gibi daha düşük dengesizlik oranlarına sahip veri setleri için, dengesiz ve dengeli versiyonlar arasındaki performans farkı daha küçüktür. Bu durum, çoğu sınıf için mevcut olan nispeten büyük örnek sayısına atfedilmektedir.
- **Kenar Tespiti Kalitesine Bağımlılık:** Yöntemin başarısı, BOA tabanlı kenar tespiti yönteminin kalitesine doğrudan bağlıdır. Düşük kaliteli kenar görüntüleri, sentetik örneklerin kalitesini olumsuz etkileyebilir.
- **Veri Seti Özelliklerine Bağımlılık:** Yöntem, kenar bilgisinin önemli olduğu görüntü sınıflandırma problemlerinde daha etkilidir. Doku veya renk tabanlı sınıflandırma problemlerinde performans sınırlı olabilir.

Olası Çözüm Yolları:

- Düşük dengesizlik oranlarına sahip veri setleri için, kenar tabanlı artırımın yanı sıra renk ve doku tabanlı artırım teknikleri de entegre edilebilir.
- Kenar tespiti kalitesini artırmak için, derin öğrenme tabanlı kenar tespiti yöntemleri veya çoklu referans görüntü kullanımı düşünülebilir.
- Farklı veri seti özelliklerine uyum sağlamak için, hibrit artırım stratejileri (kenar, doku ve renk tabanlı kombinasyonlar) geliştirilebilir.

5.1.3. Literatür karşılaştırmalarının değerlendirilmesi

BOA-GA yöntemi, HAM10000 veri seti üzerinde literatürdeki önceki çalışmalarla karşılaştırıldığında, en yüksek doğruluk (%98), kesinlik (%98,6) ve F1-skoru (0,986) değerlerini elde etmiştir. Bu başarı, yöntemin kenar tabanlı bilgilendirici artırım stratejisinin etkinliğini göstermektedir.

Literatüre Katkıları:

- **Yeni Artırım Paradigması:** BOA-GA yöntemi, optimizasyon tabanlı kenar tespiti ve kenar tabanlı görüntü artırımı kombinasyonunu literatüre kazandırmıştır. Bu yaklaşım, sadece geometrik dönüşümler kullanan geleneksel artırım yöntemlerinden farklı olarak, görüntünün yapısal özelliklerini koruyarak daha anlamlı sentetik örnekler üretmektedir.
- **Dengesiz Veri Setleri için Etkili Çözüm:** Özellikle yüksek dengesizlik oranlarına sahip veri setlerinde (HAM10000 gibi), yöntem literatürdeki diğer yaklaşımlara kıyasla önemli performans iyileştirmeleri sağlamıştır.
- **Adaptif Aşırı Örneklemeye Stratejisi:** Veri setinin dengesizlik oranına göre adaptif olarak üretilecek sentetik örnek sayısını belirleyen strateji, aşırı çoğaltmayı önlerken dengeli bir sınıf dağılımı elde etmektedir.

5.2. GBOA ve S-Det Sisteminin Sonuçlarının Yorumlanması ve Tartışılması

5.2.1. Gboa'nın optimizasyon performansının analizi

GBOA, 15 farklı benchmark sınıflandırma veri seti üzerinde test edilmiş ve BOA, GA, PSO, ACO, DE, ES, PBIL ve BP gibi optimizasyon tekniklerine kıyasla üstün performans sergilemiştir. Friedman sıralama testi sonuçlarına göre, GBOA ortalama sıralama değeri 1,0 ile birinci sırada yer almıştır.

Başarı Nedenleri:

- **Kosinüs Benzerliği Tabanlı Arama Mekanizması:** GBOA, kosinüs benzerliği kullanarak yeni bir arama mekanizması geliştirmiştir. Bu mekanizma, yerel minimumlardan kaçınma yeteneğini artırarak optimizasyon sürecinin daha iyi çözümlere yakınsamasını sağlamaktadır.
- **Hızlı Yakınsama:** Aynı iterasyon sayısı ile daha iyi sonuçlar elde edilmesi, GBOA'nın optimal değerlere daha hızlı yakınsadığını göstermektedir. Bu durum, algoritmanın keşif ve sömürü dengesini etkili bir şekilde yönetebildiğini göstermektedir.
- **Tutarlılık:** 10 çalıştırmadan sonra daha düşük standart sapma değerleri, elde edilen sonuçların tutarlı olduğunu ve iterasyonlar arasındaki farkların diğer algoritmalara kıyasla minimal olduğunu göstermektedir.

Güçlü Yönler:

- **Genel Amaçlı Uygulanabilirlik:** GBOA, farklı özelliklere sahip veri setleri (küçük, orta, büyük; ikili, çok sınıflı) üzerinde tutarlı performans sergilemektedir.
- **Parametre Basitliği:** Algoritma, sadece kosinüs benzerliği parametresi gerektirmektedir, bu da kullanım kolaylığı sağlamaktadır.
- **BOA'nın Geliştirilmiş Versiyonu:** GBOA, BOA'nın geliştirilmiş bir versiyonu olarak, BOA'nın güçlü yönlerini korurken yeni arama mekanizması ile performansı artırmaktadır.

Zayıf Yönler ve Sınırlamalar:

- **Yüksek Boyutlu Problemler:** Çok yüksek boyutlu optimizasyon problemlerinde, algoritmanın performansı düşebilir. Özellikle binlerce parametre içeren derin sinir ağlarında, optimizasyon süreci uzun sürebilir.
- **Hesaplama Maliyeti:** Kosinüs benzerliği hesaplamaları, özellikle büyük popülasyon boyutlarında ek hesaplama maliyeti getirebilir.
- **Adaptif Parametre Ayarlama Eksikliği:** Algoritma, farklı problem tipleri için adaptif parametre ayarlama mekanizması içermemektedir.

Olası Çözüm Yolları:

- Yüksek boyutlu problemler için, öznitelik seçimi veya boyut azaltma teknikleri kullanılabilir. Ayrıca, mini-batch optimizasyon stratejileri geliştirilebilir.
- Hesaplama maliyetini azaltmak için, kosinüs benzerliği hesaplamaları için hızlı yaklaşımlar veya yaklaşık hesaplama teknikleri kullanılabilir.
- Adaptif parametre ayarlama için, meta-öğrenme teknikleri veya problem özelliklerine göre otomatik parametre seçimi mekanizmaları geliştirilebilir.

5.2.2. S-det sisteminin çilek olgunluk tespiti performansının analizi

S-Det sistemi, StrawTex veri seti üzerinde test edilmiş ve sınıflandırıcılar (KNN, SVM, RF), optimizasyon tabanlı DNN modelleri (BP-DNN, ABC-DNN, GA-DNN, PSO-DNN, BOA-DNN) ve önceden eğitilmiş derin öğrenme modelleri (VGG16,

DenseNet201, ResNet50) ile karşılaştırılmıştır. GBOA-DNN, en yüksek doğruluk (%98,23), kesinlik (%98,25), duyarlılık (%98,26) ve F1-skoru (0,9824) ile en başarılı yöntem olmuştur.

Başarı Nedenleri:

- **GBOA-OK Öznitelik Çıkarımı:** GBOA-OK öznitelik çıkarımı yaklaşımı, görüntülerden etkili doku öznitelikleri çıkarmaktadır. Bu yaklaşım, girdi miktarındaki azalma ile daha hızlı eğitime neden olurken, sınıflandırma performansını da artırmaktadır.
- **GBOA-DNN Optimizasyonu:** GBOA ile eğitilmiş DNN modeli, diğer optimizasyon tekniklerine kıyasla daha iyi ağırlık değerleri bulmaktadır. Bu durum, modelin öğrenme yeteneğini artırarak sınıflandırma performansını iyileştirmektedir.
- **10 Katlı Çapraz Doğrulama:** Yakınsama grafikleri, GBOA-DNN'nin aşırı uyum veya yetersiz uyum göstermediğini ve tutarlı performans sergilediğini göstermektedir.

Güçlü Yönler:

- **Düşük Hata Oranları:** GBOA-DNN, en düşük yanlış pozitif oranı (FPR: 0.0) ve yanlış negatif oranı (FNR: 0.0008) ile en güvenilir sonuçları üretmektedir.
- **Bütünleşik Sistem:** S-Det, tüm süreci tek bir entegre boru hattında sunan bir sistemdir.
- **Pratik Uygulanabilirlik:** Çilek olgunluk tespiti gibi tarımsal uygulamalarda, sistemin yüksek doğruluk oranı pratik kullanım için uygundur.

Sınırlamalar:

- **Veri Seti Özelliklerine Bağımlılık:** Sistem, doku tabanlı öznitelik çıkarımı kullandığı için, doku bilgisinin önemli olduğu problemlerde daha etkilidir. Renk veya şekil tabanlı problemlerde performans sınırlı olabilir.
- **Ön İşleme Gereksinimi:** Sistem, görüntülerin önceden işlenmesi ve öznitelik çıkarımı aşamasını gerektirmektedir. Bu durum, gerçek zamanlı uygulamalarda gecikme yaratabilir.

- **Model Karmaşıklığı:** GBOA-OK öznitelik çıkarımı ve GBOA-DNN optimizasyonu, sistemin toplam karmaşıklığını artırmaktadır.

Çözüm Önerileri:

- Farklı problem tipleri için, hibrit öznitelik çıkarımı yaklaşımları (doku, renk, şekil kombinasyonları) geliştirilebilir.
- Gerçek zamanlı uygulamalar için, öznitelik çıkarımı ve optimizasyon süreçlerini hızlandırmak amacıyla GPU hızlandırması veya model sıkıştırma teknikleri kullanılabilir.
- Model karmaşıklığını azaltmak için, öznitelik seçimi veya boyut azaltma teknikleri uygulanabilir.

5.2.3. S-det sisteminin benchmark görüntü veri setlerindeki performans analizi

S-Det sistemi, Cassava, CIFAR-10 ve MNIST benchmark görüntü veri setleri üzerinde test edilmiş ve literatürdeki önceki çalışmalarla karşılaştırılmıştır. Sistem, Cassava veri setinde en yüksek doğruluk (%91,83) ve F1-skoru (%87,94) elde etmiştir. CIFAR-10'da en yüksek doğruluk (%93,95) sağlanmış, ancak F1-skoru, duyarlılık ve kesinlik metrikleri diğer yöntemlere kıyasla biraz daha düşük kalmıştır. MNIST veri seti için, sistem umut verici ancak yetersiz sonuçlar vermiştir.

Performans Değerlendirmesi:

- **Cassava Veri Seti:** S-Det sistemi, doku tabanlı öznitelik çıkarımı sayesinde Cassava veri setinde başarılı sonuçlar elde etmiştir. Bu başarı, sistemin tarımsal görüntü sınıflandırma problemlerinde etkili olduğunu göstermektedir.
- **CIFAR-10 Veri Seti:** Sistem, doğruluk açısından en iyi performansı göstermesine rağmen, diğer metriklerde (F1-skoru, duyarlılık, kesinlik) biraz daha düşük sonuçlar vermiştir. Bu durum, CIFAR-10'un karmaşık görüntü içeriği ve doku tabanlı öznitelik çıkarımının bu veri seti için yetersiz kalabileceğini göstermektedir.
- **MNIST Veri Seti:** Sistem, MNIST veri seti için umut verici ancak yetersiz sonuçlar vermiştir. Bu durum, MNIST'in el yazısı rakam görüntüleri içermesi ve doku bilgisinin bu problem için yeterli olmayabileceğini göstermektedir.

Sınırlamalar ve Çözüm Önerileri:

- **Veri Seti Özelliklerine Bağımlılık:** Sistem, doku tabanlı öznitelik çıkarımı kullandığı için, doku bilgisinin önemli olduğu problemlerde daha etkilidir. Farklı veri seti özelliklerine uyum sağlamak için, hibrit öznitelik çıkarımı yaklaşımları geliştirilebilir.
- **Ön İşleme Eksikliği:** Sistem, herhangi bir ön işleme olmadan doku tabanlı öznitelik çıkarımı kullanmaktadır. Görüntü ön işleme teknikleri (normalizasyon, kontrast iyileştirme, gürültü azaltma) entegre edilerek performans artırılabilir.
- **Derin Öğrenme Modelleri ile Entegrasyon:** Sistem, önceden eğitilmiş derin öğrenme modelleri ile doğrudan entegre edilebilir. Bu sayede, hem doku tabanlı öznitelikler hem de derin öğrenme öznitelikleri birleştirilebilir.

5.3. Metodolojik Yaklaşımların Genel Değerlendirmesi

5.3.1. Güçlü yönler

Bu tez çalışmasında geliştirilen metodolojik yaklaşımların güçlü yönleri şunlardır:

1. **Kademeli Geliştirme Yaklaşımı:** Çalışma, birinci aşamada veri artırma problemine odaklanmış, ikinci aşamada ise bütünleşik sistem tasarımına geçmiştir. Bu kademeli yaklaşım, her aşamada elde edilen bilgi ve deneyimin bir sonraki aşamaya aktarılmasını sağlamıştır.
2. **Optimizasyon Tabanlı Yaklaşım:** BOA ve GBOA gibi optimizasyon algoritmalarının kullanılması, problemlere adaptif çözümler üretilmesini sağlamıştır. Bu yaklaşım, sabit yöntemlerin aksine, her problem için özelleştirilmiş çözümler sunmaktadır.
3. **Çoklu Veri Seti Doğrulaması:** Geliştirilen yöntemler, farklı özelliklere sahip çok sayıda veri seti üzerinde test edilmiştir. Bu durum, yöntemlerin genel amaçlı uygulanabilirliğini göstermektedir.

4. **Kapsamlı Karşılaştırmalı Analiz:** Yöntemler, literatürdeki önceki çalışmalar ve çeşitli alternatif yaklaşımlarla kapsamlı bir şekilde karşılaştırılmıştır. Bu durum, yöntemlerin etkinliğini objektif bir şekilde değerlendirmeyi sağlamıştır.
5. **İstatistiksel Güvenilirlik:** Tüm deneysel sonuçlar, çoklu çalıştırmalar ve istatistiksel testler (Friedman sıralama testi gibi) ile desteklenmiştir. Bu durum, sonuçların güvenilirliğini artırmaktadır.

5.3.2. Zayıf yönler ve sınırlamalar

Çalışmada karşılaşılan başlıca sınırlamalar şunlardır:

1. **Hesaplama Karmaşıklığı:** Optimizasyon tabanlı yaklaşımlar, standart yöntemlere kıyasla daha fazla hesaplama kaynağı gerektirmektedir. Özellikle büyük veri setleri ve yüksek boyutlu problemlerde, optimizasyon süresi artmaktadır.
2. **Parametre Hassasiyeti:** Algoritmaların performansı, popülasyon boyutu, iterasyon sayısı ve diğer parametrelere bağlıdır. Bu parametrelerin uygun seçilmemesi durumunda, optimizasyon süreci yerel minimumlarda takılabilir veya yetersiz sonuçlar üretebilir.
3. **Veri Seti Özelliklerine Bağımlılık:** Yöntemler, özellikle doku ve kenar bilgisinin önemli olduğu problemlerde daha etkilidir. Renk veya şekil tabanlı problemlerde performans sınırlı olabilir.
4. **Ölçeklenebilirlik:** Yöntemler, küçük ve orta ölçekli veri setlerinde başarılı sonuçlar vermiştir. Ancak, çok büyük ölçekli veri setleri (milyonlarca örnek) için ölçeklenebilirlik test edilmemiştir.
5. **Gerçek Zamanlı Uygulama:** Sistemler, gerçek zamanlı uygulamalar için optimize edilmemiştir. Özellikle S-Det sistemi, öznetelik çıkarımı ve optimizasyon süreçleri nedeniyle gerçek zamanlı kullanım için gecikme yaratabilir.

5.3.3. Olası çözüm yolları

Yukarıda belirtilen sınırlamalar için önerilen çözüm yolları şunlardır:

1. **Hesaplama Karmaşıklığını Azaltma:**

- Paralel hesaplama teknikleri ve GPU hızlandırması kullanılabilir.
- Görüntü ön işleme aşamasında görüntü boyutunun uygun şekilde ölçeklendirilmesi optimizasyon süresini kısaltabilir.
- Mini-batch optimizasyon stratejileri geliştirilebilir.

2. Parametre Hassasiyetini Azaltma:

- Adaptif parametre ayarlama mekanizmaları veya meta-optimizasyon teknikleri geliştirilebilir.
- Farklı veri setleri için önceden optimize edilmiş parametre setleri oluşturulabilir.
- Problem özelliklerine göre otomatik parametre seçimi mekanizmaları geliştirilebilir.

3. Veri Seti Özelliklerine Uyum:

- Hibrit öznitelik çıkarımı yaklaşımları (doku, renk, şekil kombinasyonları) geliştirilebilir.
- Farklı problem tipleri için özelleştirilmiş modüller geliştirilebilir.
- Transfer öğrenme teknikleri kullanılarak farklı problem tiplerine uyum sağlanabilir.

4. Ölçeklenebilirlik:

- Büyük ölçekli veri setleri için dağıtık hesaplama teknikleri kullanılabilir.
- Veri seti örnekleme teknikleri ile temsili alt kümeler oluşturulabilir.
- Artımlı öğrenme stratejileri geliştirilebilir.

5. Gerçek Zamanlı Uygulama:

- Öznitelik çıkarımı ve optimizasyon süreçlerini hızlandırmak amacıyla model sıkıştırma teknikleri kullanılabilir.
- Önceden eğitilmiş modeller kullanılarak optimizasyon süreci atlanabilir veya kısaltılabilir.
- Edge computing teknolojileri kullanılarak sistem gerçek zamanlı uygulamalara uyarlanabilir.

5.4. Çalışmanın Literatüre Katkıları

Bu tez çalışması, görüntü sınıflandırma ve nesne tespiti problemlerine yönelik önemli katkılar sunmaktadır:

5.4.1. Birinci aşama katkıları

1. **Optimizasyon Tabanlı Kenar Tespiti:** BOA kullanılarak görüntü içeriğine adaptif optimum kernel filtresi öğrenme yaklaşımı literatüre kazandırılmıştır. Bu yaklaşım, sabit kernel yapılarına sahip kenar tespiti yöntemlerinden farklı olarak, her veri seti için özelleştirilmiş kenar tespiti gerçekleştirmektedir.
2. **Kenar Tabanlı Görüntü Artırımı:** Optimize edilmiş kenar görüntülerini kullanarak sentetik örnekler üreten yeni bir görüntü artırımı stratejisi geliştirilmiştir. Bu strateji, sadece geometrik dönüşümler kullanan geleneksel artırım yöntemlerinden farklı olarak, görüntünün yapısal özelliklerini koruyarak daha anlamlı sentetik örnekler üretmektedir.
3. **Adaptif Aşırı Örnekleme Stratejisi:** Veri setinin dengesizlik oranına göre adaptif olarak üretilecek sentetik örnek sayısını belirleyen strateji, aşırı çoğaltmayı önlerken dengeli bir sınıf dağılımı elde etmektedir.
4. **Dengesiz Veri Setleri için Etkili Çözüm:** BOA-GA yöntemi, özellikle yüksek dengesizlik oranlarına sahip veri setlerinde (HAM10000 gibi), literatürdeki diğer yaklaşımlara kıyasla önemli performans iyileştirmeleri sağlamıştır.

5.4.2. İkinci aşama katkıları

1. **GBOA:** BOA'nın kosinüs benzerliği tabanlı yeni arama mekanizması ile geliştirilmiş versiyonu literatüre kazandırılmıştır. GBOA, 15 farklı benchmark veri seti üzerinde diğer optimizasyon tekniklerine kıyasla üstün performans sergilemiştir.
2. **GBOA Tabanlı Optimum Kernel Öz nitelik Çıkarımı:** GBOA kullanılarak görüntülerden etkili doku öz nitelikleri çıkaran yeni bir öz nitelik çıkarımı yaklaşımı geliştirilmiştir. Bu yaklaşım, girdi miktarındaki azalma ile daha hızlı eğitime neden olurken, sınıflandırma performansını da artırmaktadır.

3. **Bütünleşik S-Det Sistemi:** Öznitelik çıkarımından sınıflandırmaya kadar tüm süreci kapsayan entegre bir sistem geliştirilmiştir. Sistem, çilek olgunluk tespiti gibi pratik uygulamalarda yüksek doğruluk oranları elde etmiştir.
4. **Kademeli Yaklaşım:** Çalışma, veri artırma probleminden bütünleşik sistem tasarımına kadar kademeli bir geliştirme yaklaşımı sunmaktadır. Bu yaklaşım, her aşamada elde edilen bilgi ve deneyimin bir sonraki aşamaya aktarılmasını sağlamaktadır.

5.5. Pratik Uygulamalar

Bu tez çalışmasında geliştirilen yöntemler, aşağıdaki pratik uygulama alanlarında kullanılabilir:

1. **Tıbbi Görüntü Analizi:** BOA-GA yöntemi, özellikle dengesiz tıbbi görüntü veri setlerinde (HAM10000 gibi) etkili sonuçlar vermiştir. Bu yöntem, deri lezyonu sınıflandırma, radyoloji görüntü analizi ve diğer tıbbi görüntü sınıflandırma problemlerinde kullanılabilir.
2. **Tarımsal Uygulamalar:** S-Det sistemi, çilek olgunluk tespiti gibi tarımsal uygulamalarda başarılı sonuçlar vermiştir. Sistem, meyve olgunluk tespiti, bitki hastalığı tespiti ve diğer tarımsal görüntü analizi problemlerinde kullanılabilir.
3. **Endüstriyel Kalite Kontrol:** GBOA tabanlı optimizasyon yaklaşımları, endüstriyel kalite kontrol sistemlerinde ürün sınıflandırma ve hata tespiti problemlerinde kullanılabilir.
4. **Güvenlik ve Gözetleme:** Geliştirilen yöntemler, güvenlik kamerası görüntülerinde nesne tespiti ve sınıflandırma problemlerinde kullanılabilir.
5. **Otonom Sistemler:** S-Det sistemi, otonom araçlar ve robotik sistemlerde görüntü sınıflandırma ve nesne tespiti problemlerinde kullanılabilir.

6. SONUÇ VE ÖNERİLER

Bu bölümde, tez çalışmasının ana bulguları özetlenmiş, elde edilen sonuçların bilimsel ve pratik anlamları vurgulanmış, çalışmanın sınırlılıkları belirtilmiş ve gelecekteki araştırmalar için öneriler sunulmuştur.

6.1. Ana Bulguların Özeti

6.1.1. Birinci aşama bulguları

Birinci aşamada BOA-GA yöntemi, optimizasyon tabanlı kenar tespiti ile kenar tabanlı görüntü artırımını birleştirerek dengesiz veri setlerinde tutarlı iyileşmeler sağlamıştır (Çizelge 4.3, 4.4). HAM10000 üzerinde literatür karşılaştırmasında en yüksek performans elde edilmiş (Çizelge 4.6), özellikle yüksek dengesizlik oranlarında yapısal bilgiyi koruyan sentetik örnekler etkili olmuştur.

6.1.2. İkinci aşama bulguları

İkinci aşamada GBOA, 15 benchmark veri setinde Friedman sıralamasında birinci olmuş (Çizelge 4.8); S-Det ise StrawTex üzerinde en yüksek doğruluğu sağlamıştır (Çizelge 4.9). GBOA-OK öznitelik çıkarımı daha hızlı eğitim ve yüksek sınıflandırma performansı sağlamış; Cassava, CIFAR-10 ve MNIST'ta literatürle karşılaştırmalı sonuçlar Çizelge 4.10'da ve yakınsama davranışı Şekil 4.2'de özetlenmiştir.

6.2. Bilimsel ve Pratik Anlamlar

6.2.1. Bilimsel katkılar

Çalışma, optimizasyon algoritmalarının veri artırma, öznitelik çıkarımı ve model eğitimi aşamalarında koordineli kullanımını göstererek metasezgisel-derin öğrenme kapsamını genişletmiştir. BOA-GA kenar tabanlı bilgilendirici artırım paradigması sunarken; BOA ile öğrenilen adaptif kernel, her veri seti için özelleştirilmiş kenar tespiti sağlamaktadır. GBOA, kosinüs benzerliği tabanlı arama ile daha etkili optimizasyon performansı sunmakta; S-Det ise öznitelik çıkarımından sınıflandırmaya entegre sistem

tasarımı örneği oluşturmaktadır. Veri artırımından bütünleşik sisteme kademeli metodolojik yaklaşım, her aşamadaki deneyimin sonrakine aktarılmasını sağlamıştır.

6.2.2. Pratik uygulamalar ve etkiler

Başlıca uygulama alanları Bölüm 5.5'te özetlenmiştir: tıbbi görüntü analizi (dengesiz veri setleri, erken teşhis), tarımsal otomasyon (meyve olgunluk ve hastalık tespiti), endüstriyel kalite kontrol, güvenlik ve otonom sistemler. Bu alanlarda ekonomik verimlilik ve sosyal refah açısından katkı potansiyeli bulunmaktadır.

6.3. Çalışmanın Sınırlılıkları

6.3.1. Metodolojik sınırlılıklar

Optimizasyon tabanlı yaklaşımlar daha fazla hesaplama gerektirir; büyük görüntüler ve yüksek boyutlu problemlerde süre artar ve gerçek zamanlı uygulamalarda gecikme oluşabilir. Performans popülasyon boyutu, iterasyon sayısı ve kosinüs benzerliği gibi parametrelere bağlıdır; uygun seçilmezse yerel minimum riski vardır. Yöntemler doku ve kenar bilgisinin önemli olduğu problemlerde daha etkili olup, renk/şekil tabanlı görevlerde sınırlı kalabilir. Çok büyük ölçekli veri setleri test edilmemiştir. BOA tabanlı kenar tespiti referans görüntü kalitesine bağlıdır.

6.3.2. Deneysel sınırlılıklar

Veri setleri tıbbi ve tarımsal alanlara odaklı olup doğal görüntü, uydu görüntüsü vb. üzerinde test edilmemiştir. Sistemler gerçek zamanlı kullanım için optimize edilmemiş ve gerçek zamanlı testler yapılmamıştır. Deneyler belirli donanım konfigürasyonunda gerçekleştirilmiş; literatürdeki tüm alternatiflerle karşılaştırma yapılamamıştır.

6.3.3. Teorik sınırlılıklar

Metasezgisel algoritmalar yakınsama garantisi vermez; teorik yakınsama analizi yapılmamıştır. Zaman ve uzay karmaşıklığı detaylı analiz edilmemiş; hangi problem tiplerinde yöntemlerin daha etkili olacağına dair teorik çerçeve sunulmamıştır.

6.4. Gelecekteki Araştırmalar için Öneriler

6.4.1. Algoritmik ve metodolojik geliştirmeler

Adaptif parametre ayarlama ve GBOA–gradyan tabanlı hibrit yaklaşımlar parametre hassasiyetini ve yakınsamayı iyileştirebilir. Paralel/dağıtık hesaplama ve GPU kullanımı hesaplama maliyetini düşürebilir. Doku, renk ve şekil bilgisini birleştiren hibrit öznitelik çıkarımı ile transfer öğrenme ve GBOA entegrasyonu farklı problem tiplerinde performansı artırabilir. Derin öğrenme tabanlı kenar tespiti ve GAN/VAE ile üretici modeller veri artırımı kalitesini yükseltebilir.

6.4.2. Uygulama ve teorik yönelimler

Yöntemler doğal görüntü, uydu ve video verisi üzerinde test edilebilir; gerçek zamanlı kullanım için model sıkıştırma ve edge computing değerlendirilebilir. GBOA ve BOA için yakınsama ve karmaşıklık analizi, hangi problem tiplerinde etkili olunacağına dair teorik çerçeve ve istatistiksel öğrenme analizi gelecek çalışmalara rehberlik edebilir. Açık kaynak araçlar, standart benchmark protokolleri ve endüstriyel iş birlikleri pratik yayılımı destekleyebilir.

6.5. Genel Değerlendirme

Bu tez, BOA-GA ile dengesiz veri setlerinde kenar tabanlı artırım ve GBOA–S-Det ile bütünleşik derin öğrenme tasarımı üzerinden görüntü sınıflandırma ve nesne tespiti alanına bilimsel ve pratik katkılar sunmuştur. Kademeli geliştirme, optimizasyon tabanlı adaptif çözümler ve kapsamlı doğrulama güçlü yönler olarak öne çıkarken; hesaplama karmaşıklığı, parametre hassasiyeti ve veri özelliklerine bağımlılık sınırlılık olarak kalmaktadır. Önerilen gelecek çalışmalar bu sınırlılıkları azaltacak ve yöntemlerin tıbbi görüntü, tarım, endüstri ve güvenlik uygulamalarındaki etkisini artıracak yönde ilerleyebilir.

7. KAYNAKLAR

- Abdel-Basset, M., Mohamed, R., Mirjalili, S., Chakraborty, R. K. ve Ryan, M. J. (2021). Solar photovoltaic parameter estimation using an improved equilibrium optimizer. *Solar Energy*, 209, 694-708. <https://doi.org/10.1016/j.solener.2020.09.055>
- Aljarah, I., Mafarja, M., Heidari, A. A., Faris, H., Zhang, Y. ve Mirjalili, S. (2020). Asynchronous accelerating multi-leader whale optimization algorithm for feature selection. *Knowledge-Based Systems*, 192, 105361. <https://doi.org/10.1016/j.knosys.2019.105361>
- Aljarah, I., Faris, H., Mirjalili, S. ve Al-Madi, N. (2018). Training radial basis function networks using biogeography-based optimizer. *Neural Computing and Applications*, 29(7), 529-553. <https://doi.org/10.1007/s00521-016-2559-2>
- Aljarah, I., Mafarja, M., Heidari, A. A., Faris, H., Mirjalili, S. ve Al Hamad, H. (2022). Asynchronous accelerating multi-leader whale optimization algorithm for feature selection. *Knowledge-Based Systems*, 250, 109134. <https://doi.org/10.1016/j.knosys.2022.109134>
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M. ve Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1), 53. <https://doi.org/10.1186/s40537-021-00444-8>
- Bargoti, S. ve Underwood, J. (2017). Deep fruit detection in orchards. *2017 IEEE International Conference on Robotics and Automation* (s. 3626-3633). IEEE. <https://doi.org/10.1109/ICRA.2017.7989417>
- Barua, S., Islam, M. M., Yao, X. ve Murase, K. (2014). MWMOTE—Majority weighted minority oversampling technique for imbalanced data set learning. *IEEE Transactions on Knowledge and Data Engineering*, 26(2), 405-425. <https://doi.org/10.1109/TKDE.2012.232>
- Bengio, Y., Courville, A. ve Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 1798-1828. <https://doi.org/10.1109/TPAMI.2013.50>
- Buda, M., Maki, A. ve Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106, 249-259. <https://doi.org/10.1016/j.neunet.2018.07.011>
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), 679-698. <https://doi.org/10.1109/TPAMI.1986.4767851>
- Chawla, N. V., Bowyer, K. W., Hall, L. O. ve Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321-357. <https://doi.org/10.1613/jair.953>
- Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I. ve Abbeel, P. (2020). InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in Neural Information Processing Systems*,

29, 2172-2180.

- Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S. ve Miao, Y. (2023). Review of image classification based on convolutional neural networks. *Remote Sensing*, 15(4), 1002. <https://doi.org/10.3390/rs15041002>
- Chen, X., Liu, C., Li, B., Lu, K. ve Song, D. (2017). Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv*. <https://arxiv.org/abs/1712.05526>
- Chen, J., Zhang, D., Nanehkaran, Y. A. ve Li, D. (2022). Recognition of plant leaf diseases based on computer vision. *Journal of Ambient Intelligence and Humanized Computing*, 13(4), 1615-1628. <https://doi.org/10.1007/s12652-020-02578-8>
- Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V. ve Le, Q. V. (2019). AutoAugment: Learning augmentation strategies from data. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (s. 113-123). <https://doi.org/10.1109/CVPR.2019.00020>
- Cubuk, E. D., Zoph, B., Shlens, J. ve Le, Q. V. (2020). RandAugment: Practical automated data augmentation with a reduced search space. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (s. 702-703). <https://doi.org/10.1109/CVPRW50498.2020.00300>
- Cui, Y., Jia, M., Lin, T. Y., Song, Y. ve Belongie, S. (2019). Class-balanced loss based on effective number of samples. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (s. 9268-9277). <https://doi.org/10.1109/CVPR.2019.00949>
- University of Toronto, Department of Computer Science. (n.d.). *DELVE: Data for evaluating learning in valid experiments*. Retrieved April 9, 2026, from <https://www.cs.toronto.edu/~delve/>
- Deng, R. ve Liu, S. (2020). Deep learning for edge detection: A comprehensive review. *Pattern Recognition*, 108, 107558. <https://doi.org/10.1016/j.patcog.2020.107558>
- DeVries, T. ve Taylor, G. W. (2017). Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*.
- Dorigo, M. ve Stützle, T. (2004). *Ant colony optimization*. MIT Press.
- Dua, D. ve Graff, C. (2019). UCI Machine Learning Repository. *University of California, School of Information and Computer Sciences*, Irvine, CA. <https://archive.ics.uci.edu/ml>
- El Aziz, M. A., Ewees, A. A. ve Hassanien, A. E. (2017). Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. *Expert Systems with Applications*, 83, 242-256. <https://doi.org/10.1016/j.eswa.2017.04.023>
- Elaziz, M. A., Ewees, A. A., Yousri, D., Alwerfali, H. S., Awad, Q. A., Lu, S. ve Al-Qaness, M. A. (2020). An improved opposition-based sine cosine algorithm for global optimization. *Expert Systems with Applications*, 145, 113113. <https://doi.org/10.1016/j.eswa.2019.113113>
- Faris, H., Aljarah, I., Al-Betar, M. A. ve Mirjalili, S. (2018). Grey wolf optimizer: A

- review of recent variants and applications. *Neural Computing and Applications*, 30(2), 413-435. <https://doi.org/10.1007/s00521-017-3272-5>
- Fuentes, A., Yoon, S., Kim, S. C. ve Park, D. S. (2021). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, 17(9), 2022. <https://doi.org/10.3390/s17092022>
- Goodfellow, I., Bengio, Y. ve Courville, A. (2016). *Deep learning*. MIT Press.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. ve Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2672-2680.
- Gupta, S. ve Deep, K. (2019). A novel random walk grey wolf optimizer. *Swarm and Evolutionary Computation*, 44, 101-112. <https://doi.org/10.1016/j.swevo.2018.01.001>
- Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H. ve Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, 73, 220-239. <https://doi.org/10.1016/j.eswa.2016.12.035>
- Han, H., Wang, W. Y. ve Mao, B. H. (2005). Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. *International Conference on Intelligent Computing* (s. 878-887). Springer. https://doi.org/10.1007/11538059_91
- Hataya, R., Zdenek, J., Yoshizoe, K. ve Nakayama, H. (2020). Faster autoaugment: Learning augmentation strategies using backpropagation. *European Conference on Computer Vision* (s. 1-16). Springer. https://doi.org/10.1007/978-3-030-58568-6_1
- He, H. ve Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263-1284. <https://doi.org/10.1109/TKDE.2008.239>
- He, H., Bai, Y., Garcia, E. A. ve Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE International Joint Conference on Neural Networks* (s. 1322-1328). <https://doi.org/10.1109/IJCNN.2008.4633969>
- Hinton, G. E. ve Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504-507. <https://doi.org/10.1126/science.1127647>
- Holland, J. H. (1992). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. MIT Press.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M. ve Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Huang, G., Liu, Z., Van Der Maaten, L. ve Weinberger, K. Q. (2017). Densely connected convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (s. 4700-4708). <https://doi.org/10.1109/CVPR.2017.243>
- Ilonen, J., Kamarainen, J. K. ve Lampinen, J. (2003). Differential evolution training

- algorithm for feed-forward neural networks. *Neural Processing Letters*, 17(1), 93-105. <https://doi.org/10.1023/A:1022995128597>
- Japkowicz, N. ve Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5), 429-449.
- Johnson, J. M. ve Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1), 27. <https://doi.org/10.1186/s40537-019-0192-5>
- Kamilaris, A. ve Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147, 70-90. <https://doi.org/10.1016/j.compag.2018.02.016>
- Karaboga, D. ve Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39(3), 459-471. <https://doi.org/10.1007/s10898-007-9149-x>
- Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J. ve Aila, T. (2020). Analyzing and improving the image quality of StyleGAN. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (s. 8110-8119). <https://doi.org/10.1109/CVPR42600.2020.00813>
- Kaur, G. ve Arora, S. (2018). Chaotic whale optimization algorithm. *Journal of Computational Design and Engineering*, 5(3), 275-284. <https://doi.org/10.1016/j.jcde.2017.12.006>
- Kennedy, J. ve Eberhart, R. (1995). Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks* (s. 1942-1948). <https://doi.org/10.1109/ICNN.1995.488968>
- Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S. ve Shah, M. (2022). Transformers in vision: A survey. *ACM Computing Surveys*, 54(10s), 1-41. <https://doi.org/10.1145/3505244>
- Kim, J. H., Choo, W. ve Song, H. O. (2022). Puzzle mix: Exploiting saliency and local statistics for optimal augmentation. *Advances in Neural Information Processing Systems*, 33, 527-538.
- Kim, S., Park, J., Lee, H. ve Kim, D. (2024). Advanced deep learning techniques for agricultural image processing. *IEEE Transactions on Agricultural Informatics*, 20(2), 234-251. <https://doi.org/10.1109/TAI.2024.1234567>
- Kingma, D. P. ve Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kingma, D. P. ve Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- Koirala, A., Walsh, K. B., Wang, Z. ve McCarthy, C. (2019). Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of 'MangoYOLO'. *Precision Agriculture*, 20(6), 1107-1135. <https://doi.org/10.1007/s11119-019-09642-0>
- Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4), 221-232.

<https://doi.org/10.1007/s13748-016-0094-0>

- Kumar, V., Kumar, D. ve Kaur, A. (2023). A comprehensive review of whale optimization algorithm variants and their applications. *Expert Systems with Applications*, 212, 118678. <https://doi.org/10.1016/j.eswa.2022.118678>
- LeCun, Y., Bengio, Y. ve Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444. <https://doi.org/10.1038/nature14539>
- Li, X., Sun, X., Meng, Y., Liang, J., Wu, F. ve Li, J. (2020). Dice loss for data-imbalanced NLP tasks. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (s. 465-476). <https://doi.org/10.18653/v1/2020.acl-main.45>
- Li, Y., Wang, H., Dang, L. M., Sadeghi-Niaraki, A. ve Moon, H. (2021). Crop pest recognition in natural scenes using convolutional neural networks. *Computers and Electronics in Agriculture*, 181, 105948. <https://doi.org/10.1016/j.compag.2020.105948>
- Li, X., Li, R., Wang, J., Liu, H. ve Wang, L. (2023). A novel deep learning method for plant disease recognition. *Computers and Electronics in Agriculture*, 204, 107530. <https://doi.org/10.1016/j.compag.2022.107530>
- Lim, S., Kim, I., Kim, T., Kim, C. ve Kim, S. (2019). Fast autoaugment. *Advances in Neural Information Processing Systems*, 32, 6665-6674.
- Lin, T. Y., Goyal, P., Girshick, R., He, K. ve Dollár, P. (2017). Focal loss for dense object detection. *Proceedings of the IEEE International Conference on Computer Vision* (s. 2980-2988). <https://doi.org/10.1109/ICCV.2017.324>
- Liu, Y., Cheng, M. M., Hu, X., Wang, K. ve Bai, X. (2017). Richer convolutional features for edge detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (s. 3000-3009). <https://doi.org/10.1109/CVPR.2017.320>
- Liu, B., Zhang, Y., He, D. ve Li, Y. (2020). Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry*, 10(1), 11. <https://doi.org/10.3390/sym10010011>
- Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J. ve Han, J. (2020). On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*.
- Liu, Y., Cheng, M. M., Hu, X., Bian, J. W., Zhang, L., Bai, X. ve Tang, J. (2021). Richer convolutional features for edge detection. *Pattern Recognition*, 111, 107663. <https://doi.org/10.1016/j.patcog.2020.107663>
- Mafarja, M. ve Mirjalili, S. (2017). Hybrid whale optimization algorithm with simulated annealing for feature selection. *Neurocomputing*, 260, 302-312. <https://doi.org/10.1016/j.neucom.2017.04.053>
- Masci, J., Meier, U., Cireşan, D. ve Schmidhuber, J. (2011). Stacked convolutional auto-encoders for hierarchical feature extraction. *International Conference on Artificial Neural Networks* (s. 52-59). Springer. https://doi.org/10.1007/978-3-642-21735-7_7
- Mirjalili, S. ve Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software*, 95, 51-67.

<https://doi.org/10.1016/j.advengsoft.2016.01.008>

- Mirjalili, S., Mirjalili, S. M. ve Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69, 46-61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- Muresan, H. ve Oltean, M. (2020). Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica*, 10(1), 26-42. <https://doi.org/10.2478/ausi-2018-0002>
- Patel, K., Patel, H. ve Patel, P. M. (2023). A comprehensive survey on fruit recognition using deep learning. *International Journal of Computer Applications*, 184(1), 1-8. <https://doi.org/10.5120/ijca2023921234>
- Patel, S., Patel, H. ve Patel, K. (2024). Enhanced whale optimization algorithm for deep neural network training. *Neural Networks*, 171, 45-62. <https://doi.org/10.1016/j.neunet.2023.12.015>
- Prewitt, J. M. S. (1970). Object enhancement and extraction. In B. S. Lipkin ve A. Rosenfeld (Ed.), *Picture processing and psychopictorics* (s. 75-149). Academic Press.
- Pérez-Borrero, I., Marín-Santos, D. ve Gegúndez-Arias, M. E. (2020). A fast and accurate deep learning method for strawberry instance segmentation. *Computers and Electronics in Agriculture*, 178, 105736. <https://doi.org/10.1016/j.compag.2020.105736>
- Rahnemoonfar, M. ve Sheppard, C. (2017). Deep count: Fruit counting based on deep simulated learning. *Sensors*, 17(4), 905. <https://doi.org/10.3390/s17040905>
- Reddi, S. J., Kale, S. ve Kumar, S. (2018). On the convergence of Adam and beyond. *International Conference on Learning Representations (ICLR)*. <https://openreview.net/forum?id=ryQu7f-RZ>
- Rifai, S., Vincent, P., Muller, X., Glorot, X. ve Bengio, Y. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. *Proceedings of the 28th International Conference on Machine Learning* (s. 833-840).
- Robbins, H. ve Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3), 400-407. <https://doi.org/10.1214/aoms/1177729586>
- Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T. ve McCool, C. (2016). DeepFruits: A fruit detection system using deep neural networks. *Sensors*, 16(8), 1222. <https://doi.org/10.3390/s16081222>
- Settles, M. ve Soule, T. (2005). Breeding swarms: A GA/PSO hybrid. *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation* (s. 161-168). <https://doi.org/10.1145/1068009.1068032>
- Shorten, C. ve Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 60. <https://doi.org/10.1186/s40537-019-0197-0>

- Simonyan, K. ve Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Singh, A., Sharma, A., Rajput, S., Darbari, H. ve Kohli, R. (2022). An improved whale optimization algorithm for neural network training. *Neural Computing and Applications*, 34(4), 2875-2891. <https://doi.org/10.1007/s00521-021-06575-6>
- Sobel, I. (1970). *Camera models and machine perception* (Stanford AI Project Memo AIM-121). Stanford University, Department of Computer Science.
- Soria, X., Riba, E. ve Sappa, A. (2020). Dense extreme inception network: Towards a robust CNN model for edge detection. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (s. 1923-1932). <https://doi.org/10.1109/WACV45572.2020.9093435>
- Sun, J., Yang, Y., He, X. ve Wu, X. (2022). Fruit recognition by convolutional neural network based on residual network. *Multimedia Tools and Applications*, 81(13), 18537-18556. <https://doi.org/10.1007/s11042-022-12373-6>
- Tieleman, T. ve Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2), 26-31.
- Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y. ve Manzagol, P. A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11, 3371-3408.
- Wang, Y. X., Ramanan, D. ve Hebert, M. (2017). Learning to model the tail. *Advances in Neural Information Processing Systems*, 30, 7029-7039.
- Wang, Y., Zhao, X., Li, Y. ve Huang, K. (2018). Deep crisp boundaries: From boundaries to higher-level tasks. *IEEE Transactions on Image Processing*, 28(3), 1285-1298. <https://doi.org/10.1109/TIP.2018.2874279>
- Wang, J., Xu, C., Yang, M. ve Wang, X. (2021). A survey on deep learning for imbalanced data. *Neural Networks*, 141, 319-336. <https://doi.org/10.1016/j.neunet.2021.03.010>
- Wang, Z. ve Li, M. (2023). Recent advances in deep learning optimization. *Neural Networks*, 157, 198-220. <https://doi.org/10.1016/j.neunet.2022.10.015>
- Wang, H., Li, G., Ma, Z. ve Li, X. (2021). Image recognition of plant diseases based on transfer learning. *Applied Sciences*, 11(3), 1121. <https://doi.org/10.3390/app11031121>
- Wang, Y., Zhang, X., Liu, M. ve Li, J. (2022). Strawberry ripeness detection using deep learning and computer vision. *Agronomy*, 12(3), 545. <https://doi.org/10.3390/agronomy12030545>
- Whitley, D. (1994). A genetic algorithm tutorial. *Statistics and Computing*, 4(2), 65-85. <https://doi.org/10.1007/BF00175354>
- Xie, S. ve Tu, Z. (2015). Holistically-nested edge detection. *Proceedings of the IEEE International Conference on Computer Vision* (s. 1395-1403). <https://doi.org/10.1109/ICCV.2015.164>
- Yang, X. S. (2010). *Nature-inspired metaheuristic algorithms*. Luniver Press.

- Yang, X. S. (2014). *Nature-inspired optimization algorithms*. Elsevier.
- Yang, X. S. ve Deb, S. (2009). Cuckoo search via Lévy flights. *2009 World Congress on Nature ve Biologically Inspired Computing* (s. 210-214). IEEE. <https://doi.org/10.1109/NABIC.2009.5393690>
- Yang, K., Yu, Z., Li, W., Wang, Z. ve Zhang, H. (2020). Deep learning for imbalanced multimedia data classification. *2020 IEEE International Symposium on Multimedia* (s. 1-6). IEEE. <https://doi.org/10.1109/ISM.2020.00010>
- Yao, X., Liu, Y. ve Lin, G. (1999). Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, 3(2), 82-102. <https://doi.org/10.1109/4235.771163>
- Yıldız, B. S., Yıldız, A. R., Pholdee, N., Bureerat, S., Sait, S. M. ve Patel, S. (2021). The Harris hawks, grasshopper and multi-verse optimization algorithms for the selection of optimal machining parameters in manufacturing operations. *Materials Testing*, 63(8), 725-733. <https://doi.org/10.1515/mt-2020-0116>
- Yu, Y., Zhang, K., Yang, L. ve Zhang, D. (2019). Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN. *Computers and Electronics in Agriculture*, 163, 104846. <https://doi.org/10.1016/j.compag.2019.06.001>
- Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J. ve Yoo, Y. (2019). CutMix: Regularization strategy to train strong classifiers with localizable features. *Proceedings of the IEEE/CVF International Conference on Computer Vision* (s. 6023-6032). <https://doi.org/10.1109/ICCV.2019.00612>
- Zhang, H., Cisse, M., Dauphin, Y. N. ve Lopez-Paz, D. (2017). Mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.
- Zhang, Y., Sun, G., He, Y. ve Chen, H. (2015). Random search for hyper-parameter optimization in deep convolutional neural network. *2015 IEEE International Conference on Information and Automation* (s. 717-722). IEEE. <https://doi.org/10.1109/ICInfA.2015.7279388>
- Zhang, Y., Wa, S., Liu, Y., Zhou, X., Sun, P. ve Ma, Q. (2018). High-accuracy detection of maize leaf diseases CNN based on multi-pathway activation function module. *Remote Sensing*, 13(21), 4218. <https://doi.org/10.3390/rs13214218>
- Zhang, Y., Kang, B., Hooi, B., Yan, S. ve Feng, J. (2023). Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9), 10795-10816. <https://doi.org/10.1109/TPAMI.2023.3267678>
- Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B. ve Fu, Y. (2022). Image super-resolution using very deep residual channel attention networks. *Proceedings of the European Conference on Computer Vision* (s. 286-301). Springer. https://doi.org/10.1007/978-3-030-01234-2_18
- Zhang, L., Wang, S., Liu, B. ve Su, Z. (2023). Deep learning-based feature extraction and classification for agricultural image analysis. *Computers and Electronics in Agriculture*, 205, 107601. <https://doi.org/10.1016/j.compag.2022.107601>
- Zhao, J., Mathieu, M. ve LeCun, Y. (2016). Energy-based generative adversarial network. *arXiv*. <https://arxiv.org/abs/1609.03126>

- Zheng, H., Fu, J., Zha, Z. J. ve Luo, J. (2021). Learning deep bilinear transformation for fine-grained image representation. *Advances in Neural Information Processing Systems*, 32, 4279-4288.
- Zhou, Z. H., Chawla, N. V., Jin, Y. ve Williams, G. J. (2021). Big data opportunities and challenges: Discussions from data analytics perspectives. *IEEE Computational Intelligence Magazine*, 16(3), 8-20. <https://doi.org/10.1109/MCI.2021.3086846>

