

T.C.  
FIRAT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ



DERİN ÖĞRENME TEKNİKLERİ İLE FUNDUS  
GÖRÜNTÜLERİNDEN GLOKOM HASTALIĞI TESPİTİ

Özcan YILDIRIM

Yüksek Lisans Tezi

YAZILIM MÜHENDİSLİĞİ ANABİLİM DALI

Yazılım Mühendisliği Bilim Dalı

OCAK 2024

T.C.  
FIRAT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

Yazılım Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

# DERİN ÖĞRENME TEKNİKLERİ İLE FUNDUS GÖRÜNTÜLERİNDEN GLOKOM HASTALIĞI TESPİTİ

Tez Yazarı  
Özcan YILDIRIM

Danışman  
Dr. Öğr. Üyesi Feyza ALTUNBEY ÖZBAY

OCAK 2024  
ELAZIĞ

**T.C.**  
**FIRAT ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

Yazılım Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

---

Başlığı: Derin Öğrenme Teknikleri ile Fundus Görüntülerinden Glokom Hastalığı Tespiti

Yazarı: Özcan YILDIRIM

İlk Teslim Tarihi: 08.12.2023

Savunma Tarihi: 18.01.2024

---

**TEZ ONAYI**

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına göre hazırlanan bu tez aşağıda imzaları bulunan jüri üyeleri tarafından değerlendirilmiş ve akademik dinleyicilere açık yapılan savunma sonucunda OYBİRLİĞİ ile kabul edilmiştir.

Danışman:	Dr. Öğr. Üyesi Feyza ALTUNBEY ÖZBAY Fırat Üniversitesi, Mühendislik Fakültesi	<i>İmza</i> Onayladım
Başkan:	Prof. Dr. Bilal ALATAŞ Fırat Üniversitesi, Mühendislik Fakültesi	Onayladım
Üye:	Doç. Dr. Muhammed YILDIRIM Malatya Turgut Özal Üniversitesi, Mühendislik Fakültesi	Onayladım

Bu tez, Enstitü Yönetim Kurulunun ...../...../20..... tarihli toplantısında tescillenmiştir.

*İmza*

Prof. Dr. Burhan ERGEN  
Enstitü Müdürü

## BEYAN

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım ‘‘Derin Öğrenme Teknikleri ile Fundus Görüntülerinden Glokom Hastalığı Tespiti’’ Başlıklı Yüksek Lisans Tezimin içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

18.01.2024

**Özcan YILDIRIM**



# ÖNSÖZ

Glokom, dünya çapında geri dönüşü olmayan körlüğün önde gelen nedenidir ve subjektif değerlendirmeye dayanması nedeniyle önemli teşhis zorlukları doğurmaktadır. Bununla birlikte, bilgisayarlı görmedeki son gelişmeler, otomatik değerlendirmenin potansiyelini ortaya koymaktadır. Bu çalışmada fundus görüntülerinden glokom tespiti için metasezgisel algoritmaları temel alan ve farklı aşamalardan oluşan yeni bir yöntem önerilmiştir. Önerilen yaklaşım ile fundus görüntülerinden derin öğrenme mimarileri ile özellik çıkarımı yapılmıştır. Ardından ilgili ve kullanışlı özelliklerin belirlenmesi için metasezgisel optimizasyon algoritmaları ile özellik çıkarımı yapılmıştır. Elde edilen veri kümesine farklı sınıflandırma algoritmaları kullanılarak glokom hastalığının tespiti üzerindeki performansları farklı performans değerlendirme ölçütlerine göre karşılaştırılmıştır.

Çalışmamı yürütürken bana destek olan danışmanım Dr. Öğr. Üyesi Feyza ALTUNBEY ÖZBAY'a görüş ve önerileri ile desteğini esirgemediği için teşekkürlerimi sunarım. Bu tez çalışması, Fırat Üniversitesi Bilimsel Araştırma Projeleri Koordinasyonu Birimi (FÜBAP) tarafından MF.23.40 protokol numaralı proje ile desteklenmiştir.

**Özcan YILDIRIM**  
ELAZIĞ, 2024

# İÇİNDEKİLER

	Sayfa
ÖNSÖZ.....	iv
İÇİNDEKİLER .....	v
ÖZET .....	vii
ABSTRACT .....	viii
ŞEKİLLER LİSTESİ .....	ix
TABLolar LİSTESİ .....	x
KISALTMALAR .....	xi
<b>1. GİRİŞ .....</b>	<b>1</b>
1.1. Motivasyon.....	2
1.2. Amaç ve Hedefler.....	2
1.3. Tezin Yapısı .....	3
<b>2. GLOKOM HASTALIĞI.....</b>	<b>4</b>
2.1. Gözün Yapısı.....	4
2.2. Glom Hastalığı.....	4
2.3. Glom Anatomisi ve Fizyolojisi.....	6
2.4. Glom Sınıflandırılması .....	7
2.4.1. Açık Açılı Glom .....	7
2.4.2. Kapalı Açılı Glom .....	8
2.4.3. Konjenital Glom.....	8
2.5. Literatür Taraması .....	8
<b>3. YAPAY ZEKÂ VE MAKİNE ÖĞRENMESİ.....</b>	<b>11</b>
3.1. Yapay Zekâ.....	11
3.2. Makine Öğrenmesi .....	12
3.2.1. Denetimli Öğrenme .....	12
3.2.2. Denetimsiz Öğrenme.....	13
3.2.3. Yarı Denetimli Öğrenme .....	13
3.2.4. Takviyeli Öğrenme.....	13
3.3. Makine Öğrenmesi Algoritmaları.....	13
3.3.1. Destek Vektör Makineleri .....	13
3.3.2. K-En Yakın Komşu Algoritması .....	14
3.3.3. Naive Bayes Algoritması.....	15
3.3.4. Topluluk Öğrenme Algoritması .....	16
3.3.5. Karar Ağacı Algoritması .....	16
3.3.6. Lojistik Regresyon Algoritması .....	17
3.4. Yapay Sinir Ağları.....	17
3.5. Evrimsel Sinir Ağları .....	18
3.5.1. Evrişim Katmanı .....	19
3.5.2. Havuzlama Katmanı .....	19
3.5.3. Tam Bağlantı Katmanı .....	20
3.5.4. Yoğun Katmanı .....	20
3.6. Derin Öğrenme Mimarileri.....	20
3.6.1. DenseNet201 Mimarisi .....	20

3.6.2. Inceptionv3 Mimarisi .....	21
3.7. Metasezgisel Optimizasyon Algoritmaları .....	22
3.7.1. Balina Optimizasyon Algoritması .....	27
3.7.2. Parçacık Sürü Optimizasyon Algoritması .....	33
<b>4. MATERYAL VE METOD .....</b>	<b>36</b>
4.1. Veri Kümesinin Hazırlanması .....	37
4.2. Değerlendirme Kriterleri .....	37
<b>5. BULGULAR VE TARTIŞMA.....</b>	<b>39</b>
5.1. Balina Optimizasyon Algoritması Sonuçları .....	39
5.2. Parçacık Sürü Optimizasyon Algoritması Sonuçları .....	41
5.3. Orijinal Veri Kümesi İçin Farklı Sınıflandırıcıların Performansları.....	43
5.4. Özellik Seçimi Sonrası Alt Veri Kümeleri İçin Farklı Sınıflandırıcıların Performansları.....	44
<b>6. SONUÇLAR.....</b>	<b>49</b>
ÖNERİLER .....	50
KAYNAKLAR .....	51
ÖZGEÇMİŞ .....	

# ÖZET

## Derin Öğrenme Teknikleri ile Fundus Görüntülerinden Glokom Hastalığı Tespiti

Özcan YILDIRIM

Yüksek Lisans Tezi

FIRAT ÜNİVERSİTESİ  
Fen Bilimleri Enstitüsü

Yazılım Mühendisliği Anabilim Dalı

Ocak 2024, Sayfa: xi + 56

Günümüzde insan nüfusunun hızlı bir şekilde artması sağlık alanında iş yükünün artmasına neden olmuştur. Doktor ve hastalar açısından hastalıkların erken tanı, teşhisi ve tedavi yöntemleri son derece önemli hale gelmiştir. İnsan gözündeki fizyolojik değişikliklerin tanımlanması, sağlık sektöründe zorlu bir görevdir. Bu nedenle retinal görüntü analizi, günümüzde hastalık tespit sistemlerinde hastalığın taranması ve teşhisinde araştırmacılar arasında daha fazla ilgi görmüştür. Özellikle tespit edilmesi zor hastalıkların erken teşhisi insan hayatının devamlılığı açısından önem arz etmektedir. Glokom hastalığı temel olarak gözdeki optik sinirin bozulmasından kaynaklanan bir göz hastalığıdır. Başlangıç evresinde kesin tanı açısından belirti göstermeyen bu hastalığın erken tespiti oldukça zordur. Bu çalışmada, genel erişime açık retinal fundus görüntülerinden glokom hastalığı tespiti metasezgisel optimizasyon algoritması ile birleştirilmiş derin öğrenme yaklaşımının kullanıldığı bir yöntem önerilmiştir. Önerilen yöntemde, öncelikle retinal fundus görüntülerinden oluşan veri kümesinden Evrişimli Sinir Ağı mimarilerinden DenseNet201 ve Inceptionv3 mimarisi ile özellik çıkarımı yapılmıştır. Veri kümesinden ilgisiz özellikleri elemek ve ideal verilere ulaşmak için metasezgisel optimizasyon algoritmalarından Balina Optimizasyon Algoritması (BOA) ve Parçacık Sürü Optimizasyonu (PSO) kullanılmıştır. BOA ve PSO farklı popülasyon büyüklükleri ve iterasyon sayılarına göre optimal özellik seçiminde kullanılmıştır. Orijinal veri kümesi ve ile özellik seçimi yapıldıktan sonra elde edilen yeni veri kümelerine Naive Bayes, Destek Vektör Makinesi, K-En Yakın Komşu, Karar Ağacı, Logistic Regression ve Topluluk Öğrenme sınıflandırıcıları uygulanmıştır. Sınıflandırma algoritmalarının performanslarının değerlendirilmesi aşamasında ise doğruluk, duyarlılık, özgüllük ve f1-ölçütü metrikleri kullanılmıştır. Elde edilen sonuçlara göre BOA ve PSO ile belirlenen alt veri kümelerinden sınıflandırıcının performanslarında umut verici değerler elde edildiği gözlemlenmiştir.

**Anahtar Kelimeler:** Glokom Hastalığı, Evrişimsel Sinir Ağları, Derin Öğrenme, Optimizasyon

# ABSTRACT

---

## Detection of Glaucoma from Fundus Images with Deep Learning Techniques

Özcan YILDIRIM

Master's Thesis

FIRAT UNIVERSITY

Graduate School of Natural and Applied Sciences

Department of Software Engineering

Electrical and Electronics Engineering Technologies

January 2024, Pages: xi + 56

---

Today, the rapid increase in the human population has caused an increase in the workload in the field of healthcare. Early diagnosis, diagnosis and treatment methods of diseases have become extremely important for doctors and patients. Identifying physiological changes in the human eye is a challenging task in the healthcare industry. For this reason, retinal image analysis has attracted more attention among researchers in disease screening and diagnosis in disease detection systems today. Early diagnosis of diseases that are especially difficult to detect is important for the continuity of human life. Glaucoma is an eye disease that is basically caused by the deterioration of the optic nerve in the eye. Early detection of this disease, which does not show any symptoms in terms of definitive diagnosis in its initial stage, is very difficult. In this study, a method for detecting glaucoma disease from publicly accessible retinal fundus images using a deep learning approach combined with a metaheuristic optimization algorithm is proposed. In the proposed method, feature extraction was first made from the dataset consisting of retinal fundus images with DenseNet201 and Inceptionv3 architectures, which are Convolutional Neural Network architectures. Whale Optimization Algorithm (BOA) and Particle Swarm Optimization (PSO), metaheuristic optimization algorithms, were used to eliminate irrelevant features from the dataset and reach ideal data. BOA and PSO were used in optimal feature selection according to different population sizes and iteration numbers. After the original data set and feature selection were made, Naive Bayes, Support Vector Machine, K-Nearest Neighbor, Decision Tree, Logistic Regression, and Ensemble Learning classifiers were applied to the new data sets obtained. Accuracy, sensitivity, specificity, and f1-criterion metrics were used to evaluate the performance of the classification algorithms. According to the results obtained, it was observed that promising values were obtained in the performance of the classifier from the sub-datasets determined by BOA and PSO.

**Keywords:** Glaucoma Disease, Convolutional Neural Networks, Deep Learning, Optimization

# ŞEKİLLER LİSTESİ

Sayfa

Şekil 1. 1. Yaşa bağlı glokomlu tahmini insan sayısı .....	1
Şekil 2. 1. İnsan gözünün anatomik yapısı [7].....	4
Şekil 2. 2. Glokomun kademeli olarak ilerlemesi nedeniyle yan görüşün azalması .....	5
Şekil 2. 3. Fundus görüntü örneği .....	6
Şekil 2. 4. Optik disk ve optik kupa örneği [18].....	7
Şekil 3. 1. Yapay zekâ ve alt dallarının şematik gösterimi.....	12
Şekil 3. 2. Destek vektör makinesi şematik gösterimi.....	14
Şekil 3. 3. KNN algoritması sınıflandırma örneği [45] .....	15
Şekil 3. 4. Karar ağacı algoritması şematik gösterimi [53] .....	17
Şekil 3. 5. Yapay sinir hücresi yapısı [56].....	18
Şekil 3. 6. Evrişimsel sinir ağlarının mimarisi .....	18
Şekil 3. 7. Evrişim katmanında filtre uygulanması [59].....	19
Şekil 3. 8. Havuzlama işlemi [60] .....	20
Şekil 3. 9. DenseNet201 mimarisi [67] .....	21
Şekil 3. 10. Standart bir başlangıç modülü yapısı [72].....	22
Şekil 3. 11. Özellik seçim süreci .....	23
Şekil 3. 12. Özellik seçimi yöntemlerinin çeşitleri.....	23
Şekil 3. 13. Metasezgisel Algoritmalar .....	27
Şekil 3. 14. Balinaların kabarcık ağıyla beslenme yöntemi [92].....	28
Şekil 3. 15. Arama ajanının iki ve üç boyutlu vektörleri ve sonraki olası konumları [90] .....	29
Şekil 3. 16. Küçülen çevreleme mekanizması [91] .....	30
Şekil 3. 17. Spiral konum güncellemesi [91].....	31
Şekil 3. 18. Arama mekanizması [90] .....	32
Şekil 3. 19. BOA'nun akış diyagramı.....	33
Şekil 3. 20. PSO'nun akış diyagramı.....	35
Şekil 4. 1. Önerilen yöntemin şematik gösterimi .....	36
Şekil 4. 2. Glokom görüntüsü (a), normal görüntüsü (b).....	37

## TABLolar LİSTESİ

	Sayfa
<b>Tablo 4. 1.</b> Veri kümesinden alınan görüntülerin dağılım sayısı .....	37
<b>Tablo 4. 2.</b> Karmaşıklık matrisi .....	37
<b>Tablo 5. 1.</b> InceptionV3 sonrası BOA'nın farklı popülasyonlarda özellik seçim sonuçları .....	40
<b>Tablo 5. 2.</b> DenseNet201 sonrası BOA'nın farklı popülasyonlarda özellik seçim sonuçları .....	41
<b>Tablo 5. 3.</b> InceptionV3 sonrası PSO'nun farklı popülasyonlarda özellik seçim sonuçları .....	42
<b>Tablo 5. 4.</b> DenseNet201 sonrası PSO'nun farklı popülasyonlarda özellik seçim sonuçları .....	43
<b>Tablo 5. 5.</b> Inceptionv3 mimarisi ile çıkan özelliklerin ile farklı sınıflandırıcı performansları .....	43
<b>Tablo 5. 6.</b> Densenet201 mimarisi ile çıkan özelliklerin farklı sınıflandırıcı performansları .....	44
<b>Tablo 5. 7.</b> InceptionV3+BOA için farklı sınıflandırıcı performansları .....	45
<b>Tablo 5. 8.</b> DenseNet201+BOA için farklı sınıflandırıcı performansları.....	46
<b>Tablo 5. 9.</b> InceptionV3+PSO için farklı sınıflandırıcı performansları.....	47
<b>Tablo 5. 10.</b> DenseNet201+ PSO için farklı sınıflandırıcı performansları.....	48

---

## KISALTMALAR

### Kısaltmalar

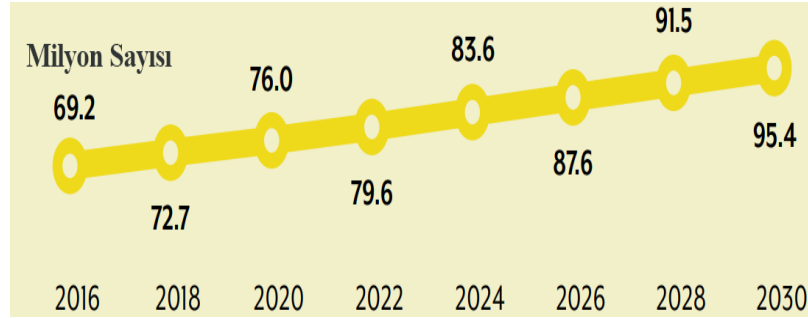
---

YSA	: Yapay sinir ađları
GİB	: Gz ii basıncı
KNN	: K-Nearest Neighbor (k-en yakın komşuluk)
SVM	: Support Vector Machine (Destek vektr makinesi)
ESA	: Evrişimsel Sinir Ađları
PSO	: Paracık Srs Optimizasyonu
BOA	: Balina Optimizasyon Algoritması
OD	: Optik disk
OC	: Optik kupa
DP	: Dođru Pozitif
YP	: Yanlıř Pozitif
YN	: Yanlıř Negatif
YP	: Yanlıř Pozitif

# 1. Giriş

Glokom, diyabetik retinopati, katarakt ve yaşa bağlı maküler dejenerasyon iyi bilinen oftalmik bozukluklardır ve dünya çapında varlıkları kanıtlanmıştır. Dünya çapında geri döndürülemez körlüğün kataraktan sonra ikinci en yaygın nedeni glokomdur [1]. Glokom, optik sinirlere zarar veren ve doğru tanı konmadığı için geri dönüşü olmayan körlüğe neden olan karmaşık bir hastalıktır. Bu nedenle erken teşhis, görme ve yaşam kalitesini korumak için oldukça önemlidir. Glokom, gözün görme sinirine zarar veren bir hastalıktır ve başlangıçta hiçbir uyarı işareti bulunmamaktadır, bu nedenle sıklıkla “sessiz görüş hırsızı” olarak adlandırılmaktadır. Glokom yeni doğan bebekler dâhil her yaştaki bireyi etkileyebilmektedir. Tahminlere göre, dünya genelinde 2013 yılında 40-80 yaş arası yaklaşık 64.3 milyon kişiye glokom teşhisi konduğu ve glokomun 2040 yılına kadar yaklaşık 112 milyon kişiyi etkilemesi beklenmektedir [2].

2019 yılında Dünya Sağlık Örgütü tarafından yapılan araştırmaya göre, 2030 yılına gelindiğinde, dünya genelinde 60 yaş ve üzeri insan sayısının 962 milyondan 1,4 milyara çıkacağı, 80 yaş üstü insan sayısının ise 137 milyondan 202 milyona çıkacağı tahmin ediliyor. Bu nüfus değişiklikleri, görme bozukluğuna neden olan önemli göz rahatsızlıklarına sahip kişilerin sayısında önemli artışlara yol açacaktır. Yaşa bağlı göz rahatsızlığı glokomu olan kişilerin tahmini sayısı Şekil 1.1’de gösterilmiştir. 2022 yılında 76.9 milyon 2030 yılında 95,4 milyon civarında olması tahmin edilmektedir. Yaklaşık 1.3 kat artması öngörülmektedir [3].



Şekil 1. 1. Yaşa bağlı glokomlu tahmini insan sayısı

Derin öğrenme ve yapay zekâ teknikleri görüntü işleme alanında önemli bir etkiye sahiptir. Bu yöntemler, nesnelere otomatik tanınması, görüntü sınıflandırması ve segmentasyonu alanlarında başarıyla kullanılmaktadır. Ayrıca gün geçtikçe artan hasta kayıtları, karmaşıklık düzeyi yükselen tıbbi verilerin doktorlar tarafından makul sürelerde analiz edilmesi daha zor hale gelmektedir. Deneyimli uzman hekimlere ulaşmada zorlukların yaşanması önemli bir sorun oluşturmaktadır. Yapay zekâ teknikleri sağlık alanındaki büyük veri yığınlarını başarı ile analiz

etmekte ve insanlar tarafından tespit edilmesi zor hastalıkları yüksek başarımlı oranları ile hızlı bir şekilde tespit edilme imkânı sunmaktadır [4,5].

## 1.1. Motivasyon

Glokom hastalığı, Dünya Sağlık Örgütü tarafından körlüğün önde gelen nedenlerinden biri olarak kabul edilmektedir. Glokom hiçbir belirti göstermeden, sinsi bir şekilde ilerleyen bir hastalıktır. Genellikle optik sinir geri dönülemez biçimde hasar görene kadar fark edilememektedir. Günümüzde göz hastalıklarını tespit etmek için otomatik bilgisayar destekli retinal fundus görüntüleri popüler olarak kullanılmaktadır. Bu görüntüler, görüntü işleme, analiz ve bilgisayarla görme teknikleri ile tıp biliminin tüm alanlarında giderek daha fazla ön plana çıkmaktadır.

Derin öğrenme modelleri, görüntü verilerinden karmaşık desenleri ve temsilleri otomatik olarak öğrenme ve yakalama konusunda doğal bir yeteneğe sahiptir; bu da onları tıbbi görüntüleri analiz etmek ve anormallikleri tespit etmek için çok uygun hale getirmektedir. Ayrıca metasezgisel tabanlı algoritmalarla büyük veri kümelerinden en ideal veri kümelerine ulaşıp, çok daha hızlı karar destek sistemlerinin oluşturulmasını sağlamaktadır. Bu çalışmada, derin öğrenme modellerini ve metasezgisel optimizasyon algoritmaları kullanarak glokom tespiti için güvenilir ve doğru bir yöntem geliştirilecektir. Önerilen yaklaşımın farklı veri kümelerine uygulanabilir olması ya da farklı algoritmalar kullanılmasına olanak tanınması farklı tıbbi alanlarda etkili ve doğru teşhis için bir çerçeve sağlayacaktır.

## 1.2. Amaç ve Hedefler

Günümüzde insan nüfusunun artması özellikle sağlık alanında iş yükünün artmasına neden olmuştur. Bu nedenle, doktor ve hastalar açısından hastalıkların erken tanı ve tedavi yöntemleri son derece önemli hale gelmiştir. Özellikle kesin tanı açısından belirti göstermeyen hastalıkların erken teşhisi insan sağlığı açısından önem arz etmektedir.

Bu çalışma ile glokomun ciddiyeti ve görülme sıklığı nedeniyle erken evrelerde daha yüksek doğrulukla teşhis edebilen, derin öğrenme modellerine dayalı otomatik sistemler ile hastalığın tespiti amaçlanmıştır. Bu kapsamda yapılacak çalışmada farklı adımlardan oluşmaktadır. İlk olarak farklı derin öğrenme mimarileri kullanarak fundus görüntülerinden özellikler çıkarılmıştır. Derin öğrenme mimarileri ile çıkarılan özelliklerden en ideal olanları seçmek için farklı metasezgisel algoritmalar kullanılmıştır. Orijinal veri kümesi ve metasezgisel algoritmaların uygulanması sonucu elde edilen alt veri kümelerine farklı sınıflandırma algoritmaları uygulanmıştır. Bu algoritmaların performansları karmaşıklık matrisinden elde edilecek doğruluk, duyarlılık, özgüllük ve f1-ölçütü bakımından karşılaştırılmıştır. Bu sayede glokom hastalığının tespiti için derin öğrenme ve metasezgisel algoritmaları temel alan etkili bir yöntem önerilmesi amaçlanmaktadır.

### 1.3. Tezin Yapısı

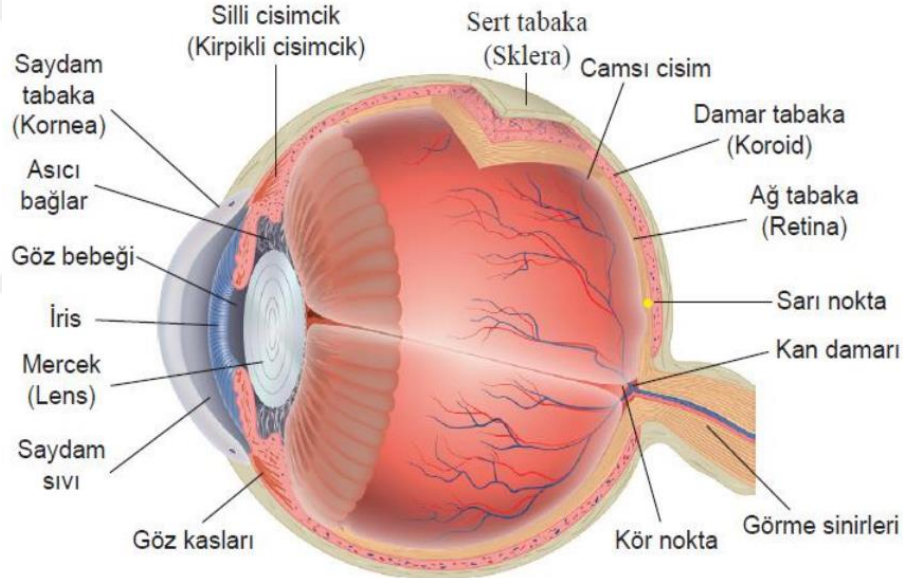
Tez çalışması altı bölümden oluşmaktadır. Birinci bölümde dünya genelinde glokom hastalığı hakkında bilgiler verilerek ve dünya çapında görülme sıklıkları incelenmiştir. İkinci bölümde glokom hastalığını hedef alan gözün yapısı ve hastalık ile ilgili detaylı bilgiler sunulmuş ve literatürde glokom hastalığının tespiti için önerilen farklı çalışmalar analiz edilmiştir. Üçüncü bölümde ise yapay zekâ ve makine öğrenmesi hakkında bilgiler verilmiş, çalışmada glokom hastalığının tespiti için kullanılan makine öğrenmesi algoritmaları incelenmiştir. Ayrıca bu bölümde evrışimli sinir ağıları ve fundus görüntülerinden özellik çıkarımı için kullanılan derin öğrenme mimarileri hakkında bilgiler verilmiştir. Özellik çıkarımı için kullanılan metasezgisel optimizasyon algoritmalar hakkında bilgiler de bu bölüm kapsamında sunulmuştur. Dördüncü bölümde kullanılan fundus görüntüleri veri kümesi hakkında bilgiler verilmiştir. Ayrıca makine öğrenmesi algoritmalarının fundus görüntülerinden glokom tespiti için performansını analiz etmek için kullanılan performans değerlendirme ölçütleri bu bölümde verilmiştir. Beşinci bölümde ise, deney sonuçları karşılaştırmalı olarak incelenmiştir.

Son bölümde ise tez çalışması kapsamında önerilen yöntem değerlendirilmiş ve ileriye dönük çalışmalar için önerilerde bulunulmuştur.

## 2. GLOKOM HASTALIĞI

### 2.1. Gözün Yapısı

Gözün dış tabakası ön kısım kornea ve arka kısım sert tabaka olmak üzere iki kısımdan oluşmaktadır. Kornea şeffaftır ve yaklaşık 8 mm'lik eğrilik yarıçapına sahip küresel kısımdır. Kornea çapı yatay olarak yaklaşık 11.5 mm dikey 10.5 mm kadardır. Sert tabaka yaklaşık 24 mm çapından olup, yoğun ve beyaz koruyucu fonksiyona sahiptir. Gözün iç tabakası, merkezi sinir sisteminin bir uzantısı olan ve optik sinir yoluyla beyne bağlanan retinadan oluşmaktadır. Arka segmentin iç kısmına ise fundus denir. Fundus genellikle retinoskopi gibi iç muayenede görünen gözün arka kısmını ifade etmektedir [6]. İnsan gözünün anatomik yapısı Şekil 2.1'de gösterilmiştir [7].



Şekil 2. 1. İnsan gözünün anatomik yapısı [7]

### 2.2. Glokom Hastalığı

Glokom, gözde optik bir nöropatiyle kendini gösteren sinsi bir hastalıktır. Göz içi aköz sıvının oluşturduğu basınç, kişinin görme sinirine sinsi zarar vererek kalıcı görme kaybına neden olmaktadır. Görme kaybı, yan görme kaybıyla başlar ve Şekil 2.2'de gösterildiği gibi gözün merkezine doğru yaklaşmaktadır. Yaş, cinsiyet, etnik köken, merkezi kornea kalınlığı ve miyopi glokomda risk faktörleri olarak sıralanabilir. Glokom açısından belli kişiler daha fazla riske sahiptir. Ailesinde glokom öyküsü olan kişiler ciddi risk altındadır ve bu kişilerin rutin olarak göz tansiyonlarını ölçtürmesi gerekmektedir. [8,9].



Görme kaybı yok (Normal)



Görme kaybı (Glokom)

Şekil 2. 2. Glokomun kademeli olarak ilerlemesi nedeniyle yan görüşün azalması

Ayrıca glokom, optik sinir başı ve retinal sinir lifi tabakasında belirgin değişikliklerle sonuçlanan, retina gangliyon hücrelerinin kaybı ile karakterize kronik nörodejeneratif bir hastalıktır. Uygulamada, glokomu klinik olarak saptamak için optik sinir başı değerlendirmesi, göz içi basıncı (GİB) ölçümü ve fonksiyona dayalı görme alanı testi gibi üç test yapılmaktadır [10,11]. Bunlar arasında, optik sinir başı değerlendirmesinin en uygun olduğu ve glokom uzmanları tarafından yaygın olarak kullanıldığı bilinmektedir [12].

**Göz içi basıncı ölçümü:** Tonometri göz içindeki basıncı ölçmektedir. Tonometri sırasında gözü uyuşturmak için göz damlaları kullanılmaktadır. Daha sonra bir doktor veya teknisyen, gözün iç basıncını ölçmek için tonometre adı verilen bir cihaz kullanmaktadır. Göz içi basınç değeri 21 mm Hg'yi aşarsa kişiye glokom tanısı konulmaktadır [13].

**Retina sinir lifi değerlendirmesi:** Oftalmoskopide doktor optik siniri glokom hasarı açısından incelemektedir. Göz damlaları gözbebeğini genişletmek için kullanılır, böylece doktor gözün içini görerek optik sinirin şeklini ve rengini inceleyebilmektedir. Göz içi basıncı normal aralıkta değilse veya optik sinir olağandışı görünüyorsa hastanın görme alanını haritalandıran görme testi yapılmaktadır [13].

**Görme alanı testi:** Perimetri kornea kalınlığını ölçen basit bir testtir. Bu test ile bireyin görme alanı haritalandırılmaktadır. Hastalık tespitinde uzmanlara klinik tanıya yardımcı olmaktadır [13].

**Fundus Fotoğrafçılığı:** Gözün arka kısmının fotoğraflanması olarak tanımlanmaktadır. Fundus kamerası veya retinal kamera, gözün iç yüzeyini fotoğraflamak için tasarlanmış flaş özellikli bir kamerayla birlikte kullanılan özel bir mikroskoptan oluşmaktadır. Görüntüleme düzlemine yerleştirilen yarı şeffaf retina dokularının fotoğrafının elde edilmesi için ışığın yansıtılması yöntemi olarak tanımlanmaktadır. Bu nedenle, gözbebeği içinden gözün iç kısmının bir dizi fotoğrafının çekilmesiyle fundus fotoğrafları elde edilmektedir. Şekil 2.3'te fundus fotoğrafı örneği gösterilmiştir [14].



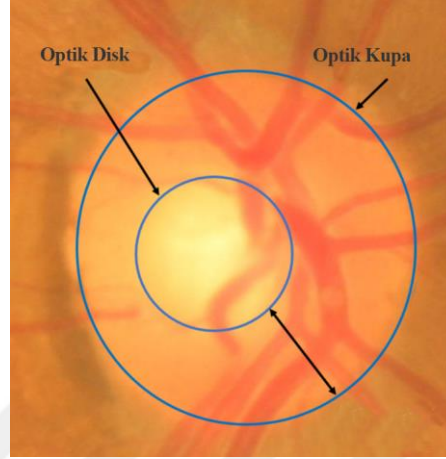
Şekil 2. 3. Fundus görüntü örneği

Fundus kamerası kullanılarak çekilen dijital fundus görüntüleri, diyabetik retinopati, glokom gibi göz hastalıklarının teşhisinde etkili bir şekilde kullanılabilir. Fundus görüntülerinde gözün retina, optik disk, kan damarları gibi klinik özellikleri net bir şekilde görüntülenebilmektedir. Ayrıca fundus kamerası güvenilirdir, daha az pahalıdır ve kullanımı kolaydır. Fundus görüntüleri gözün çukur-disk oranındaki değişiklik, optik sinir başı, çukur çapı vb. gibi çeşitli yapılarını ölçmek içinde kullanılmaktadır. Bu nedenle, fundus görüntüleri retina sağlığının ve göz anormalliklerinin teşhisi için uygun maliyetli bir araç olarak etkili bir şekilde tercih edilmektedir [15]. Deneyimli oftalmologlar, retina hastalıklarının her birinin varlığını veya yokluğunu tespit etmek için fundus kameraları veya tomografi cihazları tarafından çekilen retina görüntülerinden yararlanmaktadır. Bu manuel işlem, zaman alıcı, yorucu ve subjektif olduğundan bu tür teşhislerin tekrar edilebilirliğini zorlaştırmaktadır. Ayrıca az gelişmiş ülkelerde, özellikle kırsal kesimde, tıbbi uzmanlara ve altyapıya erişim sınırlıdır. Ayrıca retinal hastalıkların otomatik tespiti ve derecelendirilmesi, dünya çapında az sayıdaki göz doktorunun yükünü hafifletmek için de oldukça önemli olacaktır [16].

### 2.3. Glokom Anatomisi ve Fizyolojisi

Göz içi basıncı glokomu ortaya çıkaran en önemli mekanik risk etkenidir. Olağan popülasyonda ortalama 10-21 mmHg arasındaki basınçlar normal sayılmaktadır. GİB, gün boyunca sistematik kan basıncı, nabız ve solunumla ilişkili olarak değişkenlik gösterebilmektedir. Normal insanlarda genellikle gündüz özellikle de sabah saatlerinde göz içi basıncın en yüksek değerleri ölçülmektedir. Göz içi basınç değeri 21 mmHg üstünde ise glokom için mühim bir risk etkeni olarak düşünülmektedir. Glokomun optik sinir uçları üzerinde de bazı etkileri bulunmaktadır. Göz içi basınçtaki yükseliş, retina gangliyon hücrelerine baskı yaparak, sinir lifi katmanının beslenmesini bozmaktadır. Optik sinirin göz içindeki bölümüne optik sinir başı denilmektedir. Optik çukurlaşma,

optik diskin ortasındaki fonksiyonel nöral yapı içermeyen soluk alandır. Çukurluk disk oranı normal gözlerde büyük ölçüde 0.4 veya daha azdır. Bireyin her iki gözünde bu oran birbirine çok yakındır. Glokomun en birincil özelliklerinden birisi de çukurlaşma disk oranındaki büyük artıştır [17]. Optik disk ve optik kupaı gösteren örnek fundus görüntüsü Şekil 2.4'te gösterilmiştir [18].



Şekil 2. 4. Optik disk ve optik kupa örneği [18]

## 2.4. Glokom Sınıflandırılması

Glokom hastalığının epidemiyolojik, anatomik ve klinik gibi farklı sınıflandırmaları mevcuttur. Literatürde en sık görülen türleri Açık Açılı Glokom, Kapalı Açılı Glokom ve Konjenital glokom olmak üzere üçe ayrılmıştır [17].

### 2.4.1. Açık Açılı Glokom

Glokomun en yaygın görülen çeşididir. Bireylerin GİB değeri 21 mmHg'den büyük olması, retina gangliyon hücreleri ve bu hücrelerin aksonları olan retinal sinir lifi tabakası hasarı ile karakterize edilen ilerleyen bir optik nöropatidir. Özellikle gözdeki drenaj kanallarının yavaş yavaş tıkanmasından kaynaklanmaktadır. Glokomatöz hasarı sonucunda ciddi görme alanı ve görme keskinliği kayıpları oluşmaktadır. Hastalığın ilerlemesine neden olan en büyük risk faktörü yüksek GİB'dir. Açık açılı glokom erken dönemlerde belirti göstermediğinden tanı genellikle geri dönüşümsüz olup, görme hasarı meydana geldikten sonra konulmaktadır. Genellikle iki gözde görülür ve ilerleyen yaşla birlikte görülme oranı artış göstermektedir Taramalar ile erken tanı ve uygun tedavi yapılarak görmede belirgin kayıp gelişmeden hastalık tespit edilmektedir. Glokom şüphesi için göz hasarı olmaksızın GİB değerinin 21 mmHg'den büyük olması glokomun erken evresi olarak kabul edilmektedir [17]

#### 2.4.2. Kapalı Açılı Glokom

Daha az görülen glokom türüdür. Gözdeki drenaj kanallarının tıkanmasıyla göz içi basınçta ani bir artışa neden olmaktadır. Gözdeki iris ile korneanın arasındaki açının kapanması ile oluşur ve çok hızlı gelişmektedir [19].

#### 2.4.3. Konjenital Glokom

Bu tür glokom bebeklerde doğum öncesi dönemlerde gözün drenaj kanallarının yanlış veya eksik gelişmesinden kaynaklanmaktadır [19].

### 2.5. Literatür Taraması

Son yıllarda, araştırmacılar tarafından glokom hastalığının tespiti için birçok farklı çalışmada yapay zekâ tabanlı farklı metotlar kullanılmıştır.

Shanmuham ve arkadaşları glokomun erken teşhisi için 650'den fazla hastaneden alınan elektronik veri kayıtlarına XGBoost (ekstrem gradyan artırma yöntemi), MLP (çok katmanlı algılayıcı), RF (random forest) ve LR (lojistik regresyon) olmak üzere dört farklı makine öğrenimi modeli üzerinde test etmişlerdir. Random forest ve XGBoost modelleri %81 oranında doğruluk gösterirken, LR modeli %71 oranında doğruluk göstermiştir [20].

Prananda ve arkadaşları glokom hastalığı tespiti için retina sinir lifi tabakasındaki hasarı tespit eden alternatif bir model önermişlerdir. Halka açık Origa-Light veri kümesinden 168 glokom birey görüntüsü, 482 sağlıklı birey görüntüsü olmak üzere toplam 650 görüntü kullanılan çalışmanın ön işlem aşamasında optik disk ve kan damarları çıkarılmıştır. Sınıflandırma aşamasında dokuz derin öğrenme mimarisini kullanarak %89.34'lük AUC eğri altı alan ve %92.88'lik doğruluk elde etmişlerdir [21].

Dos Santos Ferreira ve arkadaşları RIM-ONE, DRIONS-DB ve DRISHTI-GS veri setlerini üzerine Evrimsel Sinir Ağları (ESA) modeli kullanarak glokom hastalığını tespit etmeye çalışmışlardır. Önerilen model temel olarak optik diskin saptanması, özelliklerin çıkarılması ve sınıflandırma aşamaları oluşmaktadır. Optik diskin bölütleme işleminden ve gri seviyelerden elde edilen görüntülerden kan damarları çıkarılmıştır. RGB kanalları için Otsu algoritmasının kullanıldığı çalışmada kırmızı kanalda %100 başarımla sağlandığı gözlemlenmiştir [22].

Simonthomas ve arkadaşları dijital fundus görüntülerinden Haralick doku özellikleri kullanarak glokom tespiti için yeni bir yöntem önermişlerdir. Sınıflandırma aşamasında K-En Yakın Komşu Algoritmasının (K-Nearest Neighbors-KNN) kullandığı modelde %98 doğruluk gözlemlenmiştir [23].

Akey Sungheetha ve Rajesh Sharma göz hastalıklarından diyabetik retinotapiyi tespit etmek için fundus görüntüleri kullanarak ESA tabanlı derin öğrenme mimarisi önermişlerdir. Önerilen

model %92 özgünlük ve %97 doğruluk göstermiştir. Diğer birçok modele göre başarı oranı oldukça yüksektir [24].

Juneja ve arkadaşları glaukoma hastalığının tespiti için 31 katmanlı U-net mimarisini değiştirerek yeni otonom bir sistem önermişlerdir. Ön işlem aşamasında fundus görüntülerini kırpma işlemi sonrasında görüntüler kırmızı, yeşil ve mavi kanallara ayrılmıştır. Görüntü segmentasyonu için en iyi kanal olan kırmızı kanal kullanılmıştır. Modelin test aşamasında 50 görüntü kullanılmış olup, optik disk (OD) ve optik kupa (OC) segmentasyonunda başarı oranı sırasıyla %95.3 ve %93.7 olarak tespit edilmiştir. Modelin diğer U-net mimarilerine göre başarı oranının yüksek olduğu gözlemlenmiştir [24].

Acharya ve arkadaşları dijital fundus görüntülerine gabor dönüşüm teknikleri uygulayarak yeni bir otonom glaukoma tespit yöntemi önermişlerdir. Toplam 510 görüntünün kullanıldığı çalışmada Destek Vektör Makinası (Support Vector Machine-SVM) sınıflandırıcısı ile 23 özelliğin belirlendiği çalışmada %93.10 doğruluk, %89.75 duyarlılık ve %96.20 özgüllük elde edilmiştir [26].

Saxena ve arkadaşları glaukoma tespiti için ESA mimarisi kullanmışlardır. ORIGA ve SCES veri setlerinin kullanıldığı çalışmada ORIGA veri kümesi için 0.82 AUC; SCES veri kümesi için 0.88 AUC değeri gözlemlenmiştir [27].

Prasad ve arkadaşları retinal fundus görüntülerinden glaukoma hastalığının tespiti için optik disk segmentasyonu ve kümeleme yöntemini kullanmışlardır. Optik disk segmentasyonu için eşik tabanlı kümeleme tabanlı ve bölge tabanlı yöntemler kullanmışlardır. Drions-db veri kümesinden 110 görüntü, hrf veri kümesinden 45 görüntü kullanılan modelde %98 doğruluk elde edilmiştir [27].

Dinç ve Kaya çalışmalarında bölge tabanlı derin ESA ve matematiksel bir yaklaşımı birleştiren melez bir mimari önermişlerdir. Önerilen modelde optik disk bölge tespiti için Faster R-CNN ile optik disk tespiti yapmışlardır. Halka açık dört farklı (ORIGA-light, DRIVE, DIARET-DB1 ve MESSIDOR) fundus görüntü veri kümesi kullanılmıştır. Veri setlerindeki görüntülerin %70 eğitim %30 test için ayrılmıştır. Sunulan model, ORIGA-light, DRIVE, DIARET-DB1 ve MESSIDOR veri tabanlarında optik disk saptanmasında sırasıyla %88.5, %75.5, %84.4 ve %86.8'lik başarı yakalayarak esnek bir performans göstermişlerdir [29].

Praveena ve GaneshBabu, çalışmalarında optik disk (OD) ve optik kupa (OC) oranını (CDR) hesaplamak için K-means algoritması, FCM bulanık kümeleme algoritması ve SWFCM uzamsal ağırlıklı bulanık kümeleme algoritmaları olmak üzere 3 farklı yöntem önermişlerdir. Veri kümesinde 50 normal fundus görüntüsü, 50 glaukoma fundus görüntüsü olmak üzere toplam 100 görüntü kullanılmıştır. En iyi performansı SWFCM tabanlı glaukoma hesaplama yöntemi gösterirken, en kötü performansı K-means yöntemi göstermiştir [30].

Kashyap ve arkadaşları toplam 650 veri kümesinden oluşan fundus görüntülerinden optik kupa segmentasyon işleminde U-Net mimarisi kullanılmıştır. Özellik çıkarımı için transfer öğrenme modeli olan DenseNet-201 mimarisi kullanıldığı çalışmada, eğitim ve test veri kümelerinde sırasıyla %98.82 ve %96.90 doğruluk elde etmişlerdir [31].

Shoukat ve arkadaşları G1020, RIM-ONE, ORIGA ve DRISHTI-GS veri kümelerini kullanarak glokomu tespit etmeye çalıştıkları araştırmada fundus görüntülerinin gri kanallarını kullanmışlardır. Sınıflandırma aşamasında ResNet-50 mimarisini kullanmışlardır. Veri kümesini artırmak için veri artırma tekniklerinin kullandığı çalışmada G1020 veri kümesinde önerilen modeli kullanarak %98.48 doğruluk, %99.30 duyarlılık, %96.52 özgüllük, %97 AUC ve %98 F1-ölçütü etmişlerdir [32].

Khan ve arkadaşları beş farklı sınıfa ait Optik Koherans Tomografi (Optical Coherence Tomography- OCT) görüntülerinden oluşan veri kümelerini kullanarak glokom hastalığını tespit etmek için yeni bir yöntem önermişlerdir. DenseNet-201, InceptionV3 ve ResNet-50 mimarilerini kullanarak görüntülerden özellikler çıkarıldıktan sonra karınca koloni optimizasyonu ile en uygun özellik seçimi yapmışlardır. Karınca koloni algoritması öncesi %97.4 olan doğruluk oranı elde etmişlerdir. Karınca koloni algoritması sonrası %99.1 doğruluk ile artan bir performans elde etmişlerdir [33].

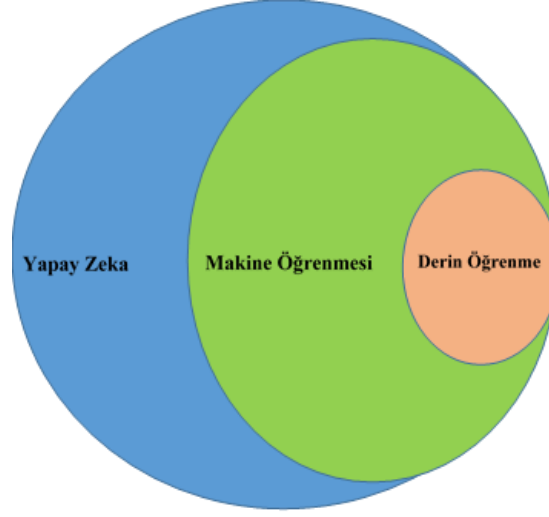
### 3. YAPAY ZEKÂ VE MAKİNE ÖĞRENMESİ

#### 3.1. Yapay Zekâ

Yapay zekâ terimi ilk olarak 1956 yılında, Amerika Birleşik Devletleri'nin Hampshire eyaletindeki Dartmouth Konferansında ortaya atılmıştır. Bu konferans akıllı davranışlar sergileyebilen makinalar yaratma fikriyle bir grup bilim adamını bir araya getirmiştir. Yapay zekâ, genellikle insan zekâsı gerektiren iş ve işlemleri yerine getirebilen bilgisayar sistemlerinin geliştirilmesini ifade eder. Bu iş ve işlemler görüntü tanıma, deneyimlerden öğrenme, dili anlama, karar verme ve problem çözme olarak sıralanabilir [34].

Yapay zekâ sistemleri problem çözme, örüntü tanıma, öğrenme ve karar gibi insanın bilişsel işlevlerini taklit etmek veya simule etmek için tasarlanmıştır. Yapay zekâ bu işlemleri doğal dil işleme, bilgisayarlı görme, robotik, makine öğrenmesi gibi birçok farklı teknikle yerine getirmektedir. Robotik dalı robotların tasarımını, yapımını, işlenmesini ve kullanımını içerir. Makine öğrenimi, ham verilerden kalıplar çıkarmak, bu verilerden insan girişi olmadan öğrenmek ve elde edilen bilgiyi çok sayıda uygulamada kullanmak için bilgisayar algoritmalarını kullanmaktadır. Derin öğrenme yöntemleri, sinir ağı katmanlarını kullanan bir makine öğrenimi türüdür. Temel makine öğrenimi modelleriyle kıyaslandığında, derin öğrenme modelleri birçok uygulama alanı için çok daha sağlamdır ve görüntüler ve videolar gibi büyük miktardaki verileri daha iyi işleyebilir. Doğal dil işleme, makine öğrenimi ve derin öğrenmeyi kullanan ve bilgisayar ile insan dili arasındaki etkileşimlerle ilgilenen bir yapay zekâ türüdür. Bu teknik, büyük miktarlarda doğal dil verilerini işlemek ve analiz etmek için kullanılır [34].

Yapay zekânın amacı, akıllı davranışlar sergileyebilen, yeni durumlara uyum sağlayabilen, deneyimlerden öğrenebilen ve zaman içinde performansını arttırabilen sistemler geliştirmektir. Günümüzde yapay zekanın sağlık, ulaşım, finans ve diğer birçok alan dahil olmak üzere farklı sektörlerde gerçek dünya uygulaması mevcuttur [34]. Yapay zekâ ve alt dallarının ilişkisel gösterimi Şekil 3.1'de gösterilmiştir.



Şekil 3. 1. Yapay zekâ ve alt dallarının şematik gösterimi

## 3.2. Makine Öğrenmesi

Makine öğrenimi yapay zekanın bir alt kümesidir. Makine öğrenimi algoritmaları, mevcut verileri bilinçli bir şekilde yorumlayarak karar veren sistemler olarak tasarlanmıştır. Makine öğrenim algoritmalarının uygulanmasının temel amaçlarından biri, geçmiş verileri almak ve bu verileri belirli özelliklerin tahmininde daha fazla kullanacak şekilde eğitmektir. Makine öğrenim algoritmalarının temel avantajı, elle kodlamaya kıyasla güçlü performansı ve veri işleme hızlarıdır. Makine öğrenim algoritmaları, konuşma tanıma, metin madenciliği, tıp, veri analizi, havacılık, veri analizi, borsa analizi ve diğerleri gibi çeşitli alanlarda performanslarını ve faydalarını kanıtlamıştır [35]. Makine öğrenmesi çalışmaları temelde denetimli öğrenme, denetimsiz öğrenme, yarı denetimli öğrenme ve takviyeli öğrenme olmak üzere dört gruba ayrılmaktadır [36].

### 3.2.1. Denetimli Öğrenme

Denetimli makine öğrenimi algoritmaları örneklerle öğrenmek üzere tasarlanmıştır. Denetimli öğrenme isminin nedeni, eğitim veri kümesinden öğrenme algoritması süreci, öğrenme sürecini denetleyen bir denetmen olarak düşünülebilir. Denetimli öğrenmede veriler bir giriş değişkeni ve çıkış değişkeninden oluşur veya veri kümesi etiketlenir. Etiketli veriler, giriş verilerinin ve bununla ilişkili çıktının veri kümesinde mevcut olduğu anlamına gelmektedir. Eğitim süreci sırasında algoritma, giriş verilerindeki kalıpları arar ve istenen çıkış verileriyle ilişkilendirir. Eğitim sürecinin ardından denetimli bir öğrenme algoritması, görünmeyen verileri girdi olarak alıp ve önceki eğitim verilerine dayanarak yeni girdilerin hangi etikete göre sınıflandırılacağını belirlemektedir. Denetimli öğrenmede modelinin amacı, yeni sunulan giriş verileri için doğru etiketi tahmin etmektir [36].

### 3.2.2. Denetimsiz Öğrenme

Denetimsiz makine öğrenimi yöntemleri, bir dizi girdiyi kullanarak etiketlenmemiş çıktıları kümeler ve analiz eder. Örneklerin kümelenebilmesi, etiketlenmemiş veri kümelerindeki gizli kalıpları, yapıları veya veri bilgisini ortaya çıkarmak için yaygın olarak kullanılmaktadır. Denetimsiz öğrenmede eğitim süreci veri örnekleri ve çıktıları arasındaki modeli daha doğru bir şekilde bulmaya yardımcı olabilecek en uygun parametrelerin tanımlanmasını içerir. Veri girişleri ve çıkışları arasındaki farkı hesaplamak için bir kayıp fonksiyonu tanımlanır. Denetimsiz modellerin eğitiminin amacı, eğitim setindeki kayıp fonksiyonunu azaltmak ve en iyi sonuçları elde etmektir. Küme tabanlı, boyut azaltma tabanlı ve sinir ağı tabanlı modellere ayrılmaktadır [37].

### 3.2.3. Yarı Denetimli Öğrenme

Yarı denetimli öğrenme, belirli öğrenme görevlerini gerçekleştirmek için etiketli ve etiketsiz verilerin kullanılmasıyla ilgilenen makine öğreniminin dalıdır. Kavramsal olarak denetimli ve denetimsiz öğrenme arasında yer alan bu sistem, birçok kullanım durumunda mevcut olan büyük miktardaki etiketlenmemiş verilerin tipik olarak daha küçük etiketli veri kümeleriyle birlikte kullanılmasına olanak tanımaktadır [38].

### 3.2.4. Takviyeli Öğrenme

Takviyeli öğrenme modelleri, bir aracıyı kendi deneyimlerinden ve eylemlerinden geri bildirim olarak etkileşimli bir ortamda öğrenmesi için eğiten başka bir makine öğrenimi kategorisidir. Takviyeli öğrenme, maksimum ödülleri almak için modelleri bir ortamda en uygun eylemleri öğrenecek şekilde eğitir. Bu optimal eylemler, bir ortamla etkileşimler ve onların tepkileri yoluyla gerçekleştirilebilir. Bir denetçinin yokluğunda, öğrencinin eylemlerinin sırasını bağımsız olarak keşfetmesi gerekir, bu da ödülleri en üst düzeye çıkarır. Takviyeli öğrenmenin avantajları arasında, dinamik ve karmaşık ortamlarla baş edebilme yetenekleri yer alır [37].

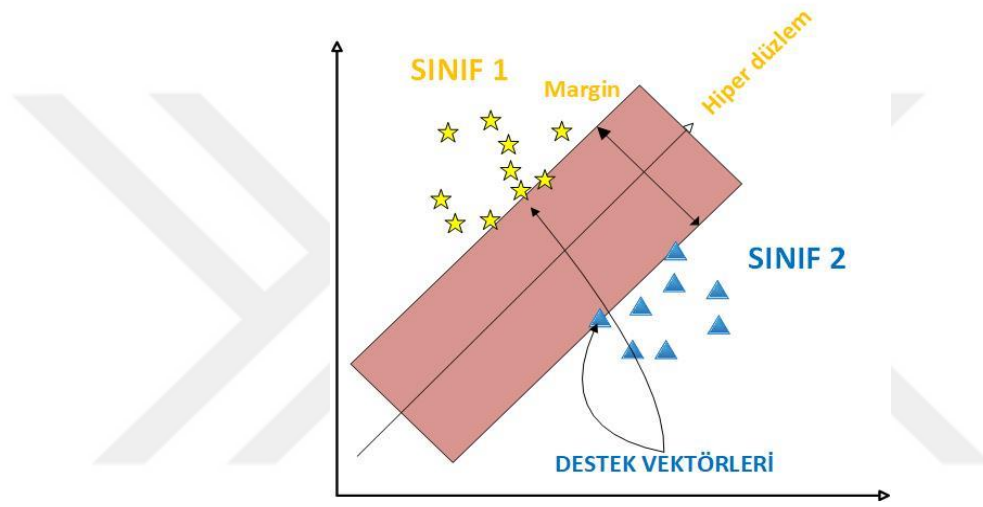
## 3.3. Makine Öğrenmesi Algoritmaları

Bu tez çalışmasında literatürde sıklıkla kullanılan Destek Vektör Makineleri (SVM-Support Vector Machine-SVM), K-En Yakın Komşu (KNN-K-Nearest Neighbors), Naive Bayes, Topluluk Öğrenme, Karar Ağacı ve Lojistik Regresyon algoritmaları kullanılmıştır.

### 3.3.1. Destek Vektör Makineleri

Destek Vektör Makineleri, verileri hiper düzlemlerle ayırmak için daha yüksek bir boyuta dönüştüren popüler bir sınıflandırma ve regresyon algoritmasıdır. Aynı zamanda bilgisayarlı görme

ve veri madenciliği alanlarında da geniş uygulamalara sahiptir. Bir vektör uzayındaki iki kümenin en uygun sınırını, kümelerdeki eğitim vektörlerinin olasılıksal dağılımlarından bağımsız olarak elde etmeye yönelik bir yöntemdir. Bir sınıflandırma bağlamında, SVM, iki sınıf arasındaki marjı maksimuma çıkarırken farklı veri noktası sınıflarını en iyi şekilde ayıran hiper düzlemi bulmaya çalışır. Kenar boşluğu, hiper düzlem ile her sınıftan en yakın veri noktaları arasındaki mesafedir. SVM, yalnızca sınıfları ayırmakla kalmayıp aynı zamanda bu marjı maksimuma çıkararak, teorik olarak aşırı uyum riskini azaltan ve modelin genelleme yeteneklerini artıran hiper düzlemi bulmayı amaçlar [39-41]. SVM şematik gösterimi Şekil 3.2’de gösterilmiştir.



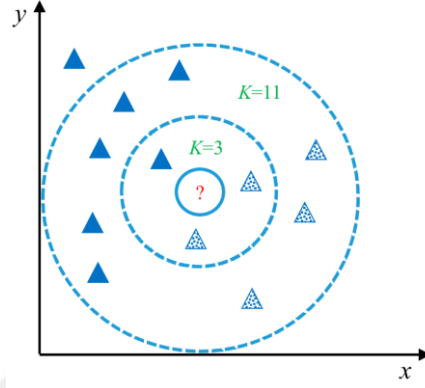
Şekil 3. 2. Destek vektör makinesi şematik gösterimi

Hiper düzleme en yakın olan ve konumunun belirlenmesinde en fazla etkiye sahip olan veri noktalarına “destek vektörleri” adı verilmektedir. Bunlar, kenar boşluğunun kenarlarında bulunan veya kenar boşluğunun kayması durumunda yanlış sınıflandırılabilir veri noktalarıdır. SVM bu destek vektörlerine odaklanır ve aykırı değerlerin veya gürültülü verilerin varlığında bile onu güçlü bir algoritma haline getirmektedir. SVM’ler, verileri örtülü olarak daha yüksek boyutlu alanlara eşleyebilen çeşitli çekirdek işlevlerinin kullanımı yoluyla farklı veri türleriyle ilgilenme konusunda esneklik sunmaktadır. Bu, SVM’lerin değişkenler arasındaki doğrusal olmayan ilişkileri yönetmesine olanak tanımaktadır [41].

### 3.3.2. K-En Yakın Komşu Algoritması

K-En Yakın Komşu (K-Nearest Neighbor-KNN) algoritması denetimli bir sınıflandırma algoritmasıdır. Çeşitli sınıflara ayrılmış veri noktalarını tanımlar ve yeni bir örnek veri noktası için sınıf etiketini tahmin etmektedir. Veri nesnelerini bir özellik uzayındaki en yakın eğitim örneklerine

göre sınıflandırmak için bilinen bir yöntemdir. KNN'deki K, sınıflandırıcının tahminini yapmak için kullanacağı en yakın komşuların sayısını ifade etmektedir. Bilinmeyen veri noktaları, seçilen 'K' en yakın komşuların çoğunluk oylarıyla sınıflandırılmaktadır [42-44]. KNN algoritmasının sınıflandırma örneği Şekil 3.3'te gösterilmiştir [45].



Şekil 3. 3. KNN algoritması sınıflandırma örneği [45]

Nesneler arasındaki mesafenin matematiksel modeli Denklem 3.1'de gösterilmiştir. KNN, en yakın komşuları bulmak için Kosinüs, Öklid ve Manhattan gibi farklı mesafe ölçümlerini kullanmaktadır [42-44].

$$d_{(i,j)} = \sqrt{\sum_{k=1}^p (X_{ik} - X_{jk})^2} \quad (3.1)$$

### 3.3.3. Naive Bayes Algoritması

Naive Bayes sınıflandırıcı denetimli bir öğrenme algoritmasıdır. Naive Bayes modelinde, tahminlerin üretilmesi için ortak olasılık dağılım fonksiyonu ve bağımsızlık varsayımları kullanılmaktadır. Naive Bayes modelinin en önemli özelliklerinden biri eğitim süresinin önemli ölçüde kısa olmasıdır. Özellikler arasında bağımsızlığı varsayan Naive Bayes sınıflandırıcı, Bayes sınıflandırma yöntemlerini kullanmaktadır. Bayes teoreminin formülü Denklem 3.2'de gösterilmiştir.

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)} \quad (3.2)$$

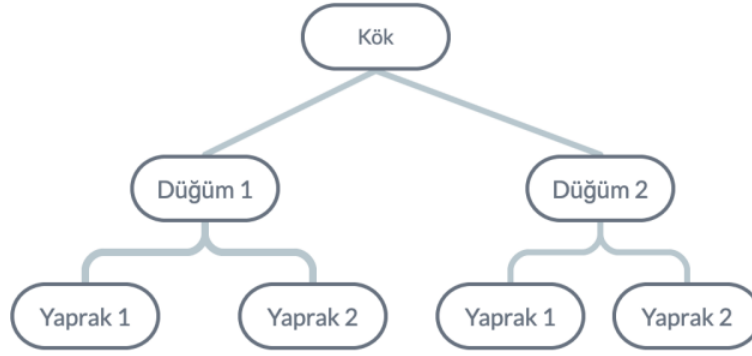
Burada  $P(A|B)$ ; B olayı gerçekleştiği durumda A olayının meydana gelme olasılığıdır.  $P(B|A)$ ; A olayı gerçekleştiği durumda B olayının meydana gelme olasılığıdır.  $P(A)$  ve  $P(B)$ ; A ve B olaylarının önsel olasılıklarıdır [46].

### 3.3.4. Topluluk Öğrenme Algoritması

Toplu öğrenme olarak bilinen çoklu sınıflandırıcı sistemi, gelişmiş sınıflandırma doğruluğu elde etmek için farklı sınıflandırıcıların eğitilmesini ve tahminlerinin birleştirilmesini içermektedir. Bir sistemde birden fazla sınıflandırıcı kullanmak, tek sınıflandırıcı sonuçlarından daha iyi performans göstermektedir. Topluluk yöntemi, tahminlerini tek bir yapıya dayanarak sınıflandırma yapan diğer öğrenme yaklaşımlarının aksine, bir grup yapıyı birleştirmeye çalışmaktadır [47]. Topluluğun ana öğrenme türleri arasında torbalama, artırma ve istifleme yer almaktadır. Torbalama genel varyansı azaltmaya odaklanırken, artırma ve istifleme model yanlılığını azaltmaya odaklanmaktadır [48]. Torbalama tekniği, veri kümesinin farklı alt kümeleri tarafından eğitilen bir dizi temel öğrenciyi kullanmaktadır. Güçlendirme, farklı veri alt kümelerine farklı ağırlıklar atayarak sıralı bir düzende düzenlenmiş zayıf öğrenenlerin performansını artırmaya çalışmaktadır. Nihai tahmin, her öğrencinin çıktısının birleştirilmesiyle elde edilmektedir [49,50].

### 3.3.5. Karar Ağacı Algoritması

Karar ağaçları, karar analizi gerçekleştirmek için kullanılan analitik bir tekniktir. Büyük veri kümelerinde bulunan ilişkileri, kalıpları ve kuralları aramak ve modellemek için kullanılırlar. Bir karar ağacının yapısı, kök düğümden başlayarak, ayırma kriteri, durdurma kuralı, budama vb. temel alınarak her dal uç düğüme ulaşıncaya kadar alt düğümlere devam eden düğümlerden oluşur. Ağacın sonunda yer alan ve daha fazla dallanmanın olmadığı düğüme uç düğüm, kök düğümden uç düğüme olan mesafeye ise derinlik adı verilir. Karar ağacı analizi süreci, karar ağacının oluşturulmasından, budama fizibilite değerlendirmesi ve yorumlanması yoluyla tahmine kadar sırayla ilerler [51]. Ayrıca karar ağacı, kullanıcıların bir ağaç yapısını takip ederek kararın nasıl alındığını görmesini kolaylaştıran şeffaf bir mekanizmadır. Ağacın temel amacı, çok sayıda girdi değişkenine dayalı olarak gerekli bir değişkenin değerini hesaplayan bir model üretmektir [52]. Karar ağacı algoritmasının şematik gösterimi Şekil 3.4'te verilmiştir [53].



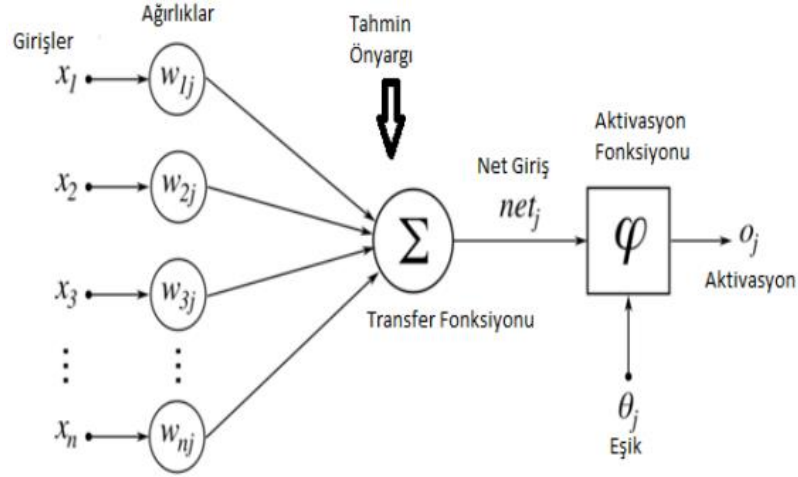
Şekil 3. 4. Karar ağacı algoritması şematik gösterimi [53]

### 3.3.6. Lojistik Regresyon Algoritması

Lojistik regresyon, makine öğreniminde popüler olan ve doğrusal regresyona benzeyen denetimli bir sınıflandırma algoritmasıdır. Lojistik fonksiyonu bağlantı fonksiyonu olarak kullanan genelleştirilmiş doğrusal bir modeldir. Sonlu sayıda sonucun olasılığını modellemek için kullanılır. Regresyon katsayılarının optimizasyonuna dayanarak bir olayın meydana gelme veya gelmeme olasılığını tahmin etmektedir. Bu sonuç her zaman 0 ile 1 arasında değişmektedir. Tahmin edilen değer bir eşik üzerinde olduğunda olayın meydana gelmesi muhtemeldir, ancak bu değer aynı eşik altında olduğunda ise gerçekleşmeyeceği anlamına gelmektedir [54].

### 3.4. Yapay Sinir Ağları

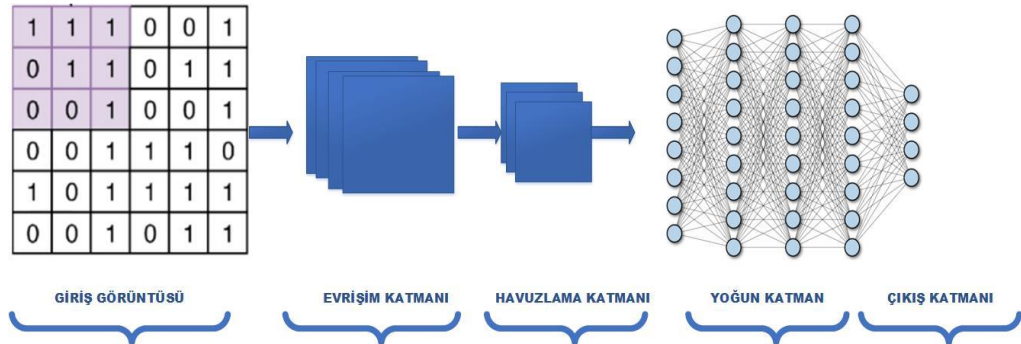
Yapay sinir ağları, biyolojik sinir ağlarından ilham alan makine öğrenme modelleridir. Bu hesaplamalı modeller, kalıpları tanımak ve verilerden ilişkileri çıkarmak için özel olarak tasarlanmıştır. Sinir ağları, nöron adı verilen alt sistemlerden oluşur. Her nöron, nöronun girişlerini çıkışıyla ilişkilendiren bir aktivasyon fonksiyonuna sahiptir. Bir YSA birbirine bağlı birçok nörondan oluşur. Bunlar bir giriş katmanında ve girişleri ağın çıkışlarına bağlayan bir veya daha fazla nöronun bulunduğu bir ara katmanda düzenlenir. Nöronlar arasındaki bağlantılar, bir nöronun çıktısını bir sonraki katmandaki nöronun girdisiyle birlikte değerlendiren ağırlıklardan oluşur. Diğer makine öğrenim algoritmaları gibi sinir ağları da bir eğitim süreci gerektirir. Bu aşamada ağ, bir giriş çıkış veri kümesinden bir sistemi modelleyebilir. Eğitim, ağ parametrelerinin yinelemeli olarak hesaplanmasından oluşur. En yaygın kullanılan eğitim yöntemi geri yayılım algoritmasıdır. Oldukça etkili ve hesaplama açısından verimli bir gradyan iniş algoritmasıdır [55]. Şekil 3.5'te yapay sinir hücresinin yapısı gösterilmiştir [56].



Şekil 3. 5. Yapay sinir hücresi yapısı [56]

### 3.5. Evrişimsel Sinir Ağları

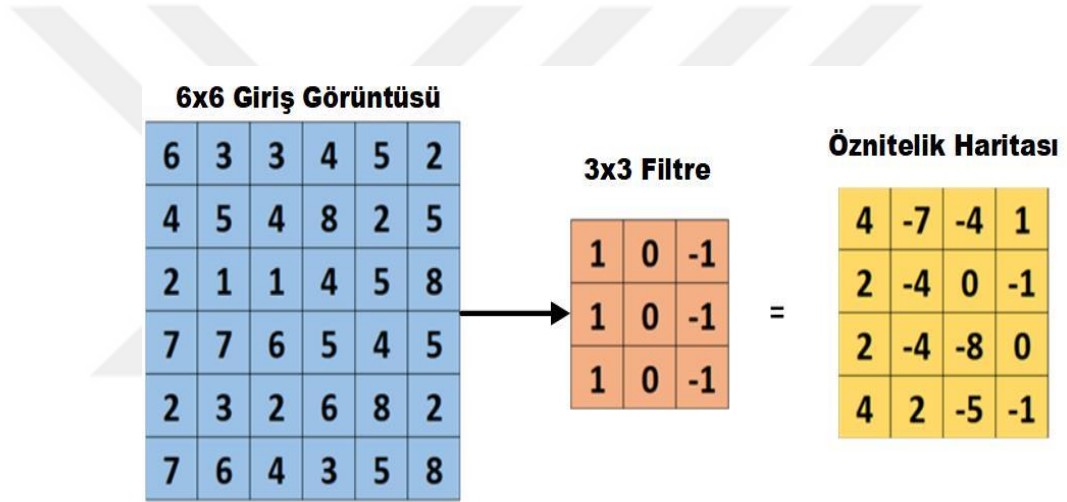
ESA'lar, özellikle görüntüler ve videolar gibi çok boyutlu veri alanlarında yeterlilikleri ile tanınan güçlü bir sinir ağı sınıfıdır. Bunlar belirli bir ileri beslemeli sinir ağı türüdür. CNN'lerin mimarisi, özellikle verilerden ilgili özelliklerin çıkarılmasında usta olan evrişim ve havuzlama katmanlarından oluşur. Daha sonra, bir veya daha fazla tamamen bağlantılı katman ile özelliklerden tahmin işlemi gerçekleştirilir. Eğitim süreci sırasında CNN'ler, çıkarılan özellikleri uygun etiketlerle ilişkilendirme yeteneğine sahiptirler. ESA'ların başarımın göstermesinde, geri yayılım ve optimizasyon gibi yinelemeli mekanizmaları kullanması önemli bir faktördür. Tipik bir ESA, giriş, evrişim, havuzlama, tam bağlantılı ve çıkış katmanları dahil olmak üzere birkaç temel katmandan oluşur. ESA'lar çok boyutlu giriş özelliği vektörleri üzerinde çalışacak şekilde tasarlanmıştır. Aynı zamanda, evrişimli sinir ağlarının parametre paylaşımı ve yerel bağlantı avantajlarından dolayı ağ hesaplama açısından çok daha az yoğun, işletimi daha hızlıdır ve büyük ölçekli görüntü verilerini işleme kapasitesine sahiptirler. Evrişimsel sinir ağlarının mimarisi Şekil 3.6'da gösterilmiştir.



Şekil 3. 6. Evrişimsel sinir ağlarının mimarisi

### 3.5.1. Evrişim Katmanı

Evrişim katmanı, özellikleri çıkarmak amacıyla giriş görüntüsü üzerinde evrişim işlemlerini gerçekleştirmek için filtreler kullanan CNN'nin temel kısmıdır. Bir filtre, giriş görüntüsünden belirli özelliklere sahip görüntü bölgelerini çıkarmak için kullanılan küçük, öğrenilebilir bir ağırlık matrisleri seti olarak anlaşılabilir. Evrişimsel katmanlar, ağıdaki parametre sayısını önemli ölçüde azaltabilen ve öğrenme verimliliğini ve genelleme yeteneğini artırabilen parametre paylaşımı ve yerel bağlantı ile karakterize edilir. Evrişimli katmanlar, özellik haritalarını çıkararak veriler içindeki içsel ilişkileri belirlemede çok önemli bir rol oynar. Bu özellik haritaları daha sonra ağ karmaşıklığını azaltmak ve aşırı uyumu önlemek için alt örnekleme gerçekleştiren havuzlama katmanları aracılığıyla işlenir [57,58]. Evrişim katmanında giriş görüntüsüne 3x3 filtre uygulanması işlemi Şekil 3.7'de gösterilmiştir [59].



Şekil 3. 7. Evrişim katmanında filtre uygulanması [59]

### 3.5.2. Havuzlama Katmanı

Havuzlama katmanı, evrişim katmanının çıktısı özellik haritasının boyutunu azaltmak, böylece parametre sayısını azaltmak ve modelin genelleme yeteneğini geliştirmek için kullanılır. Yaygın olarak kullanılan havuzlama yöntemleri arasında, daha küçük boyutlu bir özellik haritası elde etmek için özellik haritasının yerel bölgelerinde toplama işlemlerinin gerçekleştirilmesini içeren maksimum havuzlama ve ortalama havuzlama yer alır. Havuzlama katmanının parametreleri genellikle evrişim çekirdeğinin boyutunu ve adım boyutu gibi içerikleri barındırır [57]. Havuzlama işleminin 2x2 şematik diyagram örneği Şekil 3.8'de gösterilmiştir [60].

4	-2	0	5	-1	<b>2x2 havuzlama boyutu</b>	4	5	-1	
-5	1	2	0	-3		<b>Maksimum Havuzlama</b>	3	0	3
2	3	0	-1	2			2	3	4
-3	0	-2	-1	3		<b>Ortalama Havuzlama</b>	-0.5	1.75	-1
2	-1	0	3	4			0.5	-1	1.25
						0.25	0.75	1	

Şekil 3. 8. Havuzlama işlemi [60]

### 3.5.3. Tam Bağlantı Katmanı

Tamamen bağlı katman, genellikle önceki evrişim ve havuzlama katmanlarından gelen özelliklerin sınıflandırılması veya regresyon tahmini için kullanılan standart bir sinir ağı yapısıdır. Tam bağlantılı katmanda, her nöron bir önceki katmandaki tüm nöronlara bağlıdır, dolayısıyla tam bağlantılı katman parametrelerinin sayısı büyüktür ve aşırı uyum eğilimi gösterir [57].

### 3.5.4. Yoğun Katmanı

Yoğun katman yoğun bir şekilde bağlanmıştır; bu, bir katmanın nöronlarının bir önceki katmanın her nöronuyla eşleştiği anlamına gelir. Yoğun bir katman, bir önceki katmanın tüm çıktılarını besler ve bir sonraki katmana çıktılar sağlar [61].

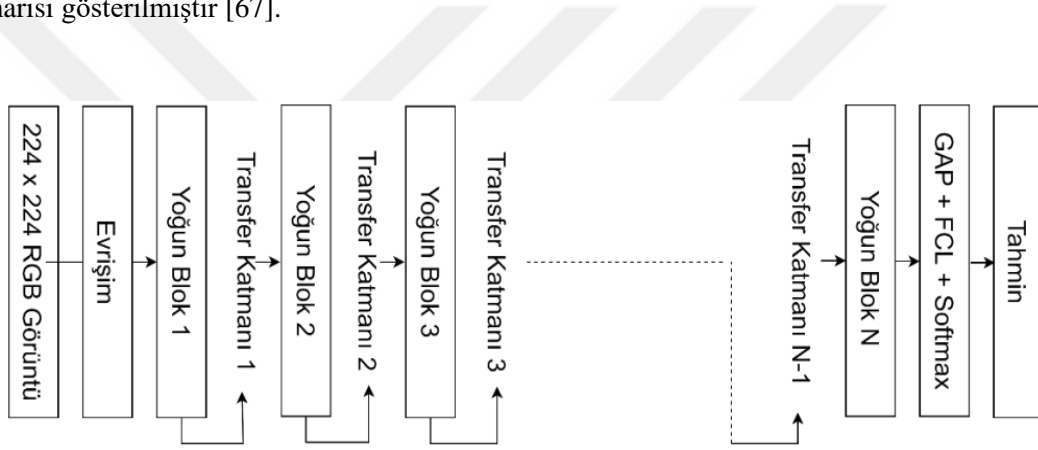
## 3.6. Derin Öğrenme Mimarileri

Özellikle son birkaç on yılda sağlık alanındaki hastalık tespiti başta olmak üzere birçok tıbbi uygulama için derin öğrenme mimarileri kullanılmaya başlanmıştır. Makine öğreniminin bir parçası olan derin öğrenme özellikle görüntü sınıflandırma, segmentasyon ve tahmin etme gibi alanlarda büyük başarı göstermektedir. Derin öğrenme mimarileri borsa, endüstri, sağlık vb. gibi birçok alanda iyi sonuçlar vermektedir. Derin öğrenme mimarileri temel olarak bir dizi gizli katmana sahip yapay sinir ağlarından oluşmaktadır [62,63]. Bu tez çalışmasında DenseNet201 ve Inceptionv3 mimarisi kullanılmıştır.

### 3.6.1. DenseNet201 Mimarisi

DenseNet modeli yoğun bir bağlantı mekanizması, yani katmanların birbirine bağlanmasına yönelik bir mekanizmadan oluşmaktadır. Spesifik olarak, her katman önceki tüm katmanları ek girdi olarak kabul etmektedir. Bu model, evrişimli ağların daha derinlemesine, doğru ve verimli bir şekilde eğitilmesine olanak tanımaktadır. DenseNet mimarilerinde her katman, kanal

boyutunda önceki tüm katmanlara bağlanmaktadır. Yoğun blok her biri aynı boyutta özelik haritasına sahip katmanların birbirine bağlı olduğu modüllerdir. Katmanlar birlikte bir sonraki katmana girdi görevi görmektedir. DenseNet mimarileri, tüm katmanları doğrudan birbirine bağlayarak ağıdaki katmanlar arasında maksimum bilgi aktarımını sağlar, özelliklerin aktarımını artırır, daha verimli kullanır ve parametre sayısını bir miktar azaltır. DenseNet modelinin mükemmel yapısal tasarımı nedeniyle, özelliğin yeniden kullanımı iyileştirilir ve gradyan dağılım sorunu daha iyi çözülürken, ağ parametrelerinin sayısı da büyük ölçüde azaltılır. DenseNet'in birçok farklı sürümü mevcuttur. Toplam 201 katmana ve 20 milyon öğrenilebilir parametreye sahiptir. Ağın boyutu 80 mb olup  $224 \times 224$  görüntü giriş boyutunu kabul etmektedir. DenseNet kullanmanın diğer mimarilere göre en büyük avantajı, görüntülerin düşük seviyeli özelliklerinin yakalanmasında diğer modellere göre daha verimli olmasıdır [64-66]. Şekil 3.9'da DenseNet201 mimarisi gösterilmiştir [67].

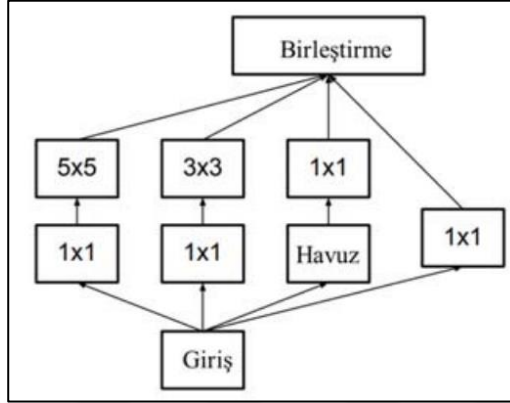


Şekil 3. 9. DenseNet201 mimarisi [67]

### 3.6.2. Inceptionv3 Mimarisi

İlk Google araştırmacıları tarafından geliştirilen inceptionv3, Inception model serisine ait olup, derin bir evrişimli sinir ağı yapısına sahiptir. Görüntü tanıma ve sınıflandırma atamalarında başarıyla kullanılan ve tercih edilen bir mimaridir. Bu mimari derin yapısı ve Inception modüllerinin kullanımı ile popülerdir. Bu modüller, farklı filtre boyutlarına sahip paralel evrişimli katmanlardan oluşur ve ağın birden fazla ölçekte özellikleri yakalamasına olanak tanır. Bu mimari paralel dalları birleştirerek, bir görüntüdeki hem yerel hem de küresel özellikleri etkili bir şekilde yakalayabilmektedir. Inceptionv3'teki en önemli yeniliklerden biri, darboğaz katmanları görevi gören  $1 \times 1$  evrişimlerin kullanılmasıdır. Bu  $1 \times 1$  evrişimler, giriş kanalı sayısını ve hesaplama karmaşıklığını azaltmaya yardımcı olarak ağı daha verimli hale getirmektedir. Aynı zamanda ağın görüntüdeki hem yerel hem de küresel özellikleri yakalamasına olanak tanıyarak, karmaşık kalıpların tanınmasında daha doğru olmasını sağlar. Ağın aynı modülü içindeki  $1 \times 1$ ,  $3 \times 3$  ve  $5 \times 5$  evrişim filtrelerinin çıktuları daha sonra üst üste yığılır ve ağıdaki bir sonraki katmana beslenir. 48

katmandan oluşan inceptionv3 mimarisi giriş görüntü boyutunu 299x299x3 piksel olarak almaktadır [68-71]. Inceptionv3 mimarisinde bulunan bir Inception modülünün standart başlangıç modülü yapısı Şekil 3.10'da gösterilmiştir. Bu mimari genel olarak bu tür modüllerin çoklu olarak birleşiminden oluşmaktadır [72].



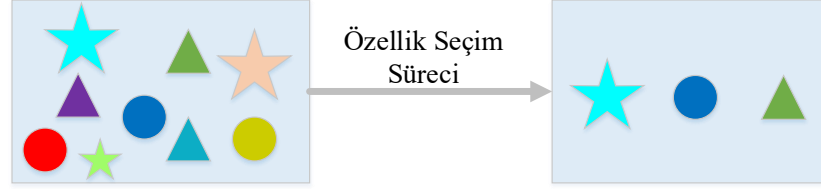
Şekil 3. 10. Standart bir başlangıç modülü yapısı [72]

### 3.7. Metasezgisel Optimizasyon Algoritmaları

Günümüz teknoloji dünyasında verinin önemi ve büyümesi hızla artmakta ve bu veriler sağlık sektöründe hastalıklar ile ilgili önceden tahmin etmemizi sağlayacak değerli bilgiler sağlama konusunda hayati bir rol oynamaktadır. Genel olarak e-kayıtlarda üretilen sağlıkla ilgili tüm veriler, birçok hastanın sağlık geçmişinden elde edilmektedir. Yapay Zekâ tabanlı bir sistem, alan farkındalığı ve sınıflandırma sinir ağlarının yanı sıra bu sağlık hizmeti verilerinden alınan arka uç desteğiyle tıp uygulayıcılarına insan hastalıklarının mümkün olduğunca erken tespit edilmesi ve sınıflandırılmasına yardımcı olur. Çoğu durumda, bu hastaların sağlık kayıtları, spesifik hastalıkla ilgili bireysel değerleri belirten çok sayıda nitelik veya özellikten oluşur. Tüm özellik seti içerisinde bazıları, yinelenen özellikler içeren gereksiz özellikler ve belirli bir sorun alanıyla ilgili olmayan ilgisiz özelliklerdir. Bu gereksiz ve ilgisiz özellikler, sistemin hastalıkları öngörmesinde veya sınıflandırmasında hiçbir zaman yardımcı olmaz ve nihai sonucun doğruluğunu iyileştirmek için kaldırılabilir [73].

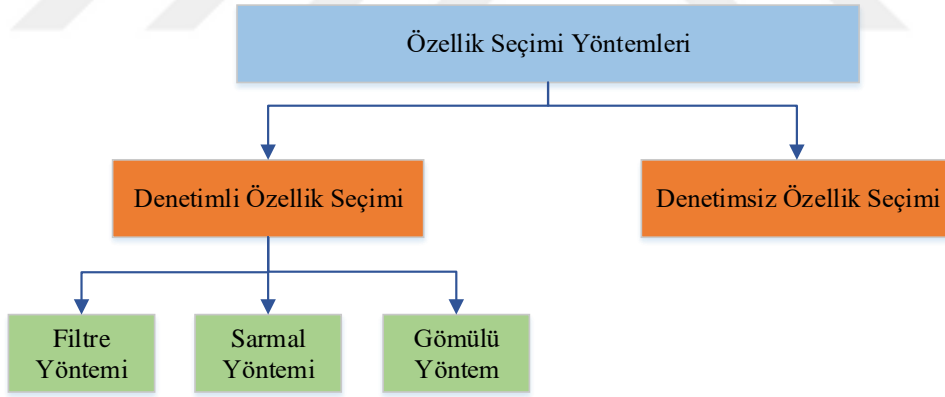
Hastalık sınıflandırmasını veya tahmin doğruluğunu artıran en iyi özellik kümesini bulma süreci, genel hastalık sınıflandırma sürecinde önemli bir adımdır. Özellik seçimi, seçilen problemle ilgili özellik kümelerinin seçilmesi işlemidir; bu, ilgili özelliklerin dahil edilmesine ve alakasız veya gereksiz özelliklerin kümeden hariç tutulmasına yardımcı olur, böylece gürültülü verileri ve tüm veri kümesinin boyutunu azaltır. Özellik seçimi problemi, veri madenciliğinde en zorlu görevlerden biridir. Bir küme  $n$  sayıda özellik içeriyorsa, en iyi alt kümenin seçilmesi gereken toplam  $2^N$  alt küme mümkündür.  $N$ 'nin büyük bir sayıya yönelmesi en zoru olacaktır, çünkü

modelin performansını her bir alt kümede değerlendirmek mümkün değildir. Özellik seçimi sırasında büyük veri kümesindeki nitelikleri en aza indirme ideolojisi Şekil 3.11’de gösterilmektedir.



Şekil 3. 11. Özellik seçim süreci

Genel olarak özellik seçimi modelleri, denetimli modeller ve denetimsiz modeller olmak üzere iki temel türe ayrılmaktadır. Denetimli metodolojide özellik seçimi için çıktı etiketi sınıfı uygulanırken, denetimsiz öğrenme modelinde özellik seçimi için herhangi bir çıktı etiketi sınıfı uygulanmamaktadır. Ayrıca, denetimli öğrenme modelleri genel olarak Filtre Yöntemi, Sarmal Yöntemi ve Gömülü Yöntemler olmak üzere üç alt türe ayrılır. Özellik seçimi yöntemleri Şekil 3.12’de gösterilmiştir.



Şekil 3. 12. Özellik seçimi yöntemlerinin çeşitleri

Filtre yaklaşımları, özelliklerin değerlendirilmesi ve sıralanması için verinin kendine özgü özelliklerini kullanmaktadır. Sınıflandırıcıdan bağımsız yöntemler olarak da bilinen filtre yaklaşımları, herhangi bir belirli öğreniciyi dahil etmeden verilerden özellikler çıkarmaktadır. Bu yöntemler mesafe, bilgi, bağımlılık ve tutarlılık olmak üzere dört farklı değerlendirme kriterine göre değerlendirilmektedir. Bu yaklaşım, eğitim verilerinin ortak özelliklerine bağlı olarak uygun özellikleri seçer. Filtre yöntemleri hızlı, hesaplama açısından basit ve ekonomiktir; bu da onları yüksek boyutlu veri kümesi problemlerinin çözümünde daha çok tercih edilmesine neden

olmaktadır. Bu yöntemlere örnek olarak; Korelasyon katsayısı, Ki-kare testi, Fisher skoru ve Bilgi kazanımı verilebilir [74].

Sarmal yöntemler, bir öğrenme algoritmasının değerlendirme kriterlerine dayalı olarak bir özellik alt kümesini seçmektedir. Ayrıca sarmal yöntemler doğası gereği sıralı veya rastgele olabilir. İleriye doğru özellik seçimi ve geriye doğru eleme yöntemi gibi sıralı sarmal yöntemlerinde, özellikler seçilen alt kümeye sırasıyla artımlı olarak dahil edilir ve çıkarılır. Ancak, bu şekilde seçilen/kaldırılan özellikler, bu yöntemlerle karar sisteminin performansını artırmak amacıyla daha sonraki yinelemelerde kaldırılamaz/seçilmez. Rastgele aramaya dayalı sarmalayıcı yöntemler, özellik alanını keşfetmek için geliştirilmiş bir strateji sağladıklarından sıralı sarmalayıcı yöntemlerin sınırlamalarının üstesinden gelmektedir. İyi bilinen rastgele aramaya dayalı sarmalayıcı yöntemlerden bazıları Genetik Algoritma, Karınca Kolonisi Optimizasyonu ve Parçacık Sürüsü Optimizasyonudur. Bu yöntemlerde, nesiller boyunca en uygun çözüme doğru yaklaşan olası çözümlerden oluşan bir popülasyon oluşturulur [74].

Son tür, özellik seçimi sürecinin sınıflandırma veya öğrenme ile birlikte bağlandığı gömülü yöntemler olarak adlandırılır. Gömülü yöntem, hem filtre hem de sarmal yöntemlerinde karşılaşılan zorlukların üstesinden gelmek için tanıtılmıştır. Bu gömülü yöntem, sarmal yöntemine benzer şekilde daha hızlıdır ve daha yüksek doğrulukta sonuçlar üretir. Örneklerden bazıları Lasso regresyonu, Ridge regresyonu, Karar ağacı ve Rastgele orman algoritmasıdır [74].

Geleneksel olarak özellik seçimi süreci alt küme oluşturma, alt küme değerlendirmesi, durdurma kriteri ve sonuç doğrulama olmak üzere dört temel adımı içermektedir. Dört adım arasında, alt küme oluşturma ve alt küme değerlendirmesi, bir özellik seçim algoritmasının oluşturulmasında iki önemli faktördür. Alt küme oluşturma, tüm olası özellik alt kümelerinin en iyi alt kümesini seçmeyi amaçlayan bir arama problemi olarak kabul edilebilir. Kapsamlı arama, sezgisel arama ve ağaçlı arama gibi çeşitli arama stratejileri, araştırmacılar tarafından optimum veya optimumun altındaki bir özellik alt kümesini tanımlamak için ağırlıklı olarak araştırılmış ve kullanılmıştır [75]. Kapsamlı strateji, tüm olası özellik kombinasyonlarını araştırırken, "kombinatoriyal patlama", özelliklerin sayısıyla birlikte katlanarak artan bir hesaplama yüküne yol açmaktadır. Sezgisel arama stratejileri, özellik çıkarımı sorunlarını çözmek için Genetik algoritma, Simüle tavlama ve Gri Kurt Optimize Edici gibi metasezgisel algoritmaları ve diğer birkaç optimizasyon tekniğini kullanmaktadır.

Optimizasyon, belirli bir sistem sorununu çözmek için en iyi performansı elde etmek amacıyla birden fazla yaklaşım arasında karar verme sürecidir. Başka bir deyişle optimizasyon, bir amaç fonksiyonunun tahminini göz önünde bulundurarak birden fazla alternatif seçenek arasından en iyi karar seçme süreciyle ilgilidir [76].

Pek çok optimizasyon probleminde, oldukça karmaşık kısıtlamalar altında belirli bir problemin optimal çözümünü makul bir süre içinde bulmak gerekmektedir [77]. Bu tür

optimizasyon sorunları genellikle modern akıllı yöntemlerle çözülmektedir. Bu sorunları çözmek için çeşitli stratejiler önerilmektedir, ancak bunlar daha iyi sonuçlar elde etmek için yeterli değildir. Son yıllarda araştırmacılar, bilimsel topluluklarda, özellikle birçok karmaşık optimizasyon probleminin çözümü için metasezgisel optimizasyon algoritmalarına çok fazla ilgi göstermektedir. Tepe Tırmanışı, Rastgele Arama ve Tavlama Simülasyonu [78], metasezgisel algoritmalarından önce optimizasyon problemlerini çözmek için kullanılan geleneksel algoritmalar olarak kabul edilmektedir. Geleneksel algoritmalarda arama tek noktadan başlatılarak ve global optimuma ulaşmak zaman alan gradyan bilgisine ihtiyaç duymaktadır. Sınırlamaları ve karmaşıklığı göz önüne alındığında, bu algoritmalar yerelleştirme sorunu, ekonomik optimizasyon, yapısal optimizasyon sorunları ve mühendislik tasarım sorunları gibi gerçekçi uygulamaların çözümünde çok etkili değildir [79].

Metasezgisel algoritmalar, optimizasyon problemlerinin optimal (optimuma yakın) çözümünü elde eden optimizasyon yöntemleridir. Bu algoritmalar türevisiz tekniklerdir ve basitliğe, esnekliğe ve yerel optimumlardan kaçınma yeteneğine sahiptirler. Metasezgisel algoritmaların davranışı stokastiktir; optimizasyon sürecine rastgele çözümler üreterek başlarlar. Gradyan arama tekniklerinde olduğu gibi arama uzayının türevinin hesaplanmasını gerektirmez. Metasezgisel algoritmalar, basit ve kolay uygulanması nedeniyle esnekler. Algoritmalar belirli bir probleme göre kolayca değiştirilebilmektedir. Metasezgisel algoritmaların temel özelliği, algoritmaların erken yakınsamasını engelleme konusunda dikkate değer bir yeteneğe sahip olmalarıdır. Algoritmaların stokastik davranışı nedeniyle, teknikler bir kara kutu gibi çalışmakta ve yerel optimumlardan kaçınmaktadır. Ayrıca arama alanını verimli ve etkili bir şekilde keşfetmektedirler.

Metasezgisel algoritmalar, optimizasyon sürecinin iki önemli aşaması olan keşif ve sömürü arasında bir denge kurmaktadır. Keşif aşamasında, algoritmalar arama alanını kapsamlı bir şekilde araştırır ve keşif aşamasında bulunan alanlar yerel olarak aranması için sömürü aşamasına gelmektedir. Bu algoritmalar, çeşitli mühendislik ve bilim problemlerine başarıyla uygulanmaktadır. Elektrik mühendisliğinde enerji üretimi için en uygun çözümün bulunması, endüstriyel alanlarda işlerin programlanması, ulaşım, araç rotalama sorunu, tesis konumu sorununun çözülmesi, inşaat mühendisliğinde köprülerin, binaların tasarlanması, iletişimde radar tasarımı, ağ oluşturma ve veri madenciliğinde sınıflandırma, tahmin, kümeleme, sistem modelleme gibi birçok farklı uygulama alanlarında metasezgisel algoritmalar aktif olarak kullanılmaktadır. Metasezgisel yöntemler genellikle tek çözümlü ve çoklu çözüme dayalı problemler olmak üzere iki sınıfa ayrılmaktadır. Tek çözüme dayalı algoritmalarda arama süreci tek bir aday çözümle başlarken, çoklu çözüme dayalı algoritmalarda optimizasyon bir dizi çözüm (yani popülasyon) kullanılarak gerçekleştirilir. Çoklu çözüme veya popülasyona dayalı meta-sezgiseller, tek çözüme dayalı meta-sezgisellere göre avantajlıdır. Avantajları aşağıdaki gibidir [80]:

- Arama süreci rastgele üretilen popülasyonla, yani çoklu çözümlerden oluşan bir grupla başlamaktadır.

- Arama uzayı etrafında birden fazla çözüm arasında bilgi paylaşılabilir ve yerel optimal çözümlerin önüne geçilebilir.

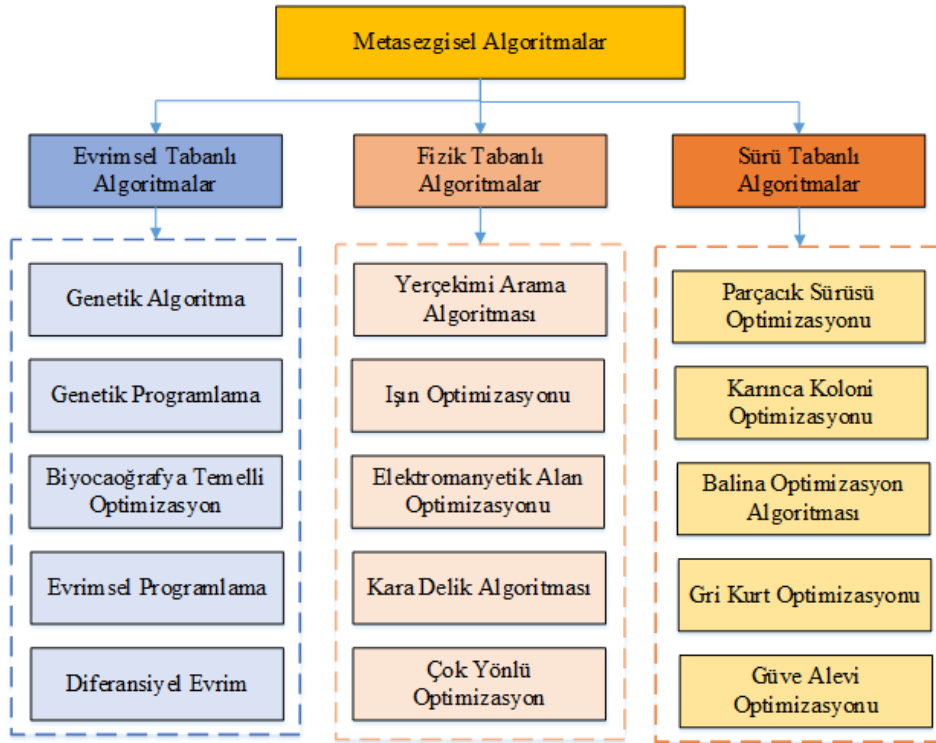
- Çoklu çözüme veya popülasyona dayalı metasezgiseller, tek çözüme dayalı tekniklere göre daha iyi keşfetme yeteneğine sahiptir.

Şekil 3.13'te gösterildiği gibi çoklu çözüme dayalı metasezgisel algoritmalar evrimsel tabanlı, fizik tabanlı ve sürü tabanlı yöntemler olmak üzere üç sınıfa ayrılmaktadır.

Evrimsel tabanlı metasezgisel algoritmalar rekombinasyon, mutasyon ve son olarak seçim olan biyolojik evrimden köken alan popülasyona dayalıdır. İyi bilinen optimizasyon yöntemlerinin bir kategorisi olan Genetik Algoritma (GA), Darwinci evrim teorisini kullanmaktadır. Biyocoğrafya Tabanlı Optimizasyon algoritması, biyocoğrafya kavramına dayalı yeni bir tür optimizasyon tekniğidir. Diferansiyel Evrim sayısal optimizasyon için en popüler ve etkili evrimsel algoritmalarından biridir ve gerçek uygulamaların yanı sıra bir dizi kıyaslama fonksiyonunda da büyük başarı kazanmıştır [81-83].

Fizik tabanlı algoritmalar, yerçekimi kuvveti, eylemsizlik kuvveti ve elektromanyetik kuvvet gibi fizik kanunlarından esinlenerek geliştirilmiştir. Örneğin; Işın Optimizasyonu Snell'in ışık kırılma yasasından, Elektromanyetik Alan Optimizasyonu ve Yerçekimi Arama Algoritması yerçekimi kanunu ve kütle etkileşimlerinden esinlenerek geliştirilmiştir [84-86].

Sürü tabanlı algoritmalar kuşlar, karıncalar ve kurtlar gibi popülasyon organizmalarının sosyal davranışlarından esinlenerek önerilmiştir. Örneğin; Parçacık Sürüsü Optimizasyonu kuşların yiyecek arama davranışını taklit ederken, Karınca Koloni Optimizasyonu karıncaların yiyecek arama davranışını taklit etmektedir. Balina Optimizasyon Algoritması ise kambur balinaların avlanma davranışlarından esinlenmiştir. Benzer şekilde Gri Kurt Optimizasyonu kurtların avlanma ve liderlik davranışlarından ilham almıştır [87-89].



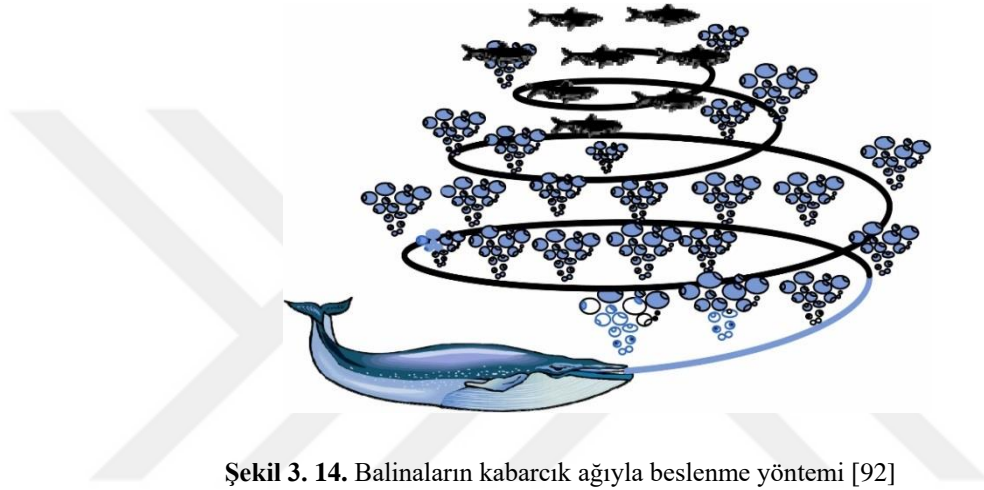
Şekil 3. 13. Metasezgisel Algoritmalar

Bu çalışmada derin öğrenme yöntemleri ile özellikleri çıkarılan fundus görüntülerinden, optimum veya optimuma yakın özellikleri belirlemek için metasezgisel algoritmalarla son yıllarda önerilen Balina Optimizasyon Algoritması ve literatürde iyi bilinen Parçacık Sürüsü Optimizasyon Algoritması tercih edilmiştir.

### 3.7.1. Balina Optimizasyon Algoritması

Balinalar dünyanın en büyük memelileri olarak bilinmektedirler. Yetişkin bir balın boyu 30 metre uzunluğuna ve ağırlığı ise 180 tona ulaşabilmektedir. Balinaların katil, minke, kuzey, kambur, buzul, oluklu ve mavi balina olmak üzere yedi farklı türü bulunmaktadır. Balinalar okyanus yüzeylerinden nefes almak zorunda oldukları için hiçbir zaman tam olarak uykuya dalmazlar. Sadece beyinlerinin yarısı uykuya dalar. Balinalar ile ilgili diğer önemli bir gerçek ise duygularının olmasıdır [90]. Balinaların beyinlerindeki alanlar insanlarınkine benzemektedir. Bu alanlara iş hücreleri denir. İnsanlardaki yargılama, duygu ve sosyal davranışlardan sorumludurlar. Yani iş hücreleri bizi diğer canlılardan ayırır. Balinaların akıllı olmasının ana nedeni bu hücrelerin sayısının yetişkin bir insanın sayısından fazla olmasıdır. Balinaların insanlar gibi düşünebildiği, öğrenebildiği, yargılayabildiği, etkileşime girebildiği ve hatta duygusallaşabildiği, ancak çok daha düşük bir zekâ seviyesine sahip olduğu bir gerçektir [91].

Bir diğ er ilginç nokta ise kambur balinaların sosyal davranışlarıdır. Tek başlarına veya gruplar halinde yaşamaktadırlar. Ancak çoğunlukla gruplar halinde görülmektedirler. Bazı türler, örneğin katil balinalar tüm yaşamları boyunca bir aile içinde yaşayabilirler. Yetişkin bir kambur balina neredeyse bir okul otobüsü büyüklüğündedir. Kambur balinaların ilginç bir başka özelliği de özel avlanma yöntemleridir. Bu yiyecek arama yöntemlerine kabarcık ağıyla beslenme yöntemi denilmektedir. Balinalar, kril sürüsünü veya yüze ye yakın küçük balıkları avlamayı tercih ederler [90,91]. Balinaların bu avlanma yöntemi kabarcık ağıyla beslenme yöntemi olup Şekil 3.14'te gösterilmiştir [92].



Şekil 3. 14. Balinaların kabarcık ağıyla beslenme yöntemi [92]

Kambur balinalar, özel manevralar sayesinde yaklaşık 10-15 m derinliğ e dalabilir ve daha sonra avın etrafında spiral şeklinde kabarcıklar oluşturmaya başlayarak yüze ye doğru yüzerler. Bu davranış balinanın avından gizlenmesini ve avın şaşırtılarak yakalanmasına olanak sağlamaktadır [93]. Kambur balinalar suyun altında soluk vererek hava kabarcığı bulutları ve sütunları meydana getirirler. Bu geniş hava kabarcıkları kümeler halinde avların bir araya toplanmasına olanak sağlar. Daha sonra balina su kabarcıkları içerisinde yüze ye doğru ilerleyerek yükselmeye başlar. Su yüzeyine doğru yükselirken soluk alıp vererek kabarcıklar çıkarmaya devam eder. Balina doğru yaklaştıkça kabarcık çemberini küçülterek hedef avın alanını daraltır [94]. Balina optimizasyon algoritması avın çevresini sarmak, ava doğru ilerlemek ve av aramak olmak üzere üç aşamadan oluşmaktadır [90-92].

**Avın Etrafını Sarmak:** Kambur balinalar avlarının yerini tanıyabilir ve avlarının etrafını sarabilirler. Arama uzayındaki optimum tasarımın konumu önceden bilinmediğ inden, balina optimizasyon algoritması mevcut en iyi aday çözü mün hedef av olduğ unu veya optimuma yakın olduğ unu varsayar. En iyi arama aracı sı tanımlandıktan sonra diğ er arama araçları konumlarını en

iyi arama aracısına göre güncellemeye çalışmaktadır. Bu davranışın matematiksel modeli Denklem 3.3 ve 3.4'te gösterilmiştir [90,91].

$$\vec{D} = |\vec{C} \cdot \vec{X}^*(t) - \vec{X}(t)| \quad (3.3)$$

$$\vec{X}(t+1) = \vec{X}^* - \vec{A} \cdot \vec{D} \quad (3.4)$$

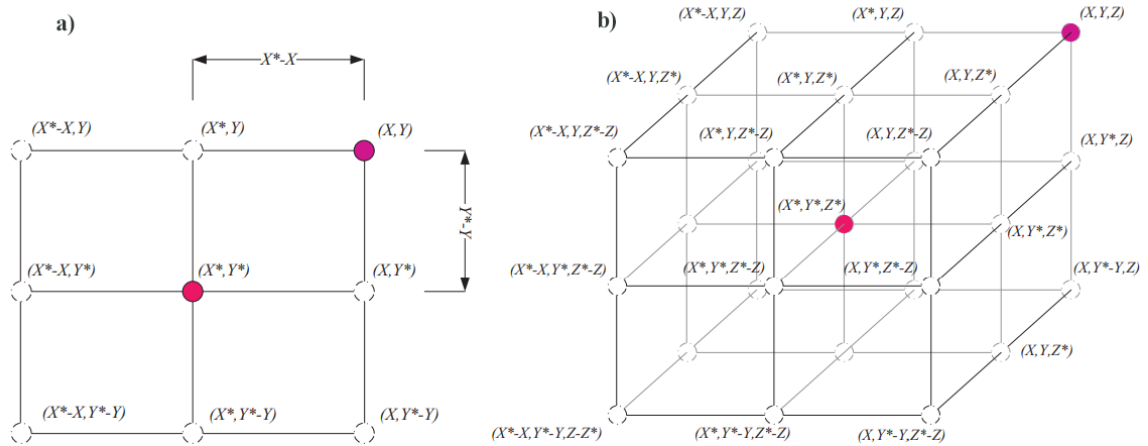
Burada  $t$  mevcut iterasyon sayısını ifade etmektedir.  $X^*$  şu ana kadar elde edilen en iyi konumun çözüm vektörüdür.  $\vec{X}$  Konum vektörüdür.  $\vec{A}$  ve  $\vec{C}$  katsayı vektörlerini göstermektedir. Katsayı vektörlerinin hesaplanması Denklem 3.5 ve Denklem 3.6'da gösterilmiştir. [90,91]

$$\vec{A} = 2\vec{a} \cdot \vec{r} - \vec{a} \quad (3.5)$$

$$\vec{C} = 2 \cdot \vec{r} \quad (3.6)$$

Denklem 3.5'teki  $a$  iterasyonlar boyunca 2'den 0'a doğru doğrusal olarak azalan vektörü göstermektedir. Denklem 3.6'daki  $\vec{r}$  ise [0,1] arasındaki rastgele bir vektörü temsil etmektedir.

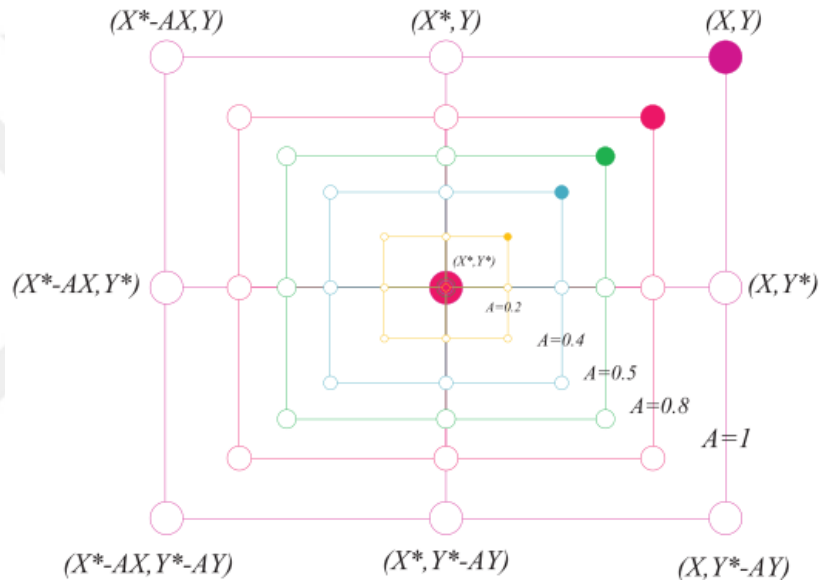
Şekil 3.15 a)'da bir arama ajanının 2 boyutlu vektörlerini ve sonraki olası konumlarını göstermektedir. Burada herhangi bir arama ajanı  $(X,Y)$  pozisyonunu en iyi arama ajanına  $(X^*,Y^*)$  göre saptayabilir. Mevcut duruma göre, A ve C vektörlerinin değeri ayarlanarak mevcut konuma göre en iyi ajanın etrafında çeşitli pozisyonlar elde edilebilir. Şekil 3.15 b)'de ise 3 boyutlu bir arama ajanının boyut vektörlerini ve sonraki olası konumlarını göstermektedir. Burada aynı kavramı n boyutlu bir uzayına genişletmek mümkündür. Bu durumda arama ajanları şimdiye kadar ulaşılan hiper küplerdeki en iyi çözüm etrafında hareket edebilirler [90,91].



Şekil 3. 15. Arama ajanının iki ve üç boyutlu vektörleri ve sonraki olası konumları [90]

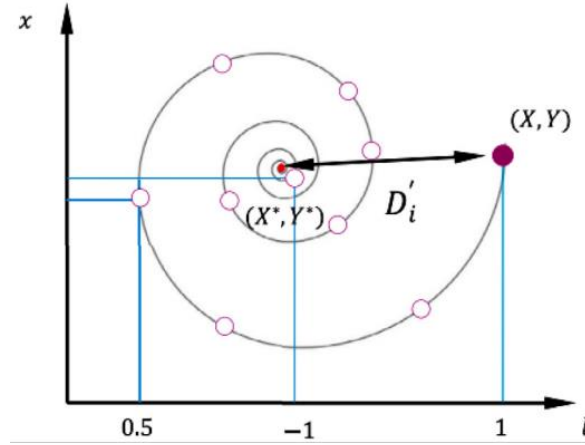
Kambur balinaların baloncuk ağı saldırısı küçülen çevreleme mekanizması ve spiral konum güncellemesi olmak üzere iki aşamadan oluşmaktadır.

**Küçülen çevreleme mekanizması:** Bu davranış, Denklem 3.5'teki  $a$  değerinin düşürülmesiyle elde edilir. Bir başka deyişle  $A$ , yinelemeler boyunca  $a$ 'nın 2'den 0'a düşürüldüğü  $[-a,a]$  aralığındaki rastgele bir değer aralığını temsil etmektedir.  $A$  rastgele  $[-1,1]$  aralığında olduğunda, bir arama aracısının yeni konumu, aracının orijinal konumu ile mevcut en iyi aracının konumu arasında bir yerde tanımlanabilir. Şekil 3.16'da küçülen çevreleme mekanizması gösterilmiştir. 2 boyutlu bir uzayda  $0 \leq A \leq 1$  aralığında  $(X,Y)$  konumunda  $(X^*,Y^*)$  konuma doğru elde edilebilecek konumlar Şekil 3.16'da gösterilmiştir [90,91].



Şekil 3. 16. Küçülen çevreleme mekanizması [91]

**Spiral Konum Güncellemesi:** Spiral güncelleme konumu Şekil 3.17'de gösterildiği gibi  $(X,Y)$ 'de bulunan balina ile  $(X^*, Y^*)$ 'de bulunan av arasındaki mesafeyi hesaplamayı sağlar.



Şekil 3. 17. Spiral konum güncellemesi [91]

Daha sonra balina ile avın arasındaki mesafe matematiksel olarak Denklem 3.7'ye göre hesaplanır.

$$\vec{X}(t+1) = \vec{D}^l \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) \quad (3.7)$$

Burada  $\vec{D}^l = \vec{X}^*(t) - \vec{X}(t)$  arama ajanı ile bilenen en iyi nokta arasındaki mesafeyi vermektedir. Denklem 3.7'de  $b$  logaritmik bir spiral sabiti,  $l$  is  $[-1,1]$  arasında rasgele bir sayıdır. Kambur balinalar avlarının üzerinde spiral bir şekilde yüzerken aynı zamanda azalan bir daire içinde yüzerler. Bu eşzamanlı davranışı göstermek amacıyla optimizasyon sırasında balinaların konumunu güncellemek için küçülen çevreleme mekanizmasını veya spiral modeli kullanma şansının oranı %50 olarak belirlenmektedir. Matematiksel modeli Denklem 3.8'de gösterilmiştir [90,91].

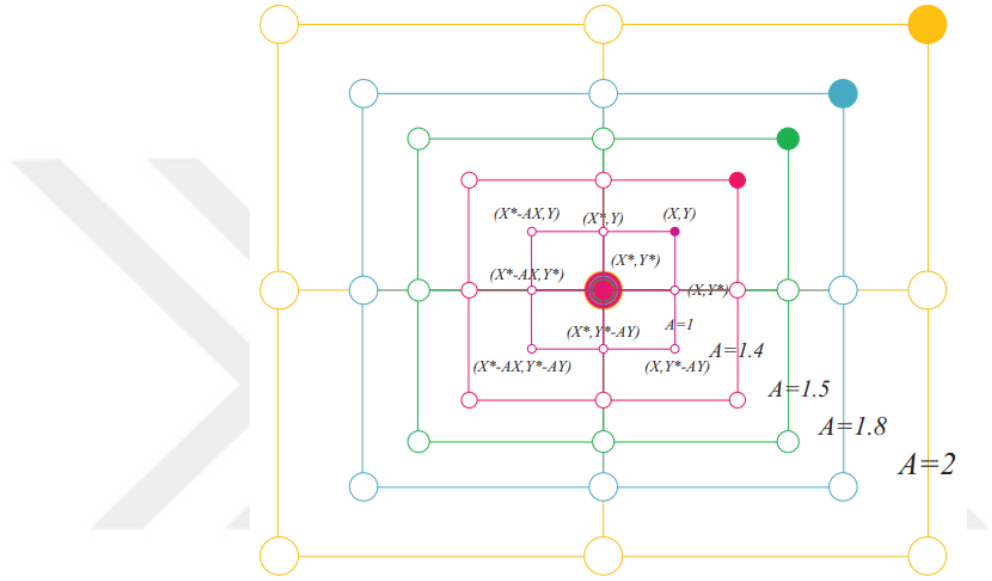
$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D} & \text{if } p < 0,5 \\ \vec{D}^l \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) & \text{if } p \geq 0,5 \end{cases} \quad (3.8)$$

**Av Aramak:** Kambur balinalar arama hareketlerinin sürüsündeki diğer balinaların konumlarına rastgele yapabilmektedir. Global çözümü elde etmek için, çözüm adaylarının yeni konumlarını belirlemek için en iyi nokta yerine, rastgele seçilen bir çözüm adayı belirlenir. Bu çözüm adayının etrafında ise oluşturulacak diğer çözüm adayları oluşur. Bu nedenle,  $A$  vektörünün 1'den büyük ya da -1'den küçük olacağı değerler kullanılmaktadır. Bu davranışın modellenmesi Denklem 3.9 ve 3.10'da gösterilmiştir. Denklemlerdeki  $X_{rand}$  vektörü, rastgele seçilen bir çözüm vektörünü temsil etmektedir.  $A$  değerinin -1'den küçük ya da 1'den büyük olduğu durumlar global arama olarak kabul edilir ve söz edilen denklemler kullanılır. Diğer durumlarda ise yerel arama gerçekleştirilir.

$A > 1$  olduğunda belirli bir çözüm uzayındaki arama mekanizmasına ait olası konumlar Şekil 3.18’de gösterilmiştir. [90-96].

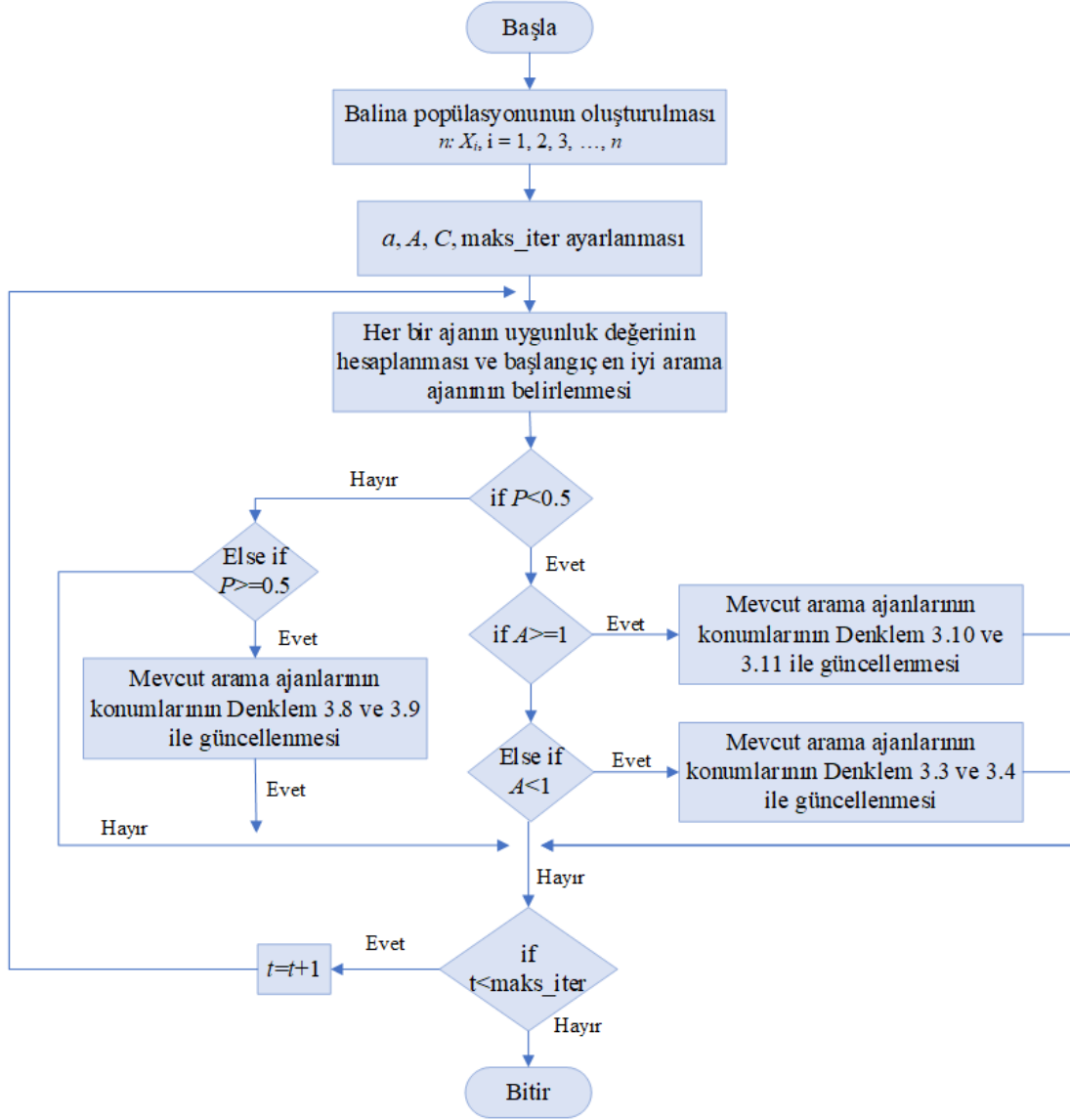
$$\vec{D} = \vec{C} \cdot \overrightarrow{X_{rand}} - \vec{X} \quad (3.9)$$

$$\overrightarrow{X(t+1)} = \overrightarrow{X_{rand}} - \vec{A} \cdot \vec{D} \quad (3.10)$$



Şekil 3. 18. Arama mekanizması [90]

BOA'nın akış diyagramı Şekil 3.19'da verilmiştir.



Şekil 3. 19. BOA'nun akış diyagramı

### 3.7.2. Parçacık Sürü Optimizasyon Algoritması

Parçacık Sürü Optimizasyonu (PSO), 1995 yılında James Kennedy ve Russell Eberhart tarafından geliştirilen popülasyon tabanlı bir meta sezgisel optimizasyon yöntemidir. PSO, küresel olarak kullanılan sürü zekâsı tabanlı optimizasyon algoritmalarından biridir. PSO, kuş ve balık sürülerinin yiyecek kaynağı ararken sürü davranışından esinlenerek geliştirilmiştir [97]. PSO, arama uzayından rastgele parçacıklar üretir. Parçacıklar kişisel ve sosyal deneyimlerini kullanarak hızlarını değiştirerek hareketlerini dinamik olarak kontrol ederler. Bir parçacığın hareketi, onun en iyi konumundan ve diğer parçacıklar tarafından oluşturulan daha iyi konumlardan etkilenir; bu da uzaydaki çözümü optimize eder. En iyi sonuç, tüm parçacıkların arama uzayının aynı konumunda bulunmasıdır. Bu pozisyon en iyi çözümdür. Parçacıklar arasındaki bilgi alışverişi, her parçacığın

konumunu ve hızını yinelemeli olarak etkiler. Kişisel deneyime dayanarak her parçacık şu ana kadar elde edilen en iyi çözüme sahiptir. Sürünün sosyal deneyiminden küresel en iyi çözüm bulunur. Kişisel en iyi ve küresel en iyinin etkisi, rastgele bir düzeltme faktörü kullanılarak dengelenir. Genel olarak parçacıkların hızı ve konumu Denklem 3.12 ve Denklem 3.13'te gösterilmiştir [98,99].

$$V_{i,j}^{k+1} = (W \times V_{i,j}^k) + c_1 (rand1 \times (Pbest_{i,j} - X_{i,j}^k)) + c_2 (rand2 \times (Gbest_{i,j} - X_{i,j}^k)) \quad (3.12)$$

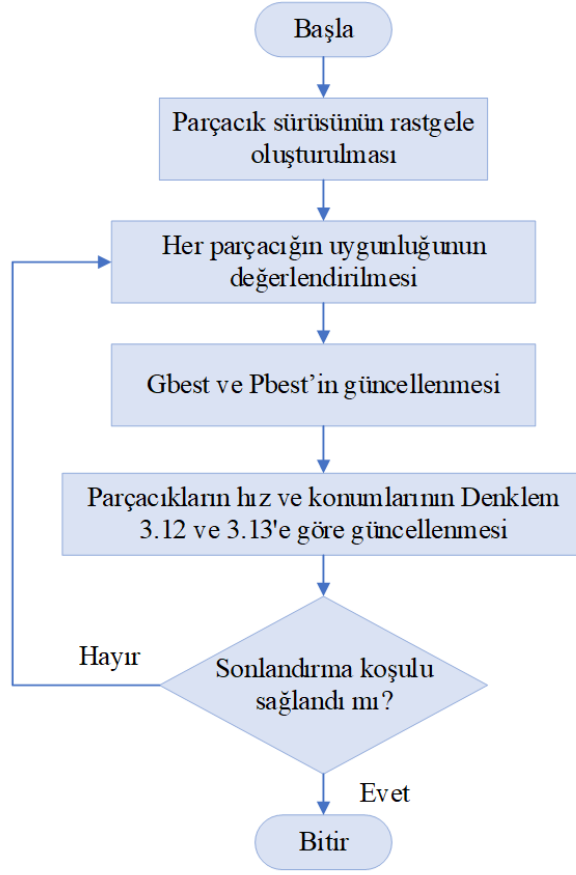
$$X_{i,j}^{k+1} = X_{i,j}^k + V_{i,j}^{k+1} \quad (3.13)$$

Burada  $X_i$   $i$ 'ninci parçacığın mevcut konumunu temsil eder.  $V_i$   $i$ 'ninci parçacığın birim zamandaki hızıdır. Parçacığın bireysel yani yerel en iyi konumunu  $P_{best}$ , küresel en iyi değerini ise  $G_{best}$  temsil etmektedir. Ayrıca  $i$ 'ninci parçacığın  $k$  yinelemesindeki hız ve konumu Denklem 3.14 ve Denklem 3.15'de gösterilmiştir [98,99].

$$V_{i,j} = (V_{i,1'}, V_{i,2'}, \dots, V_{i,j'}) \quad (3.14)$$

$$X_{i,j} = (X_{i,1'}, X_{i,2'}, \dots, X_{i,j'}) \quad (3.15)$$

$W$  atalet ağırlık değeri,  $c_1$  ve  $c_2$  üst çözümlerin parçacığın hızı üzerindeki ivme katsayılarıdır. PSO'nun akış diyagramı Şekil 3.20'de verilmiştir.

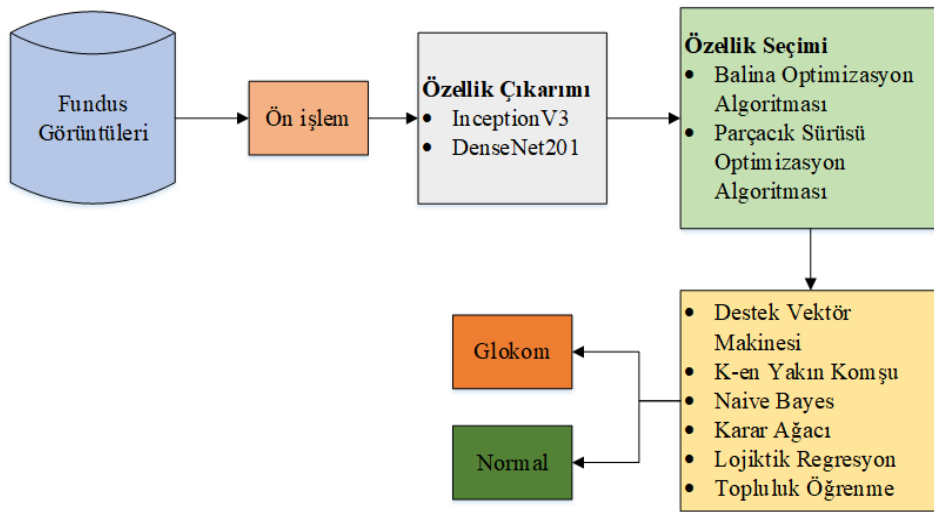


Şekil 3. 20. PSO'nun akış diyagramı

## 4. MATERYAL VE METOD

Bu çalışmada, fundus görüntülerinden özellik haritasını çıkarmak için önceden eğitilmiş transfer öğrenme modeli DenseNet201 ve Inceptionv3 mimarileri kullanılmıştır. Ön işlem aşamasında veri kümesindeki görüntüler derin öğrenme mimarilerinin giriş görüntüsüne göre yeniden boyutlandırılmıştır. DenseNet201 mimarisinde orijinal boyutta alınan görüntüler, Inceptionv3 mimarisi için 299x299x3 boyutunda kullanılmıştır. Derin öğrenme mimarilerinin tam bağlantı katmanları görüntüden özellikleri çıkarmak için kullanılmıştır. DenseNet201 ağı üzerindeki tamamen bağlı fc1000 katmanı ile inceptionv3 ağı üzerindeki predictions katmanları veri kümesinden özellikler çıkarmak için kullanılır. Her iki mimari ile 705 görüntüden oluşan veri kümesi için son vektör  $705 \times 1000$ 'dir. Bu satır bazında özellikler, sınıflandırma için matris oluşturularak kullanılmıştır.

Çıkarılan özellik haritasından ilgisiz özellikleri elemek ve ideal verilere ulaşmak için meta sezgisel optimizasyon algoritmalarından BOA ve PSO kullanılmıştır. BOA ve PSO algoritmaları farklı popülasyon büyüklüklerine göre optimal özelliklerin seçimi için kullanılmıştır. Daha sonra, orijinal veri kümesine SVM, KNN, Naive Bayes, Logistic Regression, Karar Ağacı ve Topluluk Öğrenme sınıflandırıcı algoritmaları uygulanmıştır. BOA ve PSO ile özellik seçimi yapıldıktan sonra elde edilen alt veri kümelerine SVM, KNN, Naive Bayes, Logistic Regression, Karar Ağacı ve Topluluk Öğrenme sınıflandırıcı algoritmaları uygulanmıştır. Modellerin doğruluğunu arttırmak ve aşırı öğrenmenin önüne geçmek için, sınıflandırma algoritmaları beş kat çapraz doğrulama ile uygulanmıştır. Önerilen yöntemin şematik gösterimi Şekil 4.1'de gösterilmiştir.



Şekil 4. 1. Önerilen yöntemin şematik gösterimi

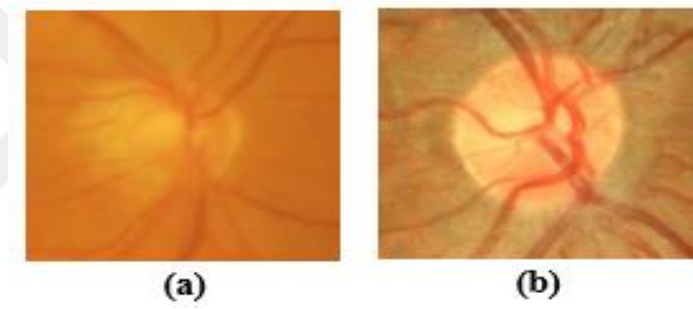
#### 4.1. Veri Kümesinin Hazırlanması

Bu çalışmada, genel erişime açık fundus görüntülerinden oluşan Acrima veri kümesi kullanılmıştır. Veri kümesinde 396 adet glokom hastasına ve 309 adet sağlıklı (normal) bireye ait olmak üzere toplam 705 görüntü bulunmaktadır. Acrima veri kümesinden sağlıklı ve hasta bireylere ait görüntülerin dağılım sayısı Tablo 4.1.'de gösterilmiştir.

**Tablo 4. 1.** Veri kümesinden alınan görüntülerin dağılım sayısı

Veri Kümesi	Glokom görüntü	Normal Görüntü	Toplam
Acrima veri kümesi	396	309	705

Veri kümesinden alınan görüntülerden glokom ve normal hastaların örnek görüntüsü Şekil 4.2'de gösterilmiştir.



**Şekil 4. 2.** Glokom görüntüsü (a), normal görüntüsü (b)

#### 4.2. Değerlendirme Kriterleri

Sınıflandırma algoritmalarının başarısını ölçmek için karmaşıklık matrisi kullanılmaktadır. Karmaşıklık matrisi ile gerçek ve tahmin edilen örneklerin sayısını gösteren matris elde edilmektedir. Tablo 4.2'de karmaşıklık matrisi gösterilmiştir.

**Tablo 4. 2.** Karmaşıklık matrisi

		Tahmin Edilen Sınıf	
		Pozitif	Negatif
Gerçek Sınıf	Pozitif	Doğru Pozitif (DP)	Yanlış Pozitif (YP)
	Negatif	Yanlış Negatif (YN)	Doğru Negatif (DN)

Karmaşıklık matrisinde bulunan terimler anlamları aşağıda gösterilmiştir. Doğru Pozitif (DP): Pozitif sınıfa ait olan ve sınıflandırıcı tarafından doğru şekilde sınıflandırılan verileri temsil

etmektedir. Doğru Negatif (DN): Sınıflandırıcı tarafından doğru olarak sınıflandırılan negatif sınıfa ait verileri temsil etmektedir. Yanlış Pozitif (YP): Negatif sınıfa ait ifadeler yanlış sınıflandırılır ve pozitif sınıf olarak tanımlanır. Yanlış Negatif (YN): Gerçekte pozitif sınıfa ait olan bir örnek negatif sınıf olarak tanımlanır [100,101].

**Doğruluk:** Modelin performansının değerlendirilmesi için kullanılmaktadır. Modeldeki doğru olarak elde edilen tahminlerin sayısının modeldeki tüm veri kümesine bölünmesiyle hesaplanmaktadır. Doğruluk değeri denklem 4.1 ile hesaplanmaktadır.

$$\text{Doğruluk} = \frac{(DP + DN)}{(DP + DN + YP + YN)} \quad (4.1)$$

**Duyarlılık:** Pozitif örneklerin doğruluğuna karşılık gelir ve pozitif sınıfların kaç örneğinin doğru şekilde etiketlendiğini ifade etmektedir. Duyarlılık değeri Denklem 4.2 ile hesaplanmaktadır.

$$\text{Duyarlılık} = \frac{DP}{(DP + YN)} \quad (4.2)$$

**Özgüllük:** Modelin, örneklerinin negatif sınıf etiketini tahmin etmedeki olasılığını ifade etmektedir. Denklem 4.3 ile hesaplanmaktadır.

$$\text{Özgüllük} = \frac{DN}{(DN + YP)} \quad (4.3)$$

**F1-ölçütü:** Pozitif sınıfın analizine odaklanan bir metriktir. Bu metriğin değerinin yüksek olması, modelin pozitif sınıfta daha iyi performans gösterdiğini gösterir. Denklem 4.4 ile hesaplanmaktadır.

$$F1 - \text{ölçütü} = \frac{2 * DP}{(2 * DP + YP + YN)} \quad (4.4)$$

## 5. BULGULAR VE TARTIŞMA

Bu bölümde ACRIMA veri kümesine ön işlem aşamasından geçirildikten InceptionV3 ve DenseNet201 yöntemleri uygulanmıştır. Her bir yöntem sonrasında elde edilen özellik kümelerinden faydalı özellikleri elde edebilmek için son yıllarda önerilen Balina Optimizasyon ve literatürde iyi biline Parçacık Sürüsü Optimizasyon Algoritmaları tercih edilmiştir. Bu metasezgisel algoritmalar ile özellik seçimi gerçekleştirilirken popülasyon büyüklüğü ve iterasyon sayılarının farklı değerleri göz önüne alınarak özellik sayısı, optimum değer ve çalışma zamanına ait veriler elde edilerek değerlendirmeler yapılmıştır. Popülasyon büyüklüğü ve iterasyon sayısı değerlerinin seçilen özellik sayısı üzerindeki etkisi incelenmiştir. Özellik seçimi sonrasında elde edilen yeni veri kümelerine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme makine öğrenmesi algoritmaları uygulanmıştır. Bu algoritmalar sonucunda elde edilen sonuçlar doğruluk, duyarlılık, özgüllük ve f1-ölçütü bakımından değerlendirilmiştir. Ayrıca, makine öğrenmesi algoritmaları InceptionV3 ve DenseNet201 ile özellik çıkarımı yapılan orijinal ACRIMA veri kümesine de uygulanmıştır. Böylelikle metasezgisel algoritmalar ile özellik seçimi yapılmasının fundus görüntülerinden glokom hastalığının tespitinin başarısı üzerindeki etkisi değerlendirilmiştir.

### 5.1. Balina Optimizasyon Algoritması Sonuçları

ACRIMA veri kümesi ön işlem aşamasından geçirildikten sonra InceptionV3 mimarisi uygulanmıştır. Bu mimari ile çıkarılan toplam özellik sayısı 1000'dir. 1000 özellikten oluşan bu veri kümesinden faydalı özellikleri çıkarabilmek için BOA farklı popülasyon büyüklüklerinde (5, 10, 20 ve 30) ve farklı iterasyon sayılarında (100, 200, 300, 400 ve 500) uygulanmıştır. 1000 özellikli veri kümesine BOA uygulanması sonucu elde edilen özellik sayıları, optimum değer (uygunluk fonksiyonu değeri) ve BOA'nın çalışma zamanı Tablo 5.1'de gösterilmiştir.

Tablo 5.1'deki değerler incelendiğinde popülasyon büyüklüğü 5 olarak belirlendiğinde en iyi uygunluk değerleri iterasyon sayısı 200 iken 0.1205 olarak elde edilmiştir. Özellik sayısı ise 412 olarak elde edilmiştir. Tablodaki diğer değerler incelendiğinde, popülasyon sayısı 10 ve maksimum iterasyon sayısı 400 olarak ayarlandığında en iyi uygunluk değeri 0.0851 ve seçilen özellik sayısı 525, popülasyon sayısı 20 ve iterasyon sayısı 500 olarak ayarlandığında en iyi uygunluk değeri 0.1205 ve seçilen özellik sayısı 244 ve popülasyon sayısı 30 ve maksimum iterasyon sayısı 400 olarak ayarlandığında en iyi uygunluk değeri 0.0992 ve seçilen özellik sayısı 148 olarak elde edilmiştir. Elde edilen en iyi değerler koyu renk ile gösterilmiştir.

**Tablo 5. 1.** InceptionV3 sonrası BOA'nın farklı popülasyonlarda özellik seçim sonuçları

Özellik Sayısı	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	100	5	404	0.1276	7.95
	<b>200</b>	<b>5</b>	<b>412</b>	<b>0.1205</b>	<b>17.63</b>
	300	5	369	0.1418	20.68
	400	5	849	0.1347	45.53
	500	5	464	0.1418	41.32
	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	100	10	454	0.1276	16.08
	200	10	634	0.1134	38.16
	300	10	55	0.1773	25.25
	<b>400</b>	<b>10</b>	<b>525</b>	<b>0.0851</b>	<b>69.70</b>
	500	10	457	0.1560	80.10
	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
1000	100	20	93	0.1489	14.86
	200	20	80	0.1276	28.03
	300	20	512	0.1631	99.53
	400	20	592	0.1276	150.19
	<b>500</b>	<b>20</b>	<b>244</b>	<b>0.1205</b>	<b>99.48</b>
	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	100	30	90	0.1205	23.61
	200	30	613	0.1063	114.10
	300	30	424	0.1276	131.71
	<b>400</b>	<b>30</b>	<b>148</b>	<b>0.0992</b>	<b>109.74</b>
	500	30	310	0.1134	167.40

ACRIMA veri kümesi ön işleminden geçirildikten sonra DenseNet201 mimarisi ile çıkarılan toplam 1000 özellikli veri kümesine BOA (5, 10, 20 ve 30) ve farklı iterasyon sayılarında (100, 200, 300, 400 ve 500) uygulandığında elde edilen özellik sayısı, optimum değer ve çalışma zamanı süreleri Tablo 5.2'de verilmiştir.

Tablo 5.2'deki değerler incelendiğinde popülasyon büyüklüğü 5 olarak belirlendiğinde en iyi uygunluk değerleri iterasyon sayısı 300 iken 0.0212 olarak elde edilmiştir. Özellik sayısı ise 146 olarak elde edilmiştir. Tablodaki diğer değerler incelendiğinde, popülasyon sayısı 10 ve maksimum iterasyon sayısı 400 olarak ayarlandığında en iyi uygunluk değeri 0.0283 ve seçilen özellik sayısı 483, popülasyon sayısı 20 ve iterasyon sayısı 200 olarak ayarlandığında en iyi uygunluk değeri 0.0297 ve seçilen özellik sayısı 97, popülasyon sayısı 30 ve maksimum iterasyon sayısı 300 olarak ayarlandığında en iyi uygunluk değeri 0.0283 ve seçilen özellik sayısı 88 olarak elde edilmiştir. Elde edilen en iyi değerler koyu renk ile gösterilmiştir.

**Tablo 5. 2.** DenseNet201 sonrası BOA'nın farklı popülasyonlarda özellik seçim sonuçları

Özellik Sayısı	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	100	5	777	0.0709	3.69
	200	5	89	0.0496	12.02
	<b>300</b>	<b>5</b>	<b>146</b>	<b>0.0212</b>	<b>12.53</b>
	400	5	536	0.0780	92.19
	500	5	97	0.0638	25.37
	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	100	10	269	0.0638	11.95
	200	10	428	0.0567	34.08
	300	10	484	0.0496	48.04
	<b>400</b>	<b>10</b>	<b>483</b>	<b>0.0283</b>	<b>66.10</b>
	500	10	607	0.0780	92.65
	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
1000	100	20	58	0.0638	16.75
	<b>200</b>	<b>20</b>	<b>97</b>	<b>0.0297</b>	<b>31.72</b>
	300	20	348	0.0354	74.31
	400	20	388	0.0638	186.82
	500	20	94	0.0496	75.40
	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	100	30	399	0.0496	43.45
	200	30	619	0.0567	119.20
	<b>300</b>	<b>30</b>	<b>88</b>	<b>0.0283</b>	<b>87.25</b>
	400	30	686	0.0709	250.25
	500	30	348	0.0425	220.01

## 5.2. Parçacık Sürü Optimizasyon Algoritması Sonuçları

Fundus görüntülerinden glokom hastalığının tespiti için özellik seçimi yöntemlerinin etkisini değerlendirebilmek için belirlenen diğer optimizasyon algoritması literatürde iyi bilinen PSO'dur. Fundus görüntülerine ön işlemler uygulandıktan sonra Inceptionv3 mimarisi uygulandığında elde edilen toplam 1000 özellikli veri kümesine PSO farklı popülasyon büyüklüklerinde (5, 10, 20 ve 30) ve farklı iterasyon sayılarında (100, 200, 300, 400 ve 500) uygulanmıştır 1000 özellikli veri kümesine BOA uygulanması sonucu elde edilen özellik sayıları, optimum değer (uygunluk fonksiyonu değeri) ve BOA'nın çalışma zamanı Tablo 5.3'te gösterilmiştir.

Tablo 5.3'e göre, popülasyon büyüklüğü 5 olarak belirlendiğinde en iyi uygunluk değerleri iterasyon sayısı 200 iken 0.0747 olarak elde edilmiştir. Özellik sayısı ise 452 olarak elde edilmiştir. Tablodaki diğer değerler incelendiğinde, popülasyon sayısı 10 ve maksimum iterasyon sayısı 300 olarak ayarlandığında en iyi uygunluk değeri 0.0818 ve seçilen özellik sayısı 466, popülasyon sayısı 20 ve iterasyon sayısı 500 olarak ayarlandığında en iyi uygunluk değeri 0.0891 ve seçilen özellik sayısı 492 ve popülasyon sayısı 30 ve maksimum iterasyon sayısı 500 olarak ayarlandığında en iyi uygunluk değeri 0.0534 ve seçilen özellik sayısı 429 olarak elde edilmiştir.

**Tablo 5. 3.** InceptionV3 sonrası PSO'nun farklı popülasyonlarda özellik seçim sonuçları

Özellik Sayısı	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	100	5	469	0.1101	9.90
	<b>200</b>	<b>5</b>	<b>452</b>	<b>0.0747</b>	<b>19.93</b>
	300	5	476	0.1241	30.29
	400	5	466	0.8891	38.87
	500	5	454	0.1379	46.52
	<b>İterasyon Sayısı</b>	<b>Popülasyon Boyutu</b>	<b>Özellik Sayısı</b>	<b>Optimum Değeri</b>	<b>Çalışma Zamanı</b>
	100	10	487	0.1172	19.57
	200	10	478	0.1381	37.55
	<b>300</b>	<b>10</b>	<b>466</b>	<b>0.0818</b>	<b>55.70</b>
	400	10	482	0.1101	74.62
	500	10	432	0.1377	88.60
	<b>İterasyon Sayısı</b>	<b>Popülasyon Boyutu</b>	<b>Özellik Sayısı</b>	<b>Optimum Değeri</b>	<b>Çalışma Zamanı</b>
1000	100	20	483	0.1312	38.90
	200	20	478	0.1030	72.50
	300	20	438	0.1026	107.20
	400	20	497	0.1102	164.20
	<b>500</b>	<b>20</b>	<b>492</b>	<b>0.0891</b>	<b>195.20</b>
	<b>İterasyon Sayısı</b>	<b>Popülasyon Boyutu</b>	<b>Özellik Sayısı</b>	<b>Optimum Değeri</b>	<b>Çalışma Zamanı</b>
	100	30	498	0.1173	61.60
	200	30	468	0.0748	110.02
	300	30	483	0.0820	161.75
	400	30	423	0.1025	192.07
	<b>500</b>	<b>30</b>	<b>429</b>	<b>0.0534</b>	<b>283.12</b>

ACRIMA veri kümesi ön işlemden geçirildikten sonra DenseNet201 mimarisi ile çıkarılan toplam 1000 özellikli veri kümesine PSO (5, 10, 20 ve 30) ve farklı iterasyon sayılarında (100, 200, 300, 400 ve 500) uygulandığında elde edilen özellik sayısı, optimum değer ve çalışma zamanı süreleri Tablo 5.4'te verilmiştir.

Tablo 5.4'teki değerler incelendiğinde popülasyon büyüklüğü 5 olarak belirlendiğinde en iyi uygunluk değerleri iterasyon sayısı 100 iken 0.0463 olarak elde edilmiştir. Özellik sayısı ise 471 olarak elde edilmiştir. Tablodaki diğer değerler incelendiğinde, popülasyon sayısı 10 ve maksimum iterasyon sayısı 400 olarak ayarlandığında en iyi uygunluk değeri 0.0318 olarak elde edilmiştir. Seçilen özellik sayısı 380 olarak elde edilmiştir. Popülasyon sayısı 20 ve iterasyon sayısı 500 olarak ayarlandığında en iyi uygunluk değeri 0.0176 ve seçilen özellik sayısı 361 olarak elde edilmiştir. Popülasyon sayısı 30 ve maksimum iterasyon sayısı 200 olarak ayarlandığında en iyi uygunluk değeri 0.0460 ve seçilen özellik sayısı 388 olarak elde edilmiştir.

**Tablo 5. 4.** DenseNet201 sonrası PSO'nun farklı popülasyonlarda özellik seçim sonuçları

Özellik Sayısı	İterasyon Sayısı	Popülasyon Boyutu	Özellik Sayısı	Optimum Değeri	Çalışma Zamanı
	<b>100</b>	<b>5</b>	<b>471</b>	<b>0.0463</b>	<b>9.62</b>
	200	5	458	0.0747	19.91
	300	5	463	0.0888	39.51
	400	5	454	0.0677	39.93
	500	5	459	0.0467	48.54
	<b>İterasyon Sayısı</b>	<b>Popülasyon Boyutu</b>	<b>Özellik Sayısı</b>	<b>Optimum Değeri</b>	<b>Çalışma Zamanı</b>
	100	10	492	0.0540	20.15
	200	10	442	0.0535	54.74
	300	10	443	0.0535	59.01
	<b>400</b>	<b>10</b>	<b>380</b>	<b>0.0318</b>	<b>71.46</b>
	500	10	406	0.0461	88.36
	<b>İterasyon Sayısı</b>	<b>Popülasyon Boyutu</b>	<b>Özellik Sayısı</b>	<b>Optimum Değeri</b>	<b>Çalışma Zamanı</b>
1000	100	20	436	0.0535	40.67
	200	20	417	0.0673	73.22
	300	20	430	0.0394	103.22
	400	20	402	0.0321	136.85
	<b>500</b>	<b>20</b>	<b>361</b>	<b>0.0176</b>	<b>159.96</b>
	<b>İterasyon Sayısı</b>	<b>Popülasyon Boyutu</b>	<b>Özellik Sayısı</b>	<b>Optimum Değeri</b>	<b>Çalışma Zamanı</b>
	100	30	418	0.0533	51.36
	<b>200</b>	<b>30</b>	<b>388</b>	<b>0.0460</b>	<b>107.87</b>
	300	30	449	0.0466	170.12
	400	30	366	0.0668	197.45
	500	30	380	0.0599	250.01

### 5.3. Orijinal Veri Kümesi İçin Farklı Sınıflandırıcıların Performansları

ACRIMA veri kümesinden Inceptionv3 mimarisi ile toplam 1000 özellik elde edilmiştir. 1000 özellikten oluşan orijinal veri kümesine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme algoritmaları 5-kat çapraz doğrulama ile uygulandığında, doğruluk, duyarlılık, özgüllük ve f1-ölçütü bakımından elde edilen değerler Tablo 5.5'te gösterilmiştir. Bu tablo incelendiğinde, en yüksek doğruluk, duyarlılık ve f1-ölçütü değerleri SVM algoritması ile özgüllük değeri ise Topluluk Öğrenme algoritması ile elde edilmiştir. Bu ölçütlere göre en düşük değerler ise Lojistik Regresyon algoritması ile elde edilmiştir.

**Tablo 5. 5.** Inceptionv3 mimarisi ile çıkan özelliklerin ile farklı sınıflandırıcı performansları

Sınıflandırıcı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	0.6057	0.6657	0.5444	0.6303
Naive Bayes	0.8128	0.8385	0.7810	0.8321
SVM	<b>0.8936</b>	<b>0.8983</b>	0.8874	<b>0.9061</b>
KNN	0.8227	0.8430	0.7968	0.8420
Karar Ağacı	0.7645	0.7700	0.7563	0.7981
Topluluk Öğrenme	0.8440	0.8164	<b>0.8933</b>	0.8730

ACRIMA veri kümesinden DenseNet201 mimarisi ile toplam 1000 özellik elde edilmiştir. 1000 özellikten oluşan orijinal veri kümesine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme algoritmaları 5-kat çapraz doğrulama ile uygulandığında, doğruluk, duyarlılık, özgüllük ve f1-ölçütü bakımından elde edilen değerler Tablo 5.6'da gösterilmiştir. Tabloya göre, en yüksek doğruluk, duyarlılık, özgüllük ve f1-ölçütü değerleri SVM algoritması ile elde edilmiştir. En düşük değerler ise Lojistik Regresyon algoritması ile elde edilmiştir.

**Tablo 5. 6.** Densenet201 mimarisi ile çıkan özelliklerin farklı sınıflandırıcı performansları

Sınıflandırıcı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	0.6355	0.6863	0.5783	0.6658
Naive Bayes	0.9021	0.8865	0.9255	0.9158
SVM	<b>0.9730</b>	<b>0.9677</b>	<b>0.9801</b>	<b>0.9762</b>
KNN	0.9106	0.8758	0.9695	0.9249
Karar Ağacı	0.8284	0.8446	0.8072	0.8478
Topluluk Öğrenme	0.9149	0.8962	0.9431	0.9268

#### 5.4. Özellik Seçimi Sonrası Alt Veri Kümeleri İçin Farklı Sınıflandırıcıların Performansları

Çalışmanın bu bölümünde InceptionV3 ve DenseNet201 mimarileri ile çıkarılan özellik kümelerine BOA ve PSO uygulanması sonucu elde edilen dört alt veri kümesine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme algoritmalarının uygulanması ile elde edilecek sonuçlar karşılaştırmalı olarak verilmiştir. Bu sayede farklı ESA mimarilerinin ve farklı metasezgisel algoritmalarının birlikte kullanımının glokom hastalığının tespiti üzerinde başarıyı incelemiştir.

Tablo 5.1'de BOA uygulanması sonucunda en düşük uygunluk değerinin elde edildiği iterasyonda seçilen alt özellik kümesine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme algoritmaları 5-kat çapraz doğrulama algoritmaları uygulanmıştır. Farklı özellik sayısındaki veri kümelerinden elde edilen doğruluk, duyarlılık, özgüllük ve f1-ölçütü değerleri Tablo 5.7'de verilmiştir. ACRIMA veri kümesinden InceptionV3 mimarisi sonrası BOA uygulanması ile seçilen özellik sayısı en az 148 ve en fazla 525'tir. Farklı özellik sayısına sahip veri kümeleri için en yüksek doğruluk, duyarlılık, özgüllük ve f1-ölçütü değerleri 244 özellik sayısına sahip kümedeki duyarlılık değeri dışında SVM algoritması ile elde edilmiştir. En düşük değerler ise Karar Ağacı ve Lojistik Regresyon algoritmaları ile elde edilmiştir. Orijinal veri kümesindeki ve InceptionV3+BOA ile elde edilen alt veri kümelerindeki sonuçlar kıyaslandığında Lojistik Regresyon algoritmasının başarı oranının arttığı gözlemlenmiştir.

**Tablo 5. 7.** InceptionV3+BOA için farklı sınıflandırıcı performansları

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	412	0.7773	0.8088	0.7390	0.7995
Naive Bayes	412	0.8014	0.8316	0.7649	0.8210
SVM	412	<b>0.8879</b>	<b>0.8783</b>	<b>0.9021</b>	<b>0.9031</b>
KNN	412	0.8151	0.8276	0.7993	0.8379
Karar Ağacı	412	0.7504	0.7619	0.7333	0.7843
Topluluk Öğrenme	412	0.8440	0.8221	0.8812	0.8690

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	525	0.6411	0.6938	0.5833	0.6693
Naive Bayes	525	0.8000	0.8228	0.7710	0.8217
SVM	525	<b>0.8837</b>	<b>0.8685</b>	<b>0.9068</b>	<b>0.9002</b>
KNN	525	0.8199	0.8036	0.8473	0.8486
Karar Ağacı	525	0.7546	0.7767	0.7252	0.7835
Topluluk Öğrenme	525	0.8298	0.8067	0.8706	0.8582

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	244	0.8397	<b>0.8656</b>	0.8082	0.8557
Naive Bayes	244	0.8156	0.8393	0.7859	0.8350
SVM	244	<b>0.8723</b>	0.8592	<b>0.8925</b>	<b>0.8905</b>
KNN	244	0.8326	0.8062	0.8805	0.8612
Karar Ağacı	244	0.7418	0.7488	0.7309	0.7797
Topluluk Öğrenme	244	0.8340	0.8178	0.8609	0.8599

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	148	0.8511	0.8721	0.8248	0.8666
Naive Bayes	148	0.8397	0.8546	0.8203	0.8579
SVM	148	<b>0.8922</b>	<b>0.8883</b>	<b>0.8976</b>	<b>0.9059</b>
KNN	148	0.8511	0.8299	0.8864	0.8746
Karar Ağacı	148	0.7305	0.7384	0.7179	0.7705
Topluluk Öğrenme	148	0.8298	0.8026	0.8795	0.8592

Tablo 5.2’de BOA uygulanması sonucunda en düşük uygunluk değerinin elde edildiği iterasyonda seçilen alt özellik kümesine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme algoritmaları 5-kat çapraz doğrulama algoritmaları uygulanmıştır. Farklı özellik sayısındaki veri kümelerinden elde edilen doğruluk, duyarlılık, özgüllük ve f1-ölçütü değerleri Tablo 5.8’de verilmiştir. ACRIMA veri kümesinden DenseNet201 mimarisi sonrası BOA uygulanması ile seçilen özellik sayısı en az 88 ve en fazla 483’tür. Farklı özellik sayısına sahip veri kümeleri için en yüksek doğruluk, duyarlılık ve f1-ölçütü değerleri SVM algoritması ile en yüksek özgüllük değerleri ise Topluluk Öğrenme algoritması ile elde edilmiştir. En düşük doğruluk değerleri ise Karar Ağacı algoritması ile elde edilmiştir. Orijinal veri kümesindeki ve DenseNet201+BOA ile elde edilen alt veri kümelerindeki sonuçlar kıyaslandığında Lojistik Regresyon ve Topluluk Öğrenme algoritmaların başarımlarının arttığı gözlemlenmiştir.

**Tablo 5. 8.** DenseNet201+BOA için farklı sınıflandırıcı performansları

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	146	0.9433	0.9428	0.9439	0.9499
Naive Bayes	146	0.8993	0.8824	0.9250	0.9135
SVM	146	<b>0.9546</b>	<b>0.9461</b>	0.9663	<b>0.9602</b>
KNN	146	0.9149	0.8853	0.9628	0.9279
Karar Ağacı	146	0.8028	0.8067	0.7972	0.8294
Topluluk Öğrenme	146	0.9532	0.9416	<b>0.9694</b>	0.9591
Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	483	0.7986	0.7702	0.8207	0.8207
Naive Bayes	483	0.9007	0.8976	0.9051	0.9132
SVM	483	<b>0.9699</b>	<b>0.9676</b>	0.9731	<b>0.9737</b>
KNN	483	0.9106	0.8863	0.9489	0.9238
Karar Ağacı	483	0.7887	0.8080	0.7632	0.8130
Topluluk Öğrenme	483	0.9688	0.9629	<b>0.9767</b>	0.9725
Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	97	0.9376	0.9467	0.9260	0.9443
Naive Bayes	97	0.9121	0.8995	0.9303	0.9238
SVM	97	<b>0.9589</b>	<b>0.9531</b>	0.9667	<b>0.9638</b>
KNN	97	0.9206	0.8935	0.9634	0.9324
Karar Ağacı	97	0.8468	0.8495	0.8430	0.8663
Topluluk Öğrenme	97	0.9532	0.9373	<b>0.9759</b>	0.9593
Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	88	0.9418	0.9494	0.9323	0.9482
Naive Bayes	88	0.8908	0.8807	0.9056	0.9055
SVM	88	<b>0.9532</b>	<b>0.9526</b>	0.9539	<b>0.9586</b>
KNN	88	0.8979	0.8699	0.9438	0.9137
Karar Ağacı	88	0.8071	0.8186	0.7912	0.8308
Topluluk Öğrenme	88	0.9461	0.9345	<b>0.9625</b>	0.9530

Tablo 5.3'te PSO uygulanması sonucunda en düşük uygunluk değerinin elde edildiği iterasyonda seçilen alt özellik kümesine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme algoritmaları 5-kat çapraz doğrulama algoritmaları uygulanmıştır. Farklı özellik sayısındaki veri kümelerinden elde edilen doğruluk, duyarlılık, özgüllük ve f1-ölçütü değerleri Tablo 5.9'da verilmiştir. ACRIMA veri kümesinden Inceptionv3 mimarisi sonrası PSO uygulanması ile seçilen özellik sayısı en az 429 ve en fazla 492'dir. 452 ve 492 özellik sayısına sahip veri kümeleri için en yüksek doğruluk ve duyarlılık değerleri sırasıyla SVM ve Topluluk Öğrenme algoritmaları ile edilmiştir. En yüksek özgüllük ve f1-ölçütü değerleri ise SVM algoritması elde edilmiştir. Orijinal veri kümesindeki ve InceptionV3+PSO ile elde edilen alt veri kümelerindeki sonuçlar kıyaslandığında genel olarak tüm algoritmaların başarı oranının arttığı gözlemlenmiştir.

**Tablo 5. 9.** InceptionV3+PSO için farklı sınıflandırıcı performansları

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	452	0.7666	0.8130	0.7160	0.7843
Naive Bayes	452	0.8170	0.8414	0.7866	0.8361
SVM	452	<b>0.8979</b>	0.8894	<b>0.9100</b>	<b>0.9113</b>
KNN	452	0.8255	0.8354	0.8121	0.8468
Karar Ağacı	452	0.7248	0.7512	0.6898	0.7569
Topluluk Öğrenme	452	0.8865	<b>0.8970</b>	0.8730	0.8992

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	466	0.7291	0.7855	0.6705	0.7470
Naive Bayes	466	0.8241	0.8417	0.8013	0.8438
SVM	466	<b>0.8879</b>	<b>0.8953</b>	<b>0.8783</b>	<b>0.9009</b>
KNN	466	0.8326	0.8145	0.8631	0.8592
Karar Ağacı	466	0.7319	0.7435	0.7143	0.7698
Topluluk Öğrenme	466	0.8865	0.8950	0.8754	0.8995

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	492	0.7106	0.7567	0.6586	0.7351
Naive Bayes	492	0.8113	0.8346	0.7821	0.8314
SVM	492	<b>0.8979</b>	0.8876	<b>0.9129</b>	<b>0.9115</b>
KNN	492	0.8255	0.8354	0.8121	0.8468
Karar Ağacı	492	0.7901	0.7911	0.7885	0.8200
Topluluk Öğrenme	492	0.8922	<b>0.8922</b>	0.8923	0.9055

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	429	0.7660	0.7969	0.7278	0.7898
Naive Bayes	429	0.8028	0.8320	0.7673	0.8225
SVM	429	0.8965	0.8800	<b>0.9214</b>	<b>0.9111</b>
KNN	429	0.8383	0.8543	0.8176	0.8564
Karar Ağacı	429	0.7674	0.7613	0.7778	0.8048
Topluluk Öğrenme	429	<b>0.8993</b>	<b>0.9052</b>	0.8914	0.9109

Tablo 5.4'te PSO uygulanması sonucunda en düşük uygunluk değerinin elde edildiği iterasyonda seçilen alt özellik kümesine Lojistik Regresyon, Naive Bayes, SVM, KNN, Karar Ağacı ve Topluluk Öğrenme algoritmaları 5-kat çapraz doğrulama algoritmaları uygulanmıştır. Farklı özellik sayısındaki veri kümelerinden elde edilen doğruluk, duyarlılık, özgüllük ve f1-ölçütü değerleri Tablo 5.10'da verilmiştir. ACRIMA veri kümesinden DenseNet201 mimarisi sonrası PSO uygulanması ile seçilen özellik sayısı en az 361 ve en fazla 471'dir. Seçilen özellik sayısı 471 iken en yüksek doğruluk, duyarlılık, özgüllük ve f1-ölçütü değerleri Topluluk öğrenme algoritması ile 380 iken ise SVM algoritması ile elde edilmiştir. En düşük doğruluk değerleri ise Karar Ağacı algoritması ile elde edilmiştir. 361 ve 388 özellik sayısına sahip veri kümeleri için ise en iyi doğruluk, özgüllük ve f1-ölçütü değerleri Topluluk Öğrenme algoritması ile en iyi duyarlılık değerleri ise SVM algoritması ile elde edilmiştir. Orijinal veri kümesindeki ve DenseNet201+PSO

ile elde edilen alt veri kümelerindeki sonuçlar kıyaslandığında tüm algoritmaların başarımlarının arttığı gözlemlenmiştir.

**Tablo 5. 10.** DenseNet201+ PSO için farklı sınıflandırıcı performansları

Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	471	0.8071	0.8250	0.7836	0.8291
Naive Bayes	471	0.9050	0.8945	0.9201	0.9176
SVM	471	0.9645	0.9603	0.9702	0.9687
KNN	471	0.9092	0.8722	0.9730	0.9240
Karar Ağacı	471	0.8113	0.7929	0.8438	0.8426
Topluluk Öğrenme	471	<b>0.9702</b>	<b>0.9676</b>	<b>0.9737</b>	<b>0.9737</b>
Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	380	0.8936	0.9063	0.8774	0.9052
Naive Bayes	380	0.9092	0.8971	0.9268	0.9214
SVM	380	<b>0.9773</b>	<b>0.9726</b>	<b>0.9835</b>	<b>0.9799</b>
KNN	380	0.9021	0.8724	0.9511	0.9174
Karar Ağacı	380	0.8028	0.8142	0.7872	0.8273
Topluluk Öğrenme	380	0.9702	0.9630	0.9800	0.9738
Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	361	0.9106	0.9132	0.9073	0.9212
Naive Bayes	361	0.9035	0.8923	0.9199	0.9165
SVM	361	0.9716	<b>0.9653</b>	0.9801	0.9750
KNN	361	0.9163	0.8821	0.9735	0.9295
Karar Ağacı	361	0.8213	0.8125	0.8352	0.8478
Topluluk Öğrenme	361	<b>0.9730</b>	0.9631	<b>0.9866</b>	<b>0.9763</b>
Sınıflandırıcı	Seçilen Özellik Sayısı	Doğruluk	Duyarlılık	Özgüllük	F1-Ölçütü
Lojistik Regresyon	388	0.9007	0.9035	0.8970	0.9125
Naive Bayes	388	0.9007	0.8995	0.9024	0.9129
SVM	388	0.9660	<b>0.9673</b>	0.9642	0.9698
KNN	388	0.9078	0.8719	0.9692	0.9227
Karar Ağacı	388	0.8299	0.8492	0.8026	0.8541
Topluluk Öğrenme	388	<b>0.9674</b>	0.9628	<b>0.9735</b>	<b>0.9712</b>

Tüm deney sonuçları incelendiğinde, ACRIMA veri kümesinden glokom hastalığının tespiti için en iyi başarıyı DenseNet201 çıkarılan özelliklere PSO uygulanması sonucunda 0.9773 doğruluk değeri ile SVM algoritması göstermiştir.

## 6. SONUÇLAR

Glokom tedavisinin bulunmaması nedeniyle erken teşhis ve tanı, hastalığın önlenmesi açısından önemli hale gelmiştir. Buna bağlı olarak glokom semptomlarının ancak hastalık ileri düzeye ilerledikten sonra ortaya çıktığı göz önüne alındığında, glokomun erken teşhisi zordur. Sonuç olarak rutin glokom taraması kritik önem taşımaktadır ve uzmanlar tarafından tavsiye edilmektedir. Ancak göz taraması şu anda subjektif, yoğun emek ve zaman alıcı bir işlemdir ve erişilebilen göz uzmanları kısıtlıdır. Bu çalışmada sürü tabanlı metasezgisel algoritmalara dayalı olarak glokom tespitine yönelik farklı aşamalardan oluşan bir yaklaşım önerilmiştir. Öncelikle glokom ve normal bireylere ait fundus görüntülerinden oluşan veri kümesine ESA mimarileri uygulanarak özellik çıkarımı gerçekleştirilmiştir. İki farklı mimari ile elde edilen özellik kümelerinden ilgili ve kullanışlı özelliklerin belirlenmesi için metasezgisel özellik seçimi yöntemleri kullanılmıştır. Metasezgisel algoritmalarından yeni önerilen Balina Optimizasyon algoritması ve literatürde farklı problemler için oldukça etkili biçimde kullanılan ve iyi bilinen Parçacık Sürüsü Optimizasyon algoritması özellik seçimi yöntemi olarak tercih edilmiştir. Bu algoritmalar veri kümelerine farklı popülasyon büyüklüğü ve iterasyon sayılarında uygulanarak en iyi uygunluk değerinin elde edilmesi amaçlanmıştır. Böylelikle InceptionV3 ve DenseNet201 mimarileri ile elde edilen iki farklı özellik kümesine iki farklı metasezgisel algoritma özellik seçimi uygulayarak dört farklı özellik kümesi elde edilmiştir. Bu dört farklı yeni veri kümesi ve orijinal veri kümesine Lojistik Regresyon, Naive Bayes, Destek Vektör Makinesi, K-En Yakın Komşu, Karar Ağacı ve Topluluk Öğrenme algoritmaları glokom hastalığının tespiti için uygulanmıştır. Değerlendirmeler doğruluk, duyarlılık, özgüllük ve f1-ölçütlerine göre yapılmıştır. Karşılaştırmalar orijinal veri kümesi ve dört alt özellik veri kümesi arasında gerçekleştirilmiştir. Yapılan karşılaştırmalar metasezgisel algoritmalar ile özellik seçiminin sınıflandırma algoritmalarının performansının artırdığı görülmüştür.

Yapılan çalışmada orijinal veri kümelerinden en yüksek doğruluk, duyarlılık, özgüllük ve f1-ölçülü sırasıyla 0.9730, 0.9677, 0.9801 ve 0.9762 olarak elde edilmiştir. Metasezgisel algoritmalar ile özellik seçimi ile elde edilen alt veri kümelerindeki en yüksek doğruluk, duyarlılık, özgüllük ve f1-ölçülü olarak sırasıyla 0.9773, 0.9726, 0.9866 ve 0.9799'dur. Elde edilen sonuçlar metasezgisel algoritmalar ile optimal özellik seçimine dayalı hastalık tespitinin gelecek çalışmalar için umut verici olduğu gözlemlenmiştir.

## ÖNERİLER

Tez çalışması ile elde edilen sonuçlara göre metasezgisel algoritmaların glokom hastalığının tespiti için sınıflandırma algoritmalarının performansını artırdığı görülmüştür. Önerilen yöntem farklı fundus görüntüleri üzerinde değerlendirilebilir. Ayrıca bu çalışma kullanılan yöntem farklı bilim alanlarında farklı problemlerin çözümüne uyarlanabilir. Metasezgisel algoritmaların ikili, paralel, çok amaçlı veya farklı metasezgisel algoritmalar ile hibrit versiyonları geliştirilerek glokom hastalığının tespiti için kullanılabilir. Önerilen çalışma, oftalmologlara oküler hastalıkları erken evrelerde tespit etme ve teşhis etme konusunda yardımcı olabilecek gerçek zamanlı bir karar destek sistemi oluşturacak şekilde genişletilebilir.



## KAYNAKLAR

- [1] D. L. Budenz et al., “Prevalence of Glaucoma in an Urban West African Population: The Tema Eye Survey,” *JAMA Ophthalmol.*, vol. 131, no. 5, pp. 651–658, May 2013, doi: 10.1001/JAMAOPHTHALMOL.2013.1686.
- [2] L. K. Singh, M. Khanna, and Pooja, “A novel multimodality based dual fusion integrated approach for efficient and early prediction of glaucoma,” *Biomed. Signal Process. Control*, vol. 73, 2022, doi: 10.1016/j.bspc.2021.103468.
- [3] WH Organization, “World Report on Vision,” *World Heal. Organ.*, 2019, [Online]. Available: <https://www.who.int/>
- [4] B. Akalın and Ü. Veranyurt, “Sağlık Hizmetleri ve Yönetiminde Yapay Zekâ,” *Acta Infologica*, vol. 5, no. 1, pp. 231–240, 2021, doi: 10.26650/acin.750857.
- [5] M. Bilen, “Yapay Zekânın Değiştirdiği Dinamikler”, Eğitim Yayınevi, İstanbul, 2023.
- [6] D. A. Atchison, “Optics of the Human Eye”, CRC Press, Amerika Birleşik Devletleri, 2023.
- [7] Y. B. Özçelik and A. Altan, “Diyabetik Retinopati Teşhisi için Fundus Görüntülerinin Derin Öğrenme Tabanlı Sınıflandırılması,” *Eur. J. Sci. Technol.*, no. 29, pp. 156–167, 2021, doi: 10.31590/ejosat.1011806.
- [8] H. Fu et al., “Disc-Aware Ensemble Network for Glaucoma Screening from Fundus Image,” *IEEE Trans. Med. Imaging*, vol. 37, no. 11, pp. 2493–2501, 2018, doi: 10.1109/TMI.2018.2837012.
- [9] O. Karaman, “Sağlık & Bilim 2022 Medikal Araştırmalar-IV”, Efe Akademi Yayınları, İstanbul, 2022.
- [10] D. R. Nayak, D. Das, B. Majhi, S. V. Bhandary, and U. R. Acharya, “ECNet: An evolutionary convolutional network for automated glaucoma detection using fundus images,” *Biomed. Signal Process. Control*, vol. 67, no. March, p. 102559, 2021, doi: 10.1016/j.bspc.2021.102559
- [11] H. Fu et al., “Disc-Aware Ensemble Network for Glaucoma Screening from Fundus Image,” *IEEE Trans. Med. Imaging*, vol. 37, no. 11, pp. 2493–2501, 2018, doi: 10.1109/TMI.2018.2837012.
- [12] H. Fu et al., “Segmentation and quantification for angle-closure glaucoma assessment in anterior segment oct,” *IEEE Trans. Med. Imaging*, vol. 36, no. 9, pp. 1930–1938, 2017, doi: 10.1109/TMI.2017.2703147.
- [13] U. Raghavendra, H. Fujita, S. V. Bhandary, A. Gudigar, J. H. Tan, and U. R. Acharya, “Deep convolution neural network for accurate diagnosis of glaucoma using digital fundus images,” *Inf. Sci. (Ny)*, vol. 441, pp. 41–49, May 2018, doi: 10.1016/J.INS.2018.01.051.
- [14] A. Z. H. Ooi et al., “Interactive blood vessel segmentation from retinal fundus image based on canny edge detector,” *Sensors*, vol. 21, no. 19, pp. 1–22, 2021, doi: 10.3390/s21196380.
- [15] U. Raghavendra, H. Fujita, S. V. Bhandary, A. Gudigar, J. H. Tan, and U. R. Acharya, “Deep convolution neural network for accurate diagnosis of glaucoma using digital fundus images,” *Inf. Sci. (Ny)*, vol. 441, pp. 41–49, May 2018, doi: 10.1016/J.INS.2018.01.051.
- [16] S. Muchuchuti and S. Viriri, “Retinal Disease Detection Using Deep Learning Techniques: A Comprehensive Review,” *J. Imaging*, vol. 9, no. 4, 2023, doi: 10.3390/jimaging9040084.
- [17] A. Kaderli, A. Karalezli, C. Kaya, Ç. Yiğit, D. Çevik Kaşıkçı, G. Alaçamlı, ... & S. B. Sarı, “Tıp Fakültesi Öğrencileri için Göz Hastalıkları”, Muğla Sıtkı Koçman Üniversitesi, 2021
- [18] B. Al-Bander, B. M. Williams, W. Al-Nuaimy, M. A. Al-Tae, H. Pratt, and Y. Zheng, “Dense fully convolutional segmentation of the optic disc and cup in colour fundus for glaucoma diagnosis,” *Symmetry (Basel)*, vol. 10, no. 4, 2018, doi: 10.3390/sym10040087.
- [19] Citation Styles, <https://www.sagarbargava.com/types-of-glaucoma/>, Erişim: 12 Eylül 2023.
- [20] M. Raju, K. P. Shanmugam, and C. R. Shyu, “Application of Machine Learning Predictive Models for Early Detection of Glaucoma Using Real World Data,” *Appl. Sci.*, vol. 13, no. 4, 2023, doi: 10.3390/app13042445.

- [21] A. R. Prananda, E. L. Frannita, A. H. T. Hutami, M. R. Maarif, N. L. Fitriyani, and M. Syafrudin, "Retinal Nerve Fiber Layer Analysis Using Deep Learning to Improve Glaucoma Detection in Eye Disease Assessment," *Appl. Sci.*, vol. 13, no. 1, 2023, doi: 10.3390/app13010037.
- [22] M. Vinícius dos Santos Ferreira, A. Oseas de Carvalho Filho, A. Dalília de Sousa, A. Corrêa Silva, and M. Gattass, "Convolutional neural network and texture descriptor-based automatic detection and diagnosis of glaucoma," *Expert Syst. Appl.*, vol. 110, pp. 250–263, 2018, doi: 10.1016/j.eswa.2018.06.010.
- [23] S. Simonthomas, N. Thulasi, and P. Asharaf, "Automated diagnosis of glaucoma using Haralick texture features," 2014 Int. Conf. Inf. Commun. Embed. Syst. ICICES 2014, no. 978, 2015, doi: 10.1109/ICICES.2014.7033743.
- [24] A. Sungheetha and R. Sharma R, "Design an Early Detection and Classification for Diabetic Retinopathy by Deep Feature Extraction based Convolution Neural Network," *J. Trends Comput. Sci. Smart Technol.*, vol. 3, no. 2, pp. 81–94, 2021, doi: 10.36548/jtcsst.2021.2.002
- [25] M. Juneja et al., "Automated detection of Glaucoma using deep learning convolution network (G-net)," *Multimed. Tools Appl.*, vol. 79, no. 21–22, pp. 15531–15553, 2020, doi: 10.1007/s11042-019-7460-4.
- [26] U. R. Acharya et al., "Decision support system for the glaucoma using Gabor transformation," *Biomed. Signal Process. Control*, vol. 15, pp. 18–26, 2015, doi: 10.1016/j.bspc.2014.09.004.
- [27] A. Saxena, A. Vyas, L. Parashar, and U. Singh, "A Glaucoma Detection using Convolutional Neural Network," *Proc. Int. Conf. Electron. Sustain. Commun. Syst. ICESC 2020*, no. Icesc, pp. 815–820, 2020, doi: 10.1109/ICESC48915.2020.9155930.
- [28] M. S. Guru Prasad, H. N. Naveen Kumar, K. Raju, D. K. Santhosh Kumar, and S. Chandrappa, "Glaucoma Detection Using Clustering and Segmentation of the Optic Disc Region from Retinal Fundus Images," *SN Comput. Sci.*, vol. 4, no. 2, Mar. 2023, doi: 10.1007/S42979-022-01592-1.
- [29] B. Dinç and Y. Kaya, "A Novel Hybrid Optic Disc Detection and Fovea Localization Method Integrating Region-Based Convnet and Mathematical Approach," *Wirel. Pers. Commun.*, vol. 129, no. 4, pp. 2727–2748, Apr. 2023, doi: 10.1007/S11277-023-10255-0.
- [30] R. Praveena and T. R. GaneshBabu, "Determination of cup to disc ratio using unsupervised machine learning techniques for glaucoma detection," *MCB Mol. Cell. Biomech.*, vol. 18, no. 2, pp. 69–86, 2021, doi: 10.32604/MCB.2021.014622.
- [31] R. Kashyap, R. Nair, S. M. P. Gangadharan, M. Botto-Tobar, S. Farooq, and A. Rizwan, "Glaucoma Detection and Classification Using Improved U-Net Deep Learning Model," *Healthc.*, vol. 10, no. 12, 2022, doi: 10.3390/healthcare10122497.
- [32] A. Shoukat, S. Akbar, S. A. Hassan, S. Iqbal, A. Mehmood, and Q. M. Ilyas, "Automatic Diagnosis of Glaucoma from Retinal Images Using Deep Learning Approach," *Diagnostics*, vol. 13, no. 10, pp. 1–17, 2023, doi: 10.3390/diagnostics13101738.
- [33] A. Khan, K. Pin, A. Aziz, J. W. Han, and Y. Nam, "Optical Coherence Tomography Image Classification Using Hybrid Deep Learning and Ant Colony Optimization," *Sensors*, vol. 23, no. 15, 2023, doi: 10.3390/s23156706.
- [34] N. D. Clement and A. H. R. W. Simpson, "Artificial intelligence in orthopaedics What Level Of Evidence Does It Represent And How Is It Validated?," *Bone Jt. Res.*, vol. 12, no. 8, pp. 494–496, 2023, doi: 10.1302/2046-3758.128.BJR-2023-0199.
- [35] S. Sado, I. Jastrz, W. Zelik, and J. Szczerba, "Current State of Application of Machine Learning for Investigation of MgO-C Refractories : A Review," pp. 1–18, 2023.
- [36] K. K. Hiran, R. K. Jain, K. Lakhwani, and R. Doshi, "Machine Learning: Master Supervised and Unsupervised Learning Algorithms with Real Examples (English Edition)", BPB Publications, 2021.
- [37] T. Talaei Khoei and N. Kaabouch, "Machine Learning: Models, Challenges, and Research Directions," *Futur. Internet*, vol. 15, no. 10, 2023, doi: 10.3390/fi15100332.

- [38] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Mach. Learn.*, vol. 109, no. 2, pp. 373–440, 2020, doi: 10.1007/s10994-019-05855-6.
- [39] M. Davis, "Desert / Forest Fire Detection Using Machine / Deep Learning Techniques," pp. 1–20, 2023.
- [40] W. Zhao, Z. Wang, H. Zhang, and T. Wang, "A Model Predicting the Maximum Face Slab Deflection of Concrete-Face Rockfill Dams: Combining Improved Support Vector Machine and Threshold Regression," *Water (Switzerland)*, vol. 15, no. 19, 2023, doi: 10.3390/w15193474.
- [41] L. D. Mang et al., "Automatic Robust Crackle Detection and Localization Approach Using AR-Based Spectral Estimation and Support Vector Machine," *Appl. Sci.*, vol. 13, no. 19, 2023, doi: 10.3390/app131910683.
- [42] J. Singh, D. Thakur, F. Ali, T. Gera, and K. S. Kwak, "Deep feature extraction and classification of android malware images," *Sensors (Switzerland)*, vol. 20, no. 24, pp. 1–29, 2020, doi: 10.3390/s20247013.
- [43] D. Kılınç, E. Borandağ, F. Yücalar, V. Tunalı, M. Şimşek, and A. Özçift, "KNN Algoritması ve R Dili ile Metin Madenciliği Kullanılarak Bilimsel Makale Tasnifi," *Marmara Fen Bilim. Derg.*, vol. 28, no. 3, pp. 89–94, 2016, doi: 10.7240/mufbed.69674.
- [44] M. F. Ak, "A comparative analysis of breast cancer detection and diagnosis using data visualization and machine learning applications," *Healthc.*, vol. 8, no. 2, 2020, doi: 10.3390/healthcare8020111.
- [45] F. Wang, Z. Zhen, B. Wang, and Z. Mi, "applied sciences Comparative Study on KNN and SVM Based Weather Classification Models for Day Ahead Short Term Solar PV Power Forecasting", doi: 10.3390/app8010028.
- [46] J. Talukdar, D. K. Gogoi, and T. P. Singh, "A comparative assessment of most widely used machine learning classifiers for analysing and classifying autism spectrum disorder in toddlers and adolescents," *Healthc. Anal.*, vol. 3, no. February, p. 100178, 2023, doi: 10.1016/j.health.2023.100178.
- [47] S. Zahid, F. Hussain, M. Rashid, M. H. Yousaf, and H. A. Habib, "Optimized Audio Classification and Segmentation Algorithm by Using Ensemble Methods," *Math. Probl. Eng.*, vol. 2015, 2015, doi: 10.1155/2015/209814.
- [48] J. Wang, D. Zhang, and Y. Zhou, "Ensemble deep learning for automated classification of power quality disturbances signals," *Electr. Power Syst. Res.*, vol. 213, no. August, p. 108695, 2022, doi: 10.1016/j.epr.2022.108695.
- [49] J. A. Jose, C. S. Kumar, and S. Sureshkumar, "Tuna classification using super learner ensemble of region-based CNN-grouped 2D-LBP models," *Inf. Process. Agric.*, vol. 9, no. 1, pp. 68–79, 2022, doi: 10.1016/j.inpa.2021.01.001.
- [50] R. Yakkundimath, V. Jadhav, B. Anami, and N. Malvade, "Co-occurrence histogram based ensemble of classifiers for classification of cervical cancer cells," *J. Electron. Sci. Technol.*, vol. 20, no. 3, pp. 1–10, 2022, doi: 10.1016/J.JNLEST.2022.100170.
- [51] S. J. Park, C. W. Lee, S. Lee, and M. J. Lee, "Landslide susceptibility mapping and comparison using decision tree models: A case study of Jumunjin Area, Korea," *Remote Sens.*, vol. 10, no. 10, 2018, doi: 10.3390/rs10101545.
- [52] J. Brasil, C. Maitelli, J. Nascimento, O. Chiavone-Filho, and E. Galvão, "Diagnosis of Operating Conditions of the Electrical Submersible Pump via Machine Learning," *Sensors*, vol. 23, no. 1, 2023, doi: 10.3390/s23010279.
- [53] D. Atasoy and H. Kara, "Karar Ağacı Optimizasyon Algoritması üzerine Bir Çalışma", *J. Inst. Sci. and Tech.*, vol. 13, no. 2, pp. 1247–1255, 2023, doi: 10.21597/jist.1214796.
- [54] A. El Alaoui El Fels, L. Mandi, A. Kammoun, N. Ouazzani, O. Monga, and M. L. Hbid, "Artificial Intelligence and Wastewater Treatment: A Global Scientific Perspective through Text Mining," *Water (Switzerland)*, vol. 15, no. 19, 2023, doi: 10.3390/w15193487.

- [55] M. Pérez-Gomariz, A. López-Gómez, and F. Cerdán-Cartagena, “Artificial Neural Networks as Artificial Intelligence Technique for Energy Saving in Refrigeration Systems—A Review,” *Clean Technol.*, vol. 5, no. 1, pp. 116–136, 2023, doi: 10.3390/cleantechnol5010007.
- [56] K. Ozturk and M. Şahin, “Yapay Sinir Ağları ve Yapay Zekâ’ya Genel Bir Bakış-A General View of Artificial Neural Networks and Artificial Intelligence,” *Tak. Vekayi*, vol. 6, no. 2, pp. 25–36, 2018.
- [57] B. Li, L. Wang, Q. Jiang, and W. Li, “applied sciences Sports Risk Prediction Model Based on Automatic Encoder and Convolutional Neural Network,” 2023.
- [58] S. Jafari, “A CNN-GRU Approach to the Accurate Prediction of Batteries ’ Remaining Useful Life from Charging Profiles,” pp. 1–25, 2023.
- [59] E. Albayrak, “Derin Öğrenme İle İha Görüntülerinden Nesne Tespitinin Yapılması”, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, Bilecik Şeyh Edebali Üniversitesi, 2021.
- [60] M. Wang, S. Chi, and S. Lu, “applied sciences Power Quality Fault Identification Method Based on SDP and Convolutional Neural Network,” 2023.
- [61] J. Hasan and F. Piltan, “LDDNet : A Deep Learning Framework for the Diagnosis of Infectious Lung Diseases,” 2023.
- [62] S. Gite, A. Mishra, and K. Kotecha, “Enhanced lung image segmentation using deep learning,” *Neural Comput. Appl.*, vol. 35, no. 31, pp. 22839–22853, 2022, doi: 10.1007/s00521-021-06719-8.
- [63] Malik, S., Tyagi, A. K., & Mahajan, S., “Architecture, Generative Model, and Deep Reinforcement Learning for IoT Applications: Deep Learning Perspective”, In *Artificial Intelligence-based Internet of Things Systems*, pp. 243-265, Springer, Cham, 2022.
- [64] Q. Jiang et al., “Research on Intelligent Recognition Algorithm of Pneumonia Based on Deep Convolution and Attention Neural Network,” vol. 2021, 2021.
- [65] N. V. Sridharan, J. V. Joseph, S. Vaithyanathan, and M. Aghaei, “Weightless Neural Network-Based Detection and Diagnosis of Visual Faults in Photovoltaic Modules,” 2023.
- [66] A. Chandak and N. Chaturvedi, “Machine-Learning-Based Human Fall Detection Using Contact-and Noncontact-Based Sensors,” vol. 2022, 2022.
- [67] B. Üniversitesi, G. M. Yüksekokulu, B. Teknolojileri, F. Üniversitesi, M. Fakültesi, and B. Mühendisliği, “Deprem Sonrası Bina Yüzeylerinde Meydana Gelen Çatlakların Tespitinde Sınıflandırma Modellerinin Karşılaştırılması Surfaces After Earthquake,” vol. 37, no. December, pp. 899–909, 2022.
- [68] E. Hitimana et al., “An Intelligent System-Based Coffee Plant Leaf Disease Recognition Using Deep Learning Techniques on Rwandan Arabica Dataset,” *Technologies*, vol. 11, no. 5, 2023, doi: 10.3390/technologies11050116.
- [69] C. Figueroa-Flores and P. San-Martin, “Deep learning for Chilean native flora classification: a comparative analysis,” *Front. Plant Sci.*, vol. 14, no. September, pp. 1–13, 2023, doi: 10.3389/fpls.2023.1211490.
- [70] Q. Guan et al., “Deep convolutional neural network Inception-v3 model for differential diagnosing of lymph node in cytological images: a pilot study,” *Ann. Transl. Med.*, vol. 7, no. 14, pp. 307–307, 2019, doi: 10.21037/atm.2019.06.29.
- [71] R. Kalaiarasan, K. Madhan Kumar, S. Sridhar, and M. Yuvarai, “Deep Learning-based Transfer Learning for Classification of Skin Cancer,” *Proc. - Int. Conf. Appl. Artif. Intell. Comput. ICAAIC 2022*, pp. 450–454, 2022, doi: 10.1109/ICAAIC53929.2022.9792651.
- [72] H. Üzen, M. Türkoğlu, A. Arı, and D. Hanbay, “InceptionV3 based enriched feature integration network architecture for pixel-level surface defect detection,” *J. Fac. Eng. Archit. Gazi Univ.*, vol. 38, no. 2, pp. 721–732, 2023, doi: 10.17341/gazimmfd.1024425.

- [73] R. Arun Kumar, J. Vijay Franklin, and N. Koppula, "A Comprehensive Survey on Metaheuristic Algorithm for Feature Selection Techniques," *Mater. Today Proc.*, vol. 64, pp. 435–441, Jan. 2022, doi: 10.1016/J.MATPR.2022.04.803.
- [74] N. Singh and P. Singh, "A hybrid ensemble-filter wrapper feature selection approach for medical data classification," *Chemom. Intell. Lab. Syst.*, vol. 217, p. 104396, Oct. 2021, doi: 10.1016/J.CHEMOLAB.2021.104396.
- [75] H. H. Hsu, C. W. Hsieh, and M. Da Lu, "Hybrid feature selection by combining filters and wrappers," *Expert Syst. Appl.*, vol. 38, no. 7, pp. 8144–8150, Jul. 2011, doi: 10.1016/J.ESWA.2010.12.156.
- [76] O. O. Akinola, A. E. Ezugwu, J. O. Agushaka, R. A. Zitar, and L. Abualigah, "Multiclass feature selection with metaheuristic optimization algorithms: a review," *Neural Comput. Appl.*, vol. 34, no. 22, pp. 19751–19790, 2022, doi: 10.1007/s00521-022-07705-4.
- [77] H. Shayanfar and F. S. Gharehchopogh, "Farmland fertility: A new metaheuristic algorithm for solving continuous optimization problems," *Appl. Soft Comput.*, vol. 71, pp. 728–746, Oct. 2018, doi: 10.1016/J.ASOC.2018.07.033.
- [78] L. Ingber, "Very fast simulated re-annealing," *Math. Comput. Model.*, vol. 12, no. 8, pp. 967–973, Jan. 1989, doi: 10.1016/0895-7177(89)90202-1.
- [79] F. S. Gharehchopogh and H. Gholizadeh, "A comprehensive survey: Whale Optimization Algorithm and its applications," *Swarm Evol. Comput.*, vol. 48, pp. 1–24, Aug. 2019, doi: 10.1016/J.SWEVO.2019.03.004.
- [80] G. Dhiman and V. Kumar, "Emperor penguin optimizer: A bio-inspired algorithm for engineering problems," *Knowledge-Based Syst.*, vol. 159, pp. 20–50, Nov. 2018, doi: 10.1016/J.KNOSYS.2018.06.001.
- [81] J. H. Holland, "Adaptation in neural and artificial system," Ann Arbor, Univeristy Michigan Press, 1975.
- [82] A. Bhattacharya and P. K. Chattopadhyay, "Solving complex economic load dispatch problems using biogeography-based optimization," *Expert Syst. Appl.*, vol. 37, no. 5, pp. 3605–3615, May 2010, doi: 10.1016/J.ESWA.2009.10.031.
- [83] K. R. Opara and J. Arabas, "Differential Evolution: A survey of theoretical analyses," *Swarm Evol. Comput.*, vol. 44, pp. 546–558, Feb. 2019, doi: 10.1016/J.SWEVO.2018.06.010.
- [84] A. Kaveh and M. Khayatazad, "A new meta-heuristic method: Ray Optimization," *Comput. Struct.*, vol. 112–113, pp. 283–294, Dec. 2012, doi: 10.1016/J.COMPSTRUC.2012.09.003.
- [85] H. Abedinpourshotorban, S. Mariyam Shamsuddin, Z. Beheshti, and D. N. A. Jawawi, "Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm," *Swarm Evol. Comput.*, vol. 26, pp. 8–22, Feb. 2016, doi: 10.1016/J.SWEVO.2015.07.002.
- [86] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm," *Inf. Sci. (Ny)*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009, doi: 10.1016/J.INS.2009.03.004.
- [87] T. Fearn, "Particle swarm optimisation," *NIR news*, vol. 25, no. 1, p. 27, 2014.
- [88] S. Mirjalili, A. L.-A. in engineering software, and undefined 2016, "The whale optimization algorithm," *Elsevier*, vol. 95, pp. 51–67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.
- [89] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/J.ADVENGSOFT.2013.12.007.
- [90] S. Mirjalili, A. L.-A. in engineering software, and undefined 2016, "The whale optimization algorithm," *Elsevier*, vol. 95, pp. 51–67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.
- [91] F. S. Gharehchopogh and H. Gholizadeh, "A comprehensive survey: Whale Optimization Algorithm and its applications," *Swarm Evol. Comput.*, vol. 48, no. March, pp. 1–24, 2019, doi: 10.1016/j.swevo.2019.03.004.

- [92] M. Y. Silaa, O. Barambones, and A. Bencherif, "A Novel Adaptive PID Controller Design for a PEM Fuel Cell Using Stochastic Gradient Descent with Momentum Enhanced by Whale Optimizer," *Electron.*, vol. 11, no. 16, 2022, doi: 10.3390/electronics11162610.
- [93] G. Kaur and S. Arora, "Chaotic whale optimization algorithm," *J. Comput. Des. Eng.*, vol. 5, no. 3, pp. 275–284, 2018, doi: 10.1016/j.jcde.2017.12.006.
- [94] J. A. Goldbogen, A. S. Friedlaender, J. Calambokidis, M. F. McKenna, M. Simon, and D. P. Nowacek, "Integrative Approaches to the Study of Baleen Whale Diving Behavior, Feeding Performance, and Foraging Ecology," *Bioscience*, vol. 63, no. 2, pp. 90–100, Feb. 2013, doi: 10.1525/bio.2013.63.2.5.
- [95] H. Alahmer, A. Alahmer, R. Alkhalaf, and M. Alrbai, "Exhaust emission reduction of a SI engine using acetone–gasoline fuel blends: Modeling, prediction, and whale optimization algorithm," *Energy Reports*, vol. 9, pp. 77–86, 2023, doi: 10.1016/j.egyr.2022.10.360.
- [96] M. Hemeida, D. Osheba, S. Alkhalaf, A. Fawzy, M. Ahmed, and M. Roshdy, "Optimized PID controller using Archimedes optimization algorithm for transient stability enhancement," *Ain Shams Eng. J.*, vol. 14, no. 10, p. 102174, 2023, doi: 10.1016/j.asej.2023.102174.
- [97] J. Kennedy, & R. Eberhart, "Particle swarm optimization", In *Proceedings of ICNN'95-international conference on neural networks*, vol. 4, pp. 1942-1948, IEEE, 1995.
- [98] G. S. Fernandez et al., "Optimal dynamic scheduling of electric vehicles in a parking lot using particle swarm optimization and shuffled frog leaping algorithm," *Energies*, vol. 13, no. 23, 2020, doi: 10.3390/en13236384.
- [99] C. Y. Lee, K. Y. Huang, Y. X. Shen, and Y. C. Lee, "Improved weighted k-nearest neighbor based on PSO for wind power system state recognition," *Energies*, vol. 13, no. 20, 2020, doi: 10.3390/en13205520.
- [100] M. A. Koşan, O. Yıldız, and H. Karacan, "Comparative analysis of machine learning algorithms in detection of phishing websites," *Pamukkale Univ. J. Eng. Sci.*, vol. 24, no. 2, pp. 276–282, 2018, doi: 10.5505/pajes.2017.27167.
- [101] H. Koçak, "Sınıflama Algoritmalarının Yağışın Varlığını Kestirme Konusundaki Performanslarının Karşılaştırması," *J. Intell. Syst. Theory Appl.*, vol. 5, no. 1, pp. 16–26, 2022, doi: 10.38016/jista.979285.

