

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

İSTATİSTİK VE BİLGİSAYAR BİLİMLERİ ANABİLİM DALI

**YANIK GÖRÜNTÜLERİNİN ÇOK DEĞİŞKENLİ İSTATİSTİKSEL
YÖNTEMLER VE DERİN ÖĞRENME YAKLAŞIMLARI İLE ANALİZİ**

DOKTORA TEZİ

Erdoğan KARAKULLUKÇU

**OCAK 2020
TRABZON**



KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

İSTATİSTİK VE BİLGİSAYAR BİLİMLERİ ANABİLİM DALI

**YANIK GÖRÜNTÜLERİNİN ÇOK DEĞİŞKENLİ İSTATİSTİKSEL
YÖNTEMLER VE DERİN ÖĞRENME YAKLAŞIMLARI İLE ANALİZİ**

Erdinç KARAKULLUKÇU

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde

DOKTOR (İSTATİSTİK)

Unvanı Verilmesi İçin Kabul Edilen Tezdir.

Tezin Enstitüye Verildiği Tarih : 31 / 12 / 2019

Tezin Savunma Tarihi : 31 / 01 / 2020

Tez Danışmanı : Dr. Öğr. Üyesi Uğur ŞEVİK

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**İstatistik ve Bilgisayar Bilimleri Anabilim Dalında
Erdoğan KARAKULLUKÇU Tarafından Hazırlanan**

**YANIK GÖRÜNTÜLERİNİN ÇOK DEĞİŞKENLİ İSTATİSTİKSEL
YÖNTEMLER VE DERİN ÖĞRENME YAKLAŞIMLARI İLE ANALİZİ**

başlıklı bu çalışma, Enstitü Yönetim Kurulunun 03 /01 /2020 gün ve 1835 sayılı
kararıyla oluşturulan jüri tarafından yapılan sınavda
DOKTORA TEZİ
olarak kabul edilmiştir.

Jüri Üyeleri

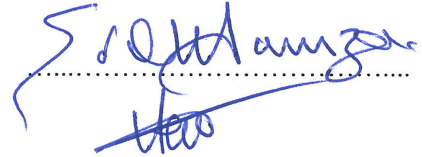
Başkan : Prof. Dr. Hamza EROL

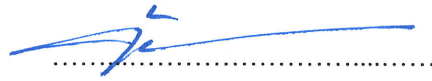
Üye : Prof. Dr. Recep Alp KUT

Üye : Prof. Dr. Serdar TÜRKYILMAZ

Üye : Dr. Öğr. Üyesi Uğur ŞEVİK

Üye : Dr. Öğr. Üyesi Tolga BERBER









**Prof. Dr. Asim KADIOĞLU
Enstitü Müdürü**

ÖNSÖZ

Doktora tezi olarak sunulan bu çalışmada, yanık hastalarından dijital kameralarla elde edilen iki boyutlu görüntüler üzerindeki sağlıklı ve yanık deri bölgelerinin tespit edilmesi amaçlanmıştır. Burada nihai hedef, bilgisayar ortamında insan derisinin ve yanığın yapısını tam anlamıyla öğrenebilmek ve KTÜ BAP FAY-2016-5588 Altyapı Projesi bünyesinde birden fazla stereo kameranın kullanımıyla kurulacak olan görüntüleme sisteminde, yanık hastalarının taranması sonrası oluşturulacak model üzerinden deriye ve yanığa ilişkin alanların tespit edilebilmesidir. Bu sayede hastaya ilişkin yanık yüzdesinin gerçeğe en uygun şekilde hesaplanabilmesi mümkün olacaktır.

Tez danışmanlığımı üstlenerek, konu seçimi ve çalışmaların yürütülmesi sırasında bilgisini ve tecrübesini benimle paylaşan ve yardımını esirgemeyen değerli hocam Sayın Dr. Öğr. Üyesi Uğur ŞEVİK'e teşekkürlerimi sunarım. Ayrıca yapıcı eleştirileri ve yöntem önerileri ile tezime büyük katkıda bulunan hocam Sayın Dr. Öğr. Üyesi Tolga BERBER'e, tezde kullandığım görüntülerin temini ve tıbbi bilgiler konusunda yardımlarından dolayı sayın hocam Prof. Dr. Serdar TÜRKYILMAZ'a ve bazı çalışmalarımızı ortak olarak yürüttüğümüz arkadaşım Arş. Gör. Yeşim AKBAŞ'a teşekkürü bir borç bilirim.

Tez çalışması sürecinde bana desteklerini esirgemeyen İstatistik ve Bilgisayar Bilimleri Bölümü'ndeki tüm değerli hocalarıma ve asistan arkadaşlarıma, uzun yıllardır dostluğu ile beni yalnız bırakmayan ve manevi desteğini her zaman üstümde hissettiğim sevgili arkadaşım Dr. Gülşen DELİKANLI AKBAY'a şükranlarımı sunarım.

Doktora eğitimim süresince maddi olarak beni destekleyen, TÜBİTAK 2211 Yurt İçi Lisansüstü Burs Programına ve sunduğu Bilimsel Araştırma Projesi imkânı ile tezime katkıda bulunan üniversitem Karadeniz Teknik Üniversitesi'ne teşekkür ederim.

Son olarak, doğduğum günden beri elimi tutan, verdiğim her kararda arkamda olan, beni ben olduğum için seven annem ve babama, benim için kardeşten öte can olan Uzm. Dr. Serdar KARAKULLUKÇU'ya, bana her koşulda tahammül eden, yüzüme gülümseme, hayatıma renk katan biricik Cancan'ıma ve doğdukları günden itibaren bana insanın kalbinde keşfedilmeyi bekleyen sınırsız sayıda duygunun olduğunu gösteren minik adamlarım Utku ve Uras'a sonsuz minnettarlığımı sunarım.

Erdoğan KARAKULLUKÇU

Trabzon 2020

TEZ ETİK BEYANNAMESİ

Doktora Tezi olarak sunduđum “Yanık Görüntülerinin Çok Deđişkenli İstatistiksel Yöntemler ve Derin Öğrenme Yaklaşımları ile Analizi” başlıklı bu çalışmayı baştan sona kadar danışmanım Dr. Öğr. Üyesi Uđur ŞEVİK’in sorumluluğunda tamamladıđımı, verileri/örnekleri kendim topladıđımı, deneyleri/analizleri ilgili laboratuvarlarda yaptıđımı/yaptırdıđımı, başka kaynaklardan aldıđım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiđimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandıđımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiđimi beyan ederim. 31/01/2020

Erdinç KARAKULLUKÇU

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VIII
SUMMARY	IX
ŞEKİLLER DİZİNİ	X
TABLolar DİZİNİ.....	XII
SEMBOLLER DİZİNİ	XIV
1. GENEL BİLGİLER.....	1
1.1. Giriş	1
1.2. Tezin Amacı ve Yapısı	2
1.3. Derinin Yapısı.....	5
1.4. Yanık.....	6
1.5. Etkenin Niteliğine Göre Yanık Türleri	6
1.5.1. Termal Yanıklar.....	6
1.5.2. Elektrik Yanıkları	7
1.5.3. İnhalasyon Yanıkları.....	7
1.5.4. Kimyasal Yanıklar	8
1.5.5. Radyasyon Yanıkları	8
1.6. Yanık Derinliğinin Sınıflandırılması	8
1.6.1. Birinci Derece Yanıklar	9
1.6.2. İkinci Derece Yanıklar.....	10
1.6.3. Üçüncü Derece Yanıklar.....	10
1.6.4. Dördüncü Derece Yanıklar	11
1.7. Yanık Yüzey Alanı	11
1.8. Yanığın Şiddeti	14
1.9. Dijital Görüntülerden Deri ve Yanık Alanı Tespitine İlişkin Literatürde Yapılan Çalışmalar	14
1.10. Yanık Görüntü Veri Tabanı.....	18
2. YAPILAN ÇALIŞMALAR.....	21

2.1.	Yanık Görüntülerinden Öznitelik Çıkarma İşlemi	21
2.1.1.	Gri Seviye Histogram İstatistikleri	21
2.1.2.	Haralick Doku Öznitelikleri	22
2.1.3.	Ortalama Parlaklık Değerleri	24
2.2.	Yanık Görüntülerinin Segmentasyonu	26
2.2.1.	k-Ortalamalar Yöntemi	26
2.2.2.	Bulanık c-Ortalamalar Yöntemi	27
2.2.3.	Beklenti Maksimizasyonu Yöntemi	29
2.2.4.	Basit Doğrusal İteratif Kümeleme	31
2.2.5.	Calinski-Harabasz İndeksi ile Optimum Küme Sayısının Belirlenmesi.....	33
2.3.	Yanık Görüntülerinin Sınıflandırılması.....	34
2.3.1.	Yapay Sinir Ağları.....	34
2.3.1.1.	Tek Katmanlı Algılayıcı (Perceptron)	37
2.3.1.2.	Çok Katmanlı Algılayıcı (Multilayer Perceptron).....	38
2.3.1.3.	İleri Beslemeli Yapay Sinir Ağları (Feedforward Networks).....	39
2.3.1.4.	Geri Beslemeli Yapay Sinir Ağları.....	40
2.3.1.5.	Gözetimli Öğrenme	40
2.3.1.6.	Gözetimsiz (Danışmansız/Denetimsiz) Öğrenme.....	41
2.3.1.7.	Yarı Gözetimli (Takviyeli (Reinforcement)) Öğrenme.....	41
2.3.1.8.	YSA'nın Eğitilmesi	41
2.3.1.9.	Geri Yayılım Algoritması (Back-Propagation)	44
2.3.2.	Karar Ağaçları	47
2.3.3.	Rastgele Orman Sınıflandırıcısı.....	49
2.3.4.	Gradyan Artırma Karar Ağacı	50
2.3.5.	Naive Bayes Sınıflandırıcısı	50
2.3.6.	k-En Yakın Komşu Sınıflandırıcısı	52
2.3.7.	Genelleştirilmiş Lineer Modeller.....	53
2.3.8.	Otomatik Çok Katmanlı Algılayıcı.....	56
2.3.9.	Kural Türetme.....	57
2.3.9.1.	Hata Azalması İçin Tekrarlanan Artımlı Budama Algoritması	58
2.4.	Derin Öğrenme ve Semantik Segmentasyon	59
2.4.1.	Derin Öğrenme	60
2.4.1.1.	Konvolüsyonel Sinir Ağları (CNN).....	62

2.4.1.1.1.	Konvolüsyon Katmanı	64
2.4.1.1.2.	Havuzlama Katmanı	68
2.4.1.1.3.	Tamamen Bağlı Katman	69
2.4.1.1.4.	Aktivasyon Fonksiyonunun Bulunduğu Katman.....	69
2.4.1.1.5.	Düğüm Seyreltme (Dropout) Katmanı	70
2.4.1.1.6.	Yığın Normalizasyonu (Batch Normalization) Katmanı	72
2.4.1.1.7.	Sınıflandırma Katmanı.....	72
2.4.1.2.	Derin Öğrenmede Kullanılan Hiperparametreler	73
2.4.1.2.1.	Öğrenme Hızı (Learning Rate)	73
2.4.1.2.2.	Mini-Yığın (Mini Batch) Boyutu	73
2.4.1.2.3.	Epok (epoch) Sayısı	74
2.4.1.2.4.	Momentum.....	75
2.4.1.2.5.	Gizli Katman Sayısı ve Gizli Katmanlardaki Nöron Sayısı	75
2.4.1.2.6.	Ağırlık Azalımı (Weight Decay)	75
2.4.1.2.7.	Başlangıç Ağırlık Değerlerinin Belirlenmesi	76
2.4.2.	Semantik Segmentasyon.....	76
2.4.2.1.	U-Net Mimarisi.....	76
2.4.2.2.	SegNet Mimarisi.....	78
3.	BULGULAR VE İRDELEME	82
3.1.	Kullanılan Değerlendirme Ölçekleri.....	82
3.2.	k-Katmanlı Çapraz Doğrulama.....	85
3.3.	İleriye Doğru Seçim.....	87
3.4.	Doku Örneklerinin Sınıflandırılmasına İlişkin Sonuçlar	88
3.4.1.	ÖM-1 ile Eğitilmiş Sınıflandırıcıların Performans Sonuçları.....	88
3.4.2.	ÖM-2 ile Eğitilmiş Sınıflandırıcıların Performans Sonuçları.....	90
3.4.3.	ÖM-3 ile Eğitilmiş Sınıflandırıcıların Performans Sonuçları.....	92
3.5.	Test Görüntülerinin Segmentasyon ve Sınıflandırma Sonuçları	94
3.6.	Derin Öğrenme Yöntemlerinden Semantik Segmentasyon ile Test Görüntülerinin Sınıflandırılmasına İlişkin Performans Sonuçları	97
3.7.	Elde Edilen Sonuçların Literatürdeki Benzer Çalışmalarla Karşılaştırılması	102
4.	SONUÇLAR.....	106
5.	ÖNERİLER.....	109
6.	KAYNAKLAR	111

ÖZGEÇMİŞ

Doktora Tezi

ÖZET

YANIK GÖRÜNTÜLERİNİN ÇOK DEĞİŞKENLİ İSTATİSTİKSEL
YÖNTEMLER VE DERİN ÖĞRENME YAKLAŞIMLARI İLE ANALİZİ

Erdoğan KARAKULLUKÇU

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
İstatistik ve Bilgisayar Bilimleri Anabilim Dalı
Danışman: Dr. Öğr. Üyesi Uğur ŞEVİK
2020, 124 Sayfa

Yanık uzmanlarının nadiren bulunduğu özellikle kırsal kesim acil servislerinde, yanık yarasının fotoğraflarının çekilmesi ve görüntülerin yanık uzmanları tarafından incelenebileceği tam teşekküllü bir sağlık merkezine gönderilmesi ile hastaya yapılacak ilk müdahaleye karar verilebilmektedir. Sunulan tezin amacı, yanık hastalarına ilişkin 105 adet dijital (2D) görüntüdeki sağlıklı ve yanık deriye ilişkin bölgelerin tespit edilebilmesidir. Tezde bu amaç doğrultusunda, iki tip yaklaşım önerilmiştir. Segmentasyon ve sınıflandırma olmak üzere iki aşamadan oluşan ilk yaklaşımda, literatürde yaygın olarak kullanılan 4 segmentasyon ve 10 sınıflandırma yönteminin performansı, değişen renk uzaylarına ($CIE L^*a^*b^*$, HSV , $YCbCr$) ve sınıflandırıcı eğitiminde kullanılan öznitelik matrislerine (gri seviye histogram istatistikleri, Haralick öznitelikleri, ortalama parlaklık değerleri) göre değerlendirilmiştir. En iyi sınıflandırma performansı, $CIE L^*a^*b^*$ renk uzayının a^* ve b^* renk kanallarının FCM’de girdi verisi olarak kullanıldığı ve segmente edilmiş görüntüdeki her bir bölgenin ileriye doğru seçim sonrasında elde edilen 9 adet Haralick özneliği kullanılarak eğitilmiş yapay sinir ağı modeli ile sınıflandırıldığı durumda elde edilmiştir. 5-katmanlı çapraz doğrulamada hesaplanan ortalama F-skoru %74.28 olarak bulunmuştur. Sunulan tezde kullanılan ikinci yaklaşımda, U-Net ve SegNet mimarileri, iki çeşit eğitim seti kullanılarak eğitilmiş ve 64x64 piksel boyutundaki bloklar ile eğitilen SegNet modeli 5-katmanlı çapraz doğrulama ortamında %80.5’lik bir ortalama F-skoru elde etmiştir.

Anahtar Kelimeler: Yanık, Deri, Sınıflandırma, Segmentasyon, Derin öğrenme, Semantik segmentasyon, SegNet, U-Net

PhD. Thesis

SUMMARY

ANALYSIS OF BURN IMAGES BY MULTIVARIATE STATISTICAL
METHODS AND DEEP LEARNING APPROACHES

Erdirinç KARAKULLUKÇU

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Statistics and Computer Sciences Graduate Program
Supervisor: Asst. Prof. Uğur ŞEVİK
2020, 124 Pages

Especially in rural emergency departments where burn specialists are rarely present, in order to decide on the first intervention to the patient, photographs of the burn wound are taken and sent to a full-fledged health center where the images can be examined by burn specialists. The aim of this thesis is to determine the healthy and burned skin regions in 105 digital (2D) images of burn patients. Two types of approaches have been proposed for this purpose. In the first approach consisting of two stages, segmentation and classification, the performance of 4 segmentation and 10 classification methods commonly used in the literature was evaluated in terms of changing color spaces (*CIE L*a*b**, *HSV*, *YCbCr*) and the feature matrices (gray level histogram statistics, Haralick attributes, average pixel intensity values) used in the training phase of the classifiers. The best classification performance was achieved in the case when *a** and *b** color channels of the *CIE L*a*b** color space were used as the input data in the FCM, and each segmented region was classified by the ANN classifier trained with 9 Haralick features that were selected by forward selection. The average F-score calculated by 5-fold cross-validation was 74.28%. In the second approach used in the thesis, U-Net and SegNet architectures were trained using two kinds of training sets and the SegNet model, which was trained with 64x64 pixel image blocks, achieved an average F-score of 80.5% in a 5-fold cross-validation environment.

Key Words: Burn, Skin, Classification, Segmentation, Deep learning, Semantic segmentation, SegNet, U-Net

ŞEKİLLER DİZİNİ

Sayfa No

Şekil 1.	Tezin genel akış şeması	4
Şekil 2.	Derinin yapısı ve tabakaları	5
Şekil 3.	Deri anatomisine göre yanık derinlikleri [22]	9
Şekil 4.	Yetişkin, çocuk ve bebek yanık hastalarında 9'lar kuralı. Copyright 2003, Regents of the University of Michigan.	12
Şekil 5.	(a) Lund-Browder şemasına göre yüzey alanları [34] (b) Avuç içi kuralı	13
Şekil 6.	Deri/yanık işaretleme örneği: soldaki orijinal renkli resim; sağdaki işaretli resim (mavi: deri, kırmızı: yanık).....	19
Şekil 7.	Tezde kullanılan alt veri seti örnekleri	19
Şekil 8.	Çıkartılan doku örnekleri. (a) Deri doku örnekleri, (b) Yanık doku örnekleri, (c) Arka plan doku örnekleri	20
Şekil 9.	Semantik segmentasyonda kullanılacak doku örnekleri.....	20
Şekil 10.	$d = 1$ olduğunda farklı oryantasyonlardaki gri seviye birlikte görülme matrisinin hesaplanmasına ilişkin bir örnek (rastgele üretilmiş dört gri seviyeye sahip 5x5 çözünürlüğünde bir 2D gri-seviye görüntü kullanılmıştır). 23	
Şekil 11.	Haralick öznitelik çıkarma prosedürü. H. Ö., Haralick özniteliği için kısaltmadır	24
Şekil 12.	Sinir ağının biyolojik ve matematiksel gösterimi.....	35
Şekil 13.	Çok katmanlı bir yapay sinir ağının genel yapısı	38
Şekil 14.	Geri beslemeli YSA örneği	40
Şekil 15.	YSA'da kullanılan öğrenme kuralları.....	43
Şekil 16.	Yapay zekâ, makine öğrenmesi ve derin öğrenmenin zaman içerisindeki gelişimi	61
Şekil 17.	Üç kanallı bir görüntü ile üç kanallı bir filtrenin konvolüsyonuna ilişkin işlemler. Girdi görüntüsündeki 3x3'lük taralı alanlar ile o alanlara karşılık gelen filtrelerin konvolüsyonu sonucunda oluşan değerler de taralı şekilde gösterilmiştir. Girdi görüntüsüne dolgu eklenmemiş, adım uzunluğu 1 olarak alınmıştır.....	65
Şekil 18.	Yükseklik için 2, genişlik için 1 kademe sıfır dolgulama uygulandıktan sonra 2 piksellik adım uzunluğuna göre iki boyutlu konvolüsyon işleminin yapılışı. P_W ve P_H ile genişlik ve yükseklik için kullanılan sıfır dolgulama miktarları, S ile adım uzunluğu miktarı gösterilmektedir. Çıktıdaki taralı değerler, girdideki taralı alanlarla 3x3'lük filtrenin konvolüsyonu sonucunda hesaplanmıştır.....	66

Şekil 19.	Üç kanallı bir görüntü ile üç kanallı bir filtrenin konvolüsyonuna ilişkin işlemler. Girdi görüntüsündeki 3x3'lük taralı alanlar ile o alanlara karşılık gelen filtrelerin konvolüsyonu sonucunda oluşan değerler de taralı şekilde gösterilmiştir. Girdi görüntüsüne dolgu eklenmemiş, adım uzunluğu 1 olarak alınmıştır.....	67
Şekil 20.	Maksimum havuzlama işlemleri. Solda, (224x224x64) boyutlu girdi 2x2'lik filtreler ile adım uzunluğu 2 alınarak havuzlanıyor ve boyutu (112x112x64)'e düşüyor. Sağda, örnek bir 2 boyutlu matriste aynı koşullarda maksimum havuzlamanın nasıl sonuçlar verdiği gösteriliyor.....	68
Şekil 21.	Havuzlama katmanının düzleştirilmesinden sonra tamamen bağlı katmanlara bağlanması.....	69
Şekil 22.	Düğüm seyreltme sinir ağı modeli. Solda, standart sinir ağı; sağda, düğüm seyreltme sonrasında sinir ağı.....	70
Şekil 23.	U-Net mimarisi (en düşük çözünürlüğün 32x32 piksel olduğu durum için bir örnek). Her mavi kutu çok kanallı bir öznetelik haritasına karşılık gelmektedir. Kutuların üstünde kanal sayısı belirtilmektedir. x-y boyutu, kutunun sol alt kenarında verilmektedir. Beyaz kutular kopyalanan öznetelik haritalarını temsil etmektedir. Her bir oka ilişkin işlev, şekle ait açıklama göstergesinde belirtilmektedir [188].....	77
Şekil 24.	SegNet mimarisinin bir örneği. Tam bağlantılı katman içermediği için tamamen konvolüsyonel bir mimaridir. Bir kod çözücü, kendisine karşılık gelen kodlayıcıdan gelen havuzlama indislerini kullanarak girdisini yukarı örnekler ve seyrek bir öznetelik haritası/haritaları üretir. Ardından, bu haritayı eğitilebilir bir filtre bankası ile konvolüsyon işlemine sokar ve haritanın yoğunlaşmasını sağlar. Son kod çözücü çıktı öznetelik haritaları piksel bazında sınıflandırma yapılabilmesi için bir softmax sınıflandırıcısına iletilir [189].....	79
Şekil 25.	Maksimum havuzlama ve havuzlama indis değerlerine göre yukarı örnekleme yapılması.....	80
Şekil 26.	k-katmanlı çapraz doğrulama yöntemi	86
Şekil 27.	Segmentasyona dayalı sınıflandırma işleminin şematik gösterimi	96
Şekil 28.	Semantik segmentasyon işleminin şematik gösterimi	99
Şekil 29.	Sınıflandırılmış görüntülerin yöntem bazında karşılaştırılması	101
Şekil 30.	Yanlış sınıflandırmaya neden olabilecek doku örnekleri. Sütunlarda sırasıyla deri, yanık ve arka plan dokuları yer almaktadır.	103
Şekil 31.	Herpes zoster enfeksiyonu için örnek bir görüntü [196]. Copyright 2019, StatPearls Publishing	105

TABLULAR DİZİNİ

Sayfa No

Tablo 1. Yanıkların derecelerine göre sınıflandırılması [27–29].....	11
Tablo 2. Yanık şiddetinin sınıflandırılması [41; 42].....	14
Tablo 3. İstatistiksel histogram özneliklerinin listesi	21
Tablo 4. Haralick öznelikleri	25
Tablo 5. Biyolojik sinir sistemi elemanları ve yapay sinir sisteminde karşılıkları.....	36
Tablo 6. Geleneksel algoritmalar ile YSA'ların karşılaştırması [100]	37
Tablo 7. Geri yayılım algoritması adımları [123].....	46
Tablo 8. GLM için kanonik bağlar	55
Tablo 9. En çok kullanılan derin öğrenme tabanlı segmentasyon algoritmalarının özeti [173]	63
Tablo 10. Literatürde en yaygın kullanılan aktivasyon fonksiyonları	71
Tablo 11. Hata matrisi tablosu	82
Tablo 12. İkili sınıflandırma için kullanılan performans metrikleri (Tablo 11'de kullanılan notasyonlara göre) [95].....	83
Tablo 13. Tablo 12'deki ölçütlerin birçok sınıf için genelleştirilmesine dayanan çok sınıflı sınıflandırma için performans metrikleri [95].....	84
Tablo 14. ÖM-1 ile eğitilmiş sınıflandırıcıların performans sonuçlarının karşılaştırılması	89
Tablo 15. ÖM-1 ile eğitilmiş YSA modeli kullanılarak 10-katmanlı çapraz doğrulama sonucunda elde edilen hata matrisi	89
Tablo 16. İleriye doğru seçim algoritması sonrası elde edilen öznelikler	90
Tablo 17. ÖM-2 ile eğitilmiş YSA modeli kullanılarak 10-katmanlı çapraz doğrulama sonucunda elde edilen hata matrisi	91
Tablo 18. ÖM-1 ve ÖM-2 ile eğitilmiş YSA sınıflandırıcılarının performans sonuçlarının karşılaştırılması.....	91
Tablo 19. ÖM-1 ve ÖM-2'nin hesaplama sürelerinin 1024x768 boyutunda görüntüler üzerinde değişen küme sayılarına göre karşılaştırılması (baz alınarak).....	92
Tablo 20. ÖM-3 ile eğitilmiş sınıflandırıcıların performans sonuçlarının karşılaştırılması	93
Tablo 21. ÖM-3 ile eğitilmiş YSA modeli kullanılarak 10-katmanlı çapraz doğrulama sonucunda elde edilen hata matrisi	93

Tablo 22. YSA-1 ve YSA-2 sınıflandırıcılarının kullanılan segmentasyon yöntemine ve girdi verisinin türüne göre ortalama performans sonuçları	95
Tablo 23. Semantik segmentasyon sonrası ortalama sınıflandırma performans sonuçları.	98
Tablo 24. Semantik segmentasyon ve çok değişkenli istatistiksel yöntemler ile sınıflandırmanın performanslarının karşılaştırılması.....	100
Tablo 25. Önerilen yöntemlere ilişkin sınıflandırma performansının literatürdeki yöntemlerde elde edilen performanslarla karşılaştırılması.....	104



SEMBOLLER DİZİNİ

2D	: 2 Boyutlu (2 Dimensional)
3D	: 3 Boyutlu (3 Dimensional)
5D	: 5 Boyutlu (5 Dimensional)
ABA	: Amerikan Yanık Birliđi (American Burn Association)
Adaline	: Adaptif Lineer Nöron (Adaptive Linear Neuron)
ART	: Adaptif Rezonans Teorisi (Adaptive Resonance Theory)
AutoMLP	: Otomatik Çok Katmanlı Algılayıcı (Automatic Multi-Layer Perceptron)
BAP	: Bilimsel Arařtırmalar Projeleri
BM	: Beklenti Maksimizasyonu (Expectation Maximization)
CART	: Sınıflandırma ve Regresyon Ağaçları-Classification and Regression Trees)
CHAID	: Otomatik Ki-Kare Etkileşim Belirleme (Chi-Squared Automatic Interaction Detector)
CIE	: Uluslararası Aydınlatma Komisyonu (Commission Internationale de l'Eclairage)
CIFAR	: Kanada İleri Arařtırma Enstitüsü (Canadian Institute For Advanced Research)
CNN	: Konvolüsyonel Sinir Ağları (Convolutional Neural Networks)
CRF	: Koşullu Rastgele Alan (Conditional Random Field)
DCT	: Ayrık Kosinüs Dönüşümü (Discrete Cosinus Transform)
DKM	: Dietterich-Kearns-Mansour Kriteri
DÖ	: Derin Öğrenme
FCM	: Bulanık c-Ortalamalar (Fuzzy c-Means)
FCN	: Tamamen Konvolüsyonel Ağlar (Fully Convolutional Networks)
FN	: Yanlış Negatif (False Negative)
FP	: Yanlış Pozitif (False Positive)
GBT	: Gradyan Artırma Ağacı (Gradient Boosted Tree)
GLCM	: Gri Seviye Birlikte Görülme Matrisleri (Gray Level Co-Occurrence Matrix)
GLM	: Genelleştirilmiş Lineer Modeller (Generalized Linear Models)
GMM	: Gauss Karışım Modeli (Gaussian Mixture Model)
GPU	: Grafik İşleme Ünitesi (Graphics Processing Unit)

HÖ	: Haralick Özniteliği
HSV	: Renk Özü-Doygunluk-Parlaklık Değeri (Hue-Saturation-Value)
IREP	: Artımlı Azaltılmış Hata Budama (Incremental Reduced Error Pruning)
IRLS	: İteratif Olarak Yeniden Ağırlıklandırılmış En Küçük Kareler (Iteratively Reweighted Least Squares)
KA	: Karar Ağaçları (Decision Trees)
kNN	: k-En Yakın Komşu (k-Nearest Neighbor)
KT	: Kural Türetme (Rule Induction)
KTÜ	: Karadeniz Teknik Üniversitesi
LSTM	: Uzun-Kısa Süreli Hafıza Ağları (Long-Short Term Memory)
MAP	: En Büyük Sonsal Sınıflandırma Yöntemi (Maximum A Posteriori Classification)
MARS	: Çok Değişkenli Uyarlanabilir Regresyon Uzanımları (Multivariate Adaptive Regression Splines)
MDL	: Minimum Açıklama Uzunluğu (Minimum Description Length)
MLP	: Çok Katmanlı Algılayıcı (Multi-Layer Perceptron)
NB	: Naive Bayes
NiN	: Ağ İçinde Ağ (Network in Network)
OD	: Ortalama Duyarlılık
OF	: Ortalama F Skoru
OH	: Ortalama Hassasiyet
OOB	: Test Verisi (Out-of-Bag)
ÖM-1	: Öznitelik Matrisi 1
ÖM-2	: Öznitelik Matrisi 2
ÖM-3	: Öznitelik Matrisi 3
QUEST	: Hızlı-Yansız-Etkin İstatistiksel Ağaç (Quick, Unbiased, Efficient Statistical Tree)
RBM	: Sınırlı Boltzman Makineleri (Restricted Boltzmann Machines)
ReLU	: Düzleştirilmiş Lineer Birim (Rectified Linear Unit)
RGB	: Kırmızı-Yeşil-Mavi (Red-Green-Blue)
RIPPER	: Hata Azalması İçin Tekrarlanan Artımlı Budama (Repeated Incremental Pruning to Produce Error Reduction)
RNN	: Tekrarlayan Sinir Ağları (Recurrent Neural Network)
RO	: Rastgele Orman (Random Forest)
SGS	: Sıralı Geri Seçme

SİS	: Sıralı İleri Seçme
SLIC	: Basit Lineer İteratif Kümeleme (Simple Linear Iterative Clustering)
SLIQ	: QUEST Algoritmasında Gözetimli Öğrenme (Supervised Learning in QUEST)
SOM	: Öz-Düzenlemeli Harita Ağı (Self Organizing Map)
SPRINT	: Ölçeklenebilir, Parellenebilir Tümevarım Karar Ağacı (Scalable Parallelizable Induction of Decision Trees)
SVM	: Destek Vektör Makinesi (Support Vector Machine)
TN	: Doğru Negatif (True Negative)
TP	: Doğru Pozitif (True Positive)
TVYA	: Toplam Vücut Yüzey Alanı
TYYA	: Toplam Yanık Yüzey Alanı
VS-1,...,VS-5	: Veri Seti 1,...,Veri Seti 5
YSA	: Yapay Sinir Ağları
YSA-1	: ÖM-2 ile Eğitilmiş YSA Modeli
YSA-2	: ÖM-3 ile Eğitilmiş YSA Modeli

1. GENEL BİLGİLER

1.1. Giriş

Günümüzde hastalıkların teşhisinde kullanılan karar destek sistemleri, birçok tıp alanında olduğu gibi deri yanıklarının tespitinde de kullanılmaktadır. Dünya Sağlık Örgütü'nün [1] verilerine göre, yanık nedeniyle her yıl yaklaşık 180,000 ölüm meydana gelmektedir. Yanıkların büyük çoğunluğu düşük ve orta seviyeli gelire sahip ülkelerde gerçekleşmektedir. Ülkemizde, Sağlık Bakanlığı tarafından yanık üniteleri için çalışmalara 2006 yılında başlanmış, T.C. Sağlık Bakanlığı 2018 Faaliyet Raporu'na [2] göre 2009 yılında 3 olan yanık ünite/merkez sayısı, 2018 yılının sonu itibariyle 14 yanık merkezi ve 32 yanık ünitesine ulaşmıştır.

Yanık yaralanmalarının tedavisinde ilk adım acil servislerde atılmaktadır. Yanığın şiddetinin belirlenmesinde, hastaya verilecek sıvı miktarının hesaplanmasında, sevk ve yatış endikasyonunun belirlenmesinde kullanılan en önemli parametrelerden biri yanık yüzey alanıdır. Ülkemizde ve dünyada en çok kullanılan yanık yüzey alanı tespit metodu 9'lar kuralıdır. Ancak bu kuralın uygulanmasında, hekim değerlendirmeleri arasında %20'lere varan farklılıkların olduğu tespit edilmiştir. Bu fark, 80 kg ağırlığındaki bir hastanın sıvı tedavisinde 6400 ml mayi fazlalığı veya azlığı gibi tolerasyonu oldukça zor bir durum ortaya çıkarmaktadır. Benzer şekilde, bir yanık ünitesine %40 yanık alanı ile sevk edilen bir hastada, gerçekte %10-15 yanık yüzey alanı olduğunun tespit edilmesi sıklıkla rastlanılan bir durumdur. Bunun sonucunda, ağır yanığa sahip hastalar, yetersiz yataktan dolayı uzak mesafelere sevk edilme veya bekletilme gibi istenmeyen durumlarla karşılaşabilmektedir.

Yanık hastası için hayati öneme sahip olan sıvı resüsitasyonunun (vücutta kaybedilen sıvının yerine konulma şekli) belirlenebilmesi için hastanın vücut yanık yüzdesinin hesaplanması büyük önem taşımaktadır. Bu aşamada, yanık uzmanlarının nadiren bulunduğu veya hiç bulunmadığı özellikle kırsal kesim acil servislerinde, yanık hastasına yapılacak ilk müdahalede geç kalınabilmektedir. Bu sorun için uygulanan yaygın bir çözüm, yanık yarasının fotoğraflarının çekilmesi ve görüntülerin yanık uzmanları tarafından incelenebileceği tam teşekküllü bir sağlık merkezine gönderilmesidir. Görüntü kalitesindeki değişkenliklerin yanı sıra, yanık uzmanlarının erişilebilirliği ve mevcudiyeti göz önüne alındığında, daha hızlı ve güvenilir bir çözüme ihtiyaç olduğu açıktır.

1.2. Tezin Amacı ve Yapısı

Uluslararası alanda yapılan arařtırmalarda, yanık yüzdesinin hastanın gerçek 3D modeli üzerinden otomatik yöntemler kullanılarak hesaplandığı bir sisteme rastlanılamamıştır. KTÜ BAP FAY-2016-5588 Altyapı Projesi bünyesinde birden fazla stereo kameranın kullanıldığı özgün bir görüntüleme sistemi ile yanık hastasına özel 3D modellerin oluşturulması düşünülmüştür. Bu 3D modellerin üzerinde yanık tespiti yapılarak hastanın yanık yüzdesinin gerçeğe en yakın şekilde hesaplanması planlanmaktadır.

Sunulan tezin amacı, yanık hastalarına ilişkin dijital (2D) görüntülerdeki sağlıklı ve yanık deriye ilişkin bölgelerin birbirinden ayırt edilebilmesi için, çok değişkenli istatistiksel yöntemler ve semantik segmentasyon derin öğrenme algoritmalarının kullanılmasıdır.

Çalışmanın sonucu, üç ana süreç için ön hazırlık olacaktır. Bunlar, (i) yanıkların sınıflandırılması, (ii) yanık derecesinin belirlenmesi ve (iii) periyodik olarak hastalardan alınan dijital fotoğraflara dayanarak yanık iyileşme sürecinin seyri hakkında bilgi edinilmesi şeklindedir.

Tez kapsamında çözülecek problemler ve kullanılacak yöntemler maddeler halinde belirtilmiştir. Çok değişkenli istatistiksel yöntemlerin kullanımını aşamasında aşağıdaki işlemler gerçekleştirilmiştir:

- 1) Yanık Görüntüsünün Kümelere Ayrılması: Dijital yanık görüntülerini oluşturan pikseller; k-Ortalamlar (k-Means), Bulanık c-Ortalamlar (Fuzzy c-Means-FCM), Basit Lineer İteratif Kümeleme (Simple Linear Iterative Clustering-SLIC) ve Beklenti Maksimizasyonu (Expectation Maximization-BM) yöntemleri ile kümelere ayrılmıştır.
- 2) Yanık Görüntülerinden Öznitelik Çıkarma İşlemi: Veri tabanında yer alan her bir görüntüye ilişkin pikseller yanık uzmanı doktorlar tarafından deri, yanık veya arka plan olarak işaretlenerek görüntüye ilişkin etiketler oluşturulmuştur. Etiketlenmiş bölgelerin farklı renk uzaylarına ilişkin piksel değerleri kullanılarak Gri Seviye Birlikte Görülme Matrisleri (Gray Level Co-Occurrence Matrix-GLCM) oluşturulmuştur. Bu matrisler üzerinden Haralick öznitelikleri ve gri seviye histogram istatistikleri çıkarılmıştır.
- 3) Öznitelikler Kullanılarak Sınıflandırıcıların Eğitilmesi: Elde edilen tüm öznitelikler kullanılarak, 10 katmanlı çapraz doğrulama ile Yapay Sinir Ağları (YSA), Gradyan Artırma Ağacı (Gradient Boosted Tree-GBT), Rastgele Orman (Random Forest-

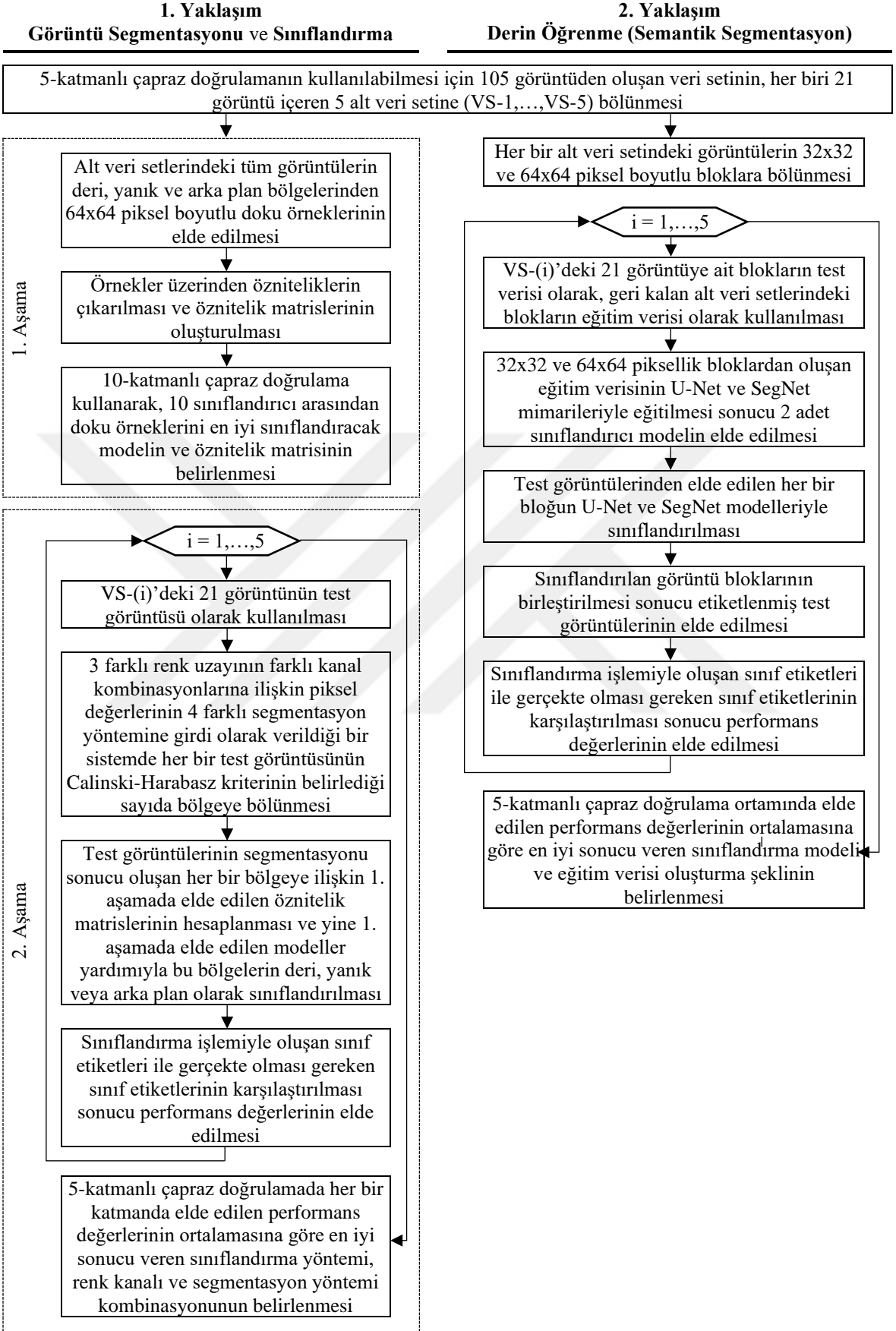
RO), Klasik Derin Öğrenme (DÖ), Genelleştirilmiş Linear Modeller (Generalized Linear Models-GLM), Otomatik Çok Katmanlı Algılayıcı (Automatic Multi-Layer Perceptron-AutoMLP), Kural Türetme (Rule Induction-KT), k-En Yakın Komşu (k-Nearest Neighbor-kNN), Karar Ağaçları (Decision Trees-KA) ve Naive Bayes (NB) sınıflandırıcılarının eğitilmesi sağlanmıştır.

- 4) Sınıflandırma Sonuçlarının Elde Edilmesi ve Öznitelik Seçimi: Eğitilen sınıflandırıcılar kullanılarak farklı segmentasyon yöntemlerine göre kümelere ayrılmış yanık görüntülerindeki her bir bölge deri, yanık veya arka plan olarak sınıflandırılmış ve uzmanlar tarafından oluşturulan gerçek etiket değerleri ile karşılaştırma sonucunda sınıflara ilişkin performans değerleri elde edilmiştir. En yüksek başarıma sahip sınıflandırıcı kullanılarak öznitelik seçme işlemi yapılmış ve istenilen sınıfları en iyi temsil eden minimum sayıda öznitelik elde edilmiştir.

Derin öğrenme algoritmalarından semantik segmentasyonun kullanımı aşamasında aşağıdaki işlemler gerçekleştirilmiştir:

- 1) Veri Tabanı Üzerinde Yapılan İşlemler: Veri tabanındaki her bir görüntü ve bu görüntüye ilişkin etiket görüntüsü, 32x32 ve 64x64 piksellik parçalara ayrılarak semantik segmentasyonda kullanılmak üzere veri tabanının genişletilmesi sağlanmıştır.
- 2) U-Net ve SegNet Semantik Segmentasyon Yöntemlerinin Kullanılması: 5 katmanlı bir çapraz doğrulamanın kullanılabilmesi için veri seti rastgele olarak beş alt veri setine bölünmüştür. Oluşan 5 alt veri setinin 4'lü kombinasyonlarına ilişkin setlerde yer alan görüntüler kullanılarak U-Net ve SegNet yöntemleri ile eğitim işlemi gerçekleştirilmiştir. Her bir yöntem için 5'i 32x32'lik, 5'i ise 64x64'lük görüntülerle eğitilmiş modeller olmak üzere 10 adet model elde edilmiş ve bu modeller kullanılarak eğitime katılmayan alt veri setleri üzerinde sınıflandırma işlemleri yapılmıştır. Elde edilen performans değerlerine göre yanık görüntülerini deri, yanık ve arka plan olarak sınıflandıracak en iyi semantik segmentasyon seçeneğine karar verilmiştir.

Tezin birinci bölümünde, derinin yapısı, yanık türleri ve yanık dereceleri hakkında bilgiler verilmiş, yanık hastalarından elde edilmiş dijital görüntüler üzerinde hastaya ilişkin deri ve yanık bölgelerinin tespiti için kullanılacak öznitelikler, sınıflandırma ve kümeleme yöntemlerine ilişkin literatür özeti sunulmuştur.



Şekil 1. Tezin genel akış şeması

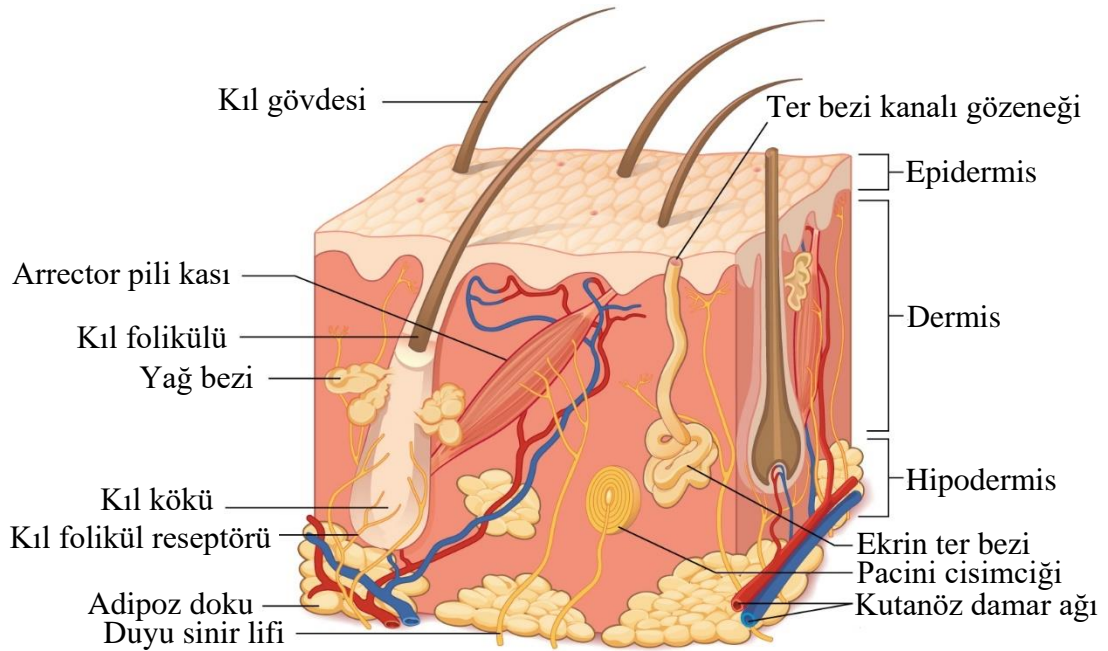
Tezin ikinci bölümünde, tezde yapılan çalışmalara yer verilmiştir. İlk olarak dijital yanık görüntülerinin deri, yanık ve arka plan olarak sınıflandırılması için önerilen çok değişkenli istatistiksel yöntemler ile sınıflandırma süreci ve bu süreçte kullanılan öznelik çıkartma, segmentasyon ve sınıflandırma yöntemleri anlatılmıştır. Daha sonra ise derin öğrenmeye ilişkin adımlar açıklanmış ve tezde kullanılan semantik segmentasyon yöntemleri ile ilgili bilgiler verilmiştir.

Tezin üçüncü bölümünde, tezde önerilen çok değişkenli istatistiksel yöntemler ile sınıflandırma ve derin öğrenme tabanlı semantik segmentasyon süreçlerinde kullanılan yöntemler literatürde kullanılan başarı metrikleri ile ölçülmüş ve bu yöntemlerin yanık görüntülerini sınıflandırmadaki başarımları karşılaştırılmıştır.

Tezin dördüncü ve beşinci bölümlerinde bu çalışmanın sonuçlarından ve gelecekteki çalışmalardan bahsedilmiştir. Tezin genel akış diyagramı Şekil 1’de görülmektedir.

1.3. Derinin Yapısı

Deri, vücudun dış yüzeyini kaplayan, hayati birçok fonksiyonu olan ayrıca çevre ile organizma arasındaki iletişimi sağlayan oldukça organize ve tabakalı bir organdır. Dıştan içe doğru epidermis, dermis ve hipodermis (subkutis) olmak üzere 3 temel tabakadan oluşmaktadır [3]. Şekil 2’de bu tabakalar gösterilmektedir.



Şekil 2. Derinin yapısı ve tabakaları

Epidermis, vücudu dış etkenlerden koruyan, cildin elastikiyetinin korunmasına yardımcı olan, vücudun sıvı ve elektrolit dengesinde rol oynayan derinin en dış tabakasıdır. Dıştan içe doğru sırasıyla; stratum korneum, stratum lusidum, stratum granulosum, stratum spinosum ve stratum germinativum olarak adlandırılan 5 ayrı alt tabakaya sahiptir [4].

Dermis, epiderminin altında ve subkutan tabakanın üzerinde yer alır ve epidermise besin ve fiziksel destek sağlamaktan sorumludur. Damar, sinir, duyu sinir sonlanmaları, ter bezleri ve kıl köklerini içermektedir [5]. Dermis embriyolojik açıdan mezoderm kökenli olup, papiller ve retiküler dermis olmak üzere iki alt katmandan oluşur. Papiller katman, epidermisi besleyen sinirleri ve kılcıl damarları içerirken; retiküler katman, kollajen ve elastik lifler içeren güçlü bir bağ dokusundan oluşmaktadır [6].

Hipodermis, yüzeysel dermise gevşek bir şekilde bağlı olan, fibröz bağ dokusundan oluşan bir tabakadır. Kalınlığı; yaş, cinsiyet, ırk, endokrin ve bireyin beslenme durumuna göre değişir. İzolasyon tabakası ve koruyucu bir yastık görevi görür ve vücut ağırlığının yaklaşık %10'unu oluşturur [7].

1.4. Yanık

Yanık; yüksek ısı, radyasyon ışınları, elektrik, yıldırım çarpması ve kimyasal maddelerle temas gibi nedenlerle oluşan, deri ve deri altı dokularda meydana gelen, vücudun dışı açılan organlarını hasara uğratabilen bir yaralanma şeklidir [8]. Yanığı oluşturan etken madde, etken maddenin sıcaklığı, etken madde ile deri arasındaki mesafe, yüzey alanı ve temas ettiği süreye göre doku yıkımı değişmektedir. Yanık, oluşumuna neden olan etkenin niteliğine göre, yanık alanının genişliği ve derinliğine göre sınıflandırılabilir [9].

1.5. Etkenin Niteliğine Göre Yanık Türleri

Yanık etkenleri; haşlanma ve alev yanıkları (termal), yıldırım ve elektrik çarpmasına bağlı yanıklar, inhalasyon yanıkları, kimyasal maddelerle temas sonucu oluşan yanıklar, radyasyona bağlı yanıklar gibi alt başlıklara ayrılabilir [10].

1.5.1. Termal Yanıklar

Termal yanıklar haşlanma, alev yanıkları ve temas yanıkları olarak üç ana başlığa ayrılabilir [11].

Çocuklarda yanıkların yaklaşık %70'ine haşlanma neden olmaktadır. Yaşlılarda da sıkça görülen haşlanma yanıklarının başlıca nedenleri sıcak içeceklerin ya da sıvıların vücuda dökülmesi veya sıcak banyo suyuna maruz kalmaktır. Haşlanmalar genellikle 1. ve 2. derece yanık derinliğine sahiptirler [11].

Yetişkinlerde görülen yanıkların yaklaşık %50'lik bir kısmını oluşturan alev yanıkları morbidite (hastalığa yakalanan sayısının sağlıklı kalmış nüfusa oranı) ve mortalite (ölüm) oranı yüksek olan, 2. derece derin ve 3. derece yanıklardır. Bu tip yanıklara genellikle inhalasyon (soluk alma) hasarı ve diğer travmalar eşlik eder [11].

Doğrudan temasta yanma, dokunulan nesnenin ya aşırı sıcak olması ya da temasın anormal şekilde uzun olduğu durumlarda gerçekleşmektedir. Özellikle çocuklarda ütü, soba gibi cisimlere dokunulması sonucu ortaya çıkan yanık türüdür. Genellikle bu tip yanıklar el bölgesini içeren yanıklardır. Genellikle 1. derece ve 2. derece yanık yaralanması sınıfındadırlar [11].

1.5.2. Elektrik Yanıkları

Elektrik yanıkları yüksek voltaj (1000 volt veya üzeri), düşük voltaj (1000 volttan az) veya elektrik arkı sonrası oluşan flaş yanıkları olarak sınıflandırılabilirler [12]. Şimşek de elektrik yanıklarına neden olabilir. Yüksek voltaj yaralanmalarında asıl hasar vücudun içinde meydana gelmiş olabilir ve bu yüzden yaralanmanın büyüklüğüne derinin tek başına değerlendirilmesi ile karar verilemez. Düşük veya yüksek voltaj ile temas kalpte anormal kalp kasılmalarına veya kalp krizine neden olabilir [13].

1.5.3. İnhalasyon Yanıkları

İnhalasyon yanığı, doğrudan yüksek sıcaklıkta soluma, karbonmonoksit soluma veya zararlı yanma maddelerini doğrudan soluma nedeniyle oluşur [14].

İnhalasyon yanığı üst solunum yollarını ilgilendiren termal hasar, alt hava yollarının kimyasal irritasyonu ve zehirli gazların inhalasyonuna bağlı sistemik zehirlenme olmak üzere üç bileşenden oluşur [15; 16].

İnhalasyon yanığı, yanık hastalarında mortalite için önemli bir risk faktörü olmakla kalmayıp aynı zamanda bu hastalarda morbidite açısından da önemli bir belirleyicidir [15; 16].

1.5.4. Kimyasal Yanıklar

Kimyasal yanıklar; asit, alkali, fosfor ve sayısı 25000'i geçen diğer kimyasal maddelerden herhangi biri ile temas sonucu oluşmaktadır. Tuvalet temizleyicilerinde bulunan sülfürik asit, çamaşır suyunda bulunan sodyum hipoklorit, boya silicilerde bulunan halojen hidrokarbonlar kimyasal ajanlardan en yaygın olanları arasındadır [12]. Korozif madde tamamen çıkıncaya kadar koagülasyon nekrozuna neden olmaya devam ettiği için kimyasal yanıklar derin olma eğilimindedir. Alkaliler daha derine nüfuz etme eğilimindedirler ve asitlerden daha kötü yanıklara neden olurlar [11].

Kimyasal yanık hasarının şiddetini belirleyen faktörler arasında ajanın gücü (konsantrasyon), tipi ve temas süresi, miktarı, penetrasyon (içine girme, sızma) özelliği ve etki mekanizması sayılabilir. Kimyasal yanıktan sonraki birkaç gün deri sağlam gözükebilir fakat yanık alanındaki hasarın boyutu belli değildir. Yanığın derecesi tespit edilene kadar deride tam kat yanığın olduğu varsayılır [17].

1.5.5. Radyasyon Yanıkları

Endüstriyel alanlarda ve laboratuvarlarda yaşanan kazalar, nükleer silahların kullanılması ve tedavi amaçlı kullanılan radyoaktif elementin yanlış kullanılan yüksek dozları sonucu iyonize olmuş radyasyona maruz kalınması ile ortaya çıkan yaralanmalardır [8].

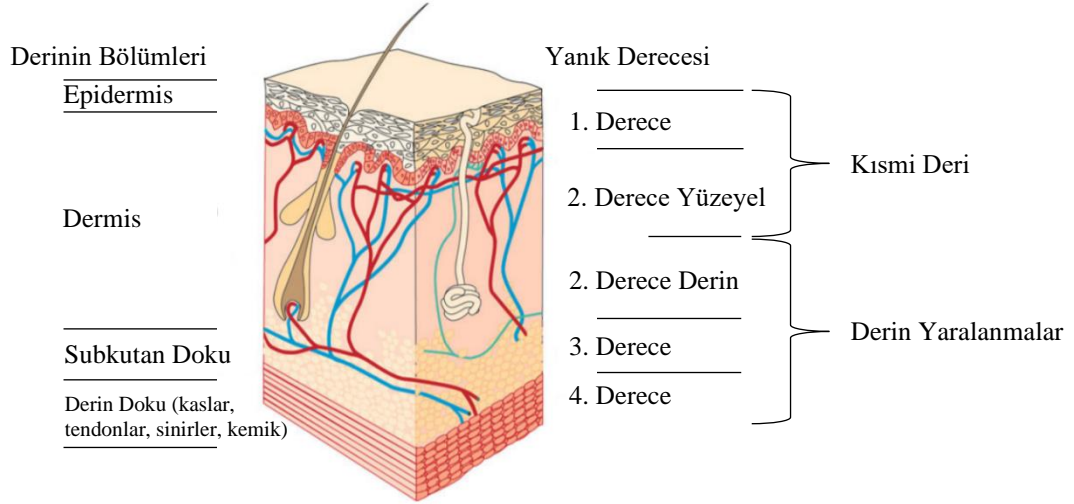
Radyasyon yanıkları, ultraviyole ışığa (güneş, bronzlaşma kabinleri veya ark kaynağı vb.) veya iyonlaştırıcı radyasyona (radyasyon terapisi, X ışınları, radyoaktif serpinti vb.) uzun süre maruz kalmaktan kaynaklanabilir. Bu tür yanıklar yaygın olmalarına rağmen nadiren ölümle sonuçlanır. Güneşten gelen ultraviyole ışınlarının ciltte meydana getirdiği yanıklar ve radyoterapi uygulanan hastalarda yüksek dozda kullanılan radyoaktif ışınların neden olduğu yanıklar radyasyon yanıklarının en yaygın örnekleridir [18].

1.6. Yanık Derinliğinin Sınıflandırılması

Yanık derecesi gelişen doku hasarına göre; birinci, ikinci, üçüncü ve dördüncü derece yanıklar olmak üzere Şekil 3'te gösterildiği gibi 4 sınıfa ayrılır. Yanık derecesini belirlemek için en sık kullanılan yöntem hekimin klinik gözlem ve değerlendirmesidir. Yanık alanının doku derinliğinin belirlenmesinde biyopsi ve histolojik analiz, altın standart olmasına karşın,

yeteri kadar pratik olmadığı için kullanılmayan bir yöntemdir. Son dönemlerde doku hasarı derecesini değerlendirmek için geliştirilen vital (canlıya enjekte edildiği zaman ölüme yol açmayan) boyaların kullanımı, lazer doppler görüntüleme ve termal görüntüleme gibi yöntemlerin klinikte kullanımı sınırlıdır. Bütün bu nedenlerle yanık derecesini değerlendirmek için genellikle yanık alanına ait klinik bulgular kullanılmaktadır [19; 20].

Yanıkların sınıflandırılmasındaki amaç, yaranın 14-21 gün arasında kendiliğinden iyileşebilir durumda mı yoksa deri kesimi veya kesilen bölgeye deri nakli gibi işlemlere ihtiyaç duyar durumda mı olduğuna karar verebilmektir. Yanlış bir pozitif değerlendirme gereksiz bir cerrahi müdahalenin yapılmasına yol açmaktadır. Yanlış bir negatif değerlendirmede ise hastanede kalma süresi artmakta, hasta kontraktür (kasın sürekli kasılma halinde oluşu) ve yara formasyonunda aşırı büyüme (hypertrophic scar formation) gibi sorunlarla yüzleşmektedir [21].



Şekil 3. Deri anatomisine göre yanık derinlikleri [22]

1.6.1. Birinci Derece Yanıklar

En az hasarlı yanık olan birinci derece yanıklar küçük ve yüzeysel yanıklardır, derinin bariyer fonksiyonlarında çok az kayıp vardır. Hasar sadece epidermiste sınırlıdır. Bu yanıklar ağrılıdır ve eritem (kılcal damarlarda kan toplanması sonucunda derinin kızarması) vardır. Dokununca yumuşaktır ve epidermal bariyer sağlamdır. İyileşme stratum germinativum tabakasındaki bazal hücreler vasıtasıyla olur ve 5-10 gün içinde yeni bir

epidermis oluşur. Olgunlaşması ise biraz daha zaman alır. Güneş yanıkları genellikle birinci derece yanıklardır [23].

1.6.2. İkinci Derece Yanıklar

İkinci derece yanıklarda doku hasarı epidermiste sınırlı kalmayıp dermise doğru uzanır. Bu yanık derecesinde doku hasarı papiller dermisi geçmiyorsa “yüzeyel ikinci derece”, retiküler dermise kadar uzanıyorsa “derin ikinci derece” olarak sınıflandırılır [24].

Yüzeyel ikinci derece yanıklarda, epidermis ve dermisen yüzeyel kısmı hasar görür. Yara, kırmızı görünümüne sahiptir ve yüzeyinde kabarcık (bül) oluşur. Yara dokusu yumuşak, nemli ve ağrılı olabilir [23]. Yüzeyel yanıkların çoğu 3 haftadan daha kısa sürede kendiliğinden iyileşir ve bunu tipik olarak fonksiyonel bozulma veya hipertrofik skar (yara tamir sürecinin beklenenin ötesinde aşırı iz oluşumu ile sonuçlanması) olmadan yapar [25].

Derin ikinci derece yanıklarda, epidermis ve papiller dermisle beraber yanık hasarı retiküler dermise kadar ilerler. Yanıklar daha soluk ve benekli görünür, dokunulmazsa ağrımaz, çimdiklemede acı hissedilir. Bu yanıklarda iyileşmeler, kıl folikülü ve ter bezlerindeki keratinositlerin re-epitelizasyonu (epitel hücrelerinin yeniden oluşması) ile gerçekleşir. İyileşme süresi 2 ile 5 hafta arası olmakla beraber genellikle dermis kaybının bir sonucu olarak ciddi skarlaşma (yara izi) görülür [26].

1.6.3. Üçüncü Derece Yanıklar

Bu yanıklar derinin tüm tabakalarının canlılığını kaybettiği tam kalınlıkta yanıklardır. Yanıklar ağrısız olup, siyah, beyaz veya kiraz kırmızısı renkte olabilirler. Yanık alan oldukça serttir ve eskar (canlılığını yitirmiş ölü doku) oluşumu görülür. Bu yanıklarda stratum germinativum, kıl folikülü ve ter bezlerindeki keratinositler canlılığını kaybettiğinden re-epitelizasyon genellikle mümkün olmaz. Derin ikinci derece ve üçüncü derece yanıklarda tedavi, eskar eksizyonu (yanık derinin kesilerek çıkarılması) ve greftleme (vücuttaki herhangi bir dokunun, vücuttaki başka bir bölgeye nakil edilmek amacıyla, bu dokuyu besleyen damar ve sinir bağlantısı olmadan alınması) gibi cerrahi işlemler ile sağlanır [23; 26].

Tablo 1. Yanıkların derecelerine göre sınıflandırılması [27–29]

Derece/Derinlik	Etkilenen Dokular	Klinik Görünüm	Bül	Ağrı	Ortalama İyileşme Süresi	Skar
Birinci Derece	Epiderminin bir kısmı	Kırmızı, parlak	Yok	Ağrılı	3-6 gün	Tam iyileşir, iz bırakmaz
İkinci Derece Yüzeysel	Epidermis ve yüzeysel dermis	Nemli, açık pembe	Var	Çok ağrılı	7-21 gün	Orta derecede iyileşme, pigmental değişiklik
İkinci Derece Derin	Epidermis ve derminin tamamı	Kuru, yamalı kırmızı renk, beyaz	Nadir	Az veya yok	21 günden fazla	Ciddi skar
Üçüncü Derece	Derinin tüm katları	Kuru kösele gibi, kirli beyaz, kahverengi, siyah	Yok	Yok	Aylarca	Ciddi iz bırakır, hipertrofik skar ve kontraktür

1.6.4. Dördüncü Derece Yanıklar

Dördüncü derece yanık; epidermis, dermis ve deri altı dokuların yanı sıra adipoz dokuların, kasların, tendonların ve hatta kemik dokuların tamamen tahrip edildiği yanmayı ifade eder. Dördüncü derece yanık, elektriğe, sıcak sıvıya veya alevli yanmaya uzun süre maruz kalındığında ortaya çıkabilir. Yanık bölgesinde kahverengi veya siyah renk değişimi gerçekleşir. Yara dokusu oldukça sert ve kurudur; ağrı olmamakla beraber dokuda his kaybı mevcuttur. Geniş ve kapsamlı cerrahi girişim gerektirir. Nekrotik (ölü) dokular debride edildikten (cerrahi yöntemler ile alındıktan) sonra sıklıkla flep (kendi kan dolaşımı olan, bir damardan kan akışı olan ve birçok doku katmanından oluşan doku parçası) ile kapatma gerektirir. Bazen amputasyon (uzuv kesilmesi) gerekebilir [14; 24].

Yanıkların derecelerine göre sınıflandırılmasına ilişkin bilgiler Tablo 1’de verilmiştir.

1.7. Yanık Yüzey Alanı

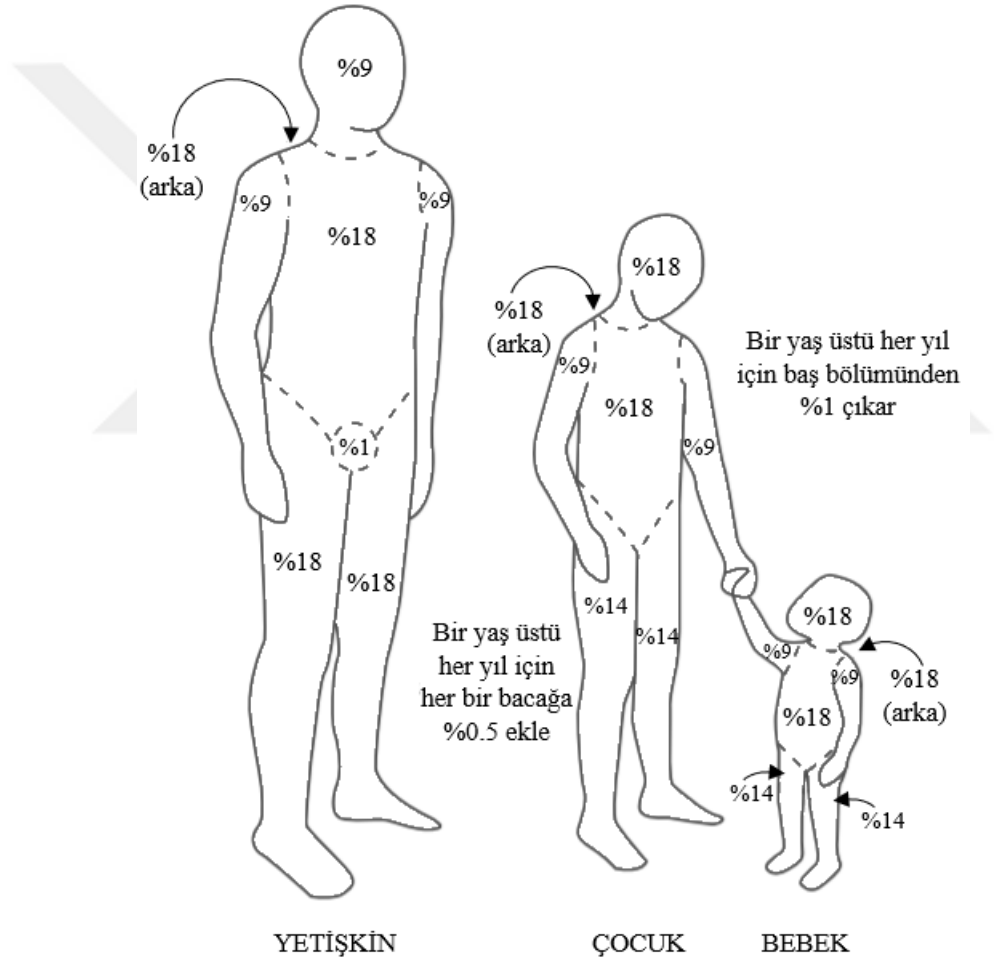
Toplam yanık yüzey alanı (TYYA) hastaya yapılacak olan sıvı resüsitasyonunun belirlenmesi ve hastaların ayaktan mı yoksa hastaneye yatırılarak mı tedavi edileceğine karar verilmesinde büyük önem arz etmektedir.

Toplam vücut yüzey alanı (TVYA) üzerindeki yanmış bölgeler dikkate alınarak yapılan yanık yüzdesi hesaplanmalarına eritemli bölgeler dahil edilmemelidir. Bu durum, TVYA odaklı yanık değerlendirilmelerinde yapılan yaygın bir hatadır. Eritemin yanık hesaplamalarına dahil edilmesi, yanık alanın gerçekte olandan daha fazla tahmin edilmesine,

hastaya aşırı sıvı resüsitasyonu yapılmasına ve uzman yanık ünitelerine yanlış yönlendirmelerin yapılmasına neden olabilmektedir. Dolayısıyla TVYA odaklı yanık hesaplamalarına sadece epitelize olmayan alanlar dahil edilmelidir [30].

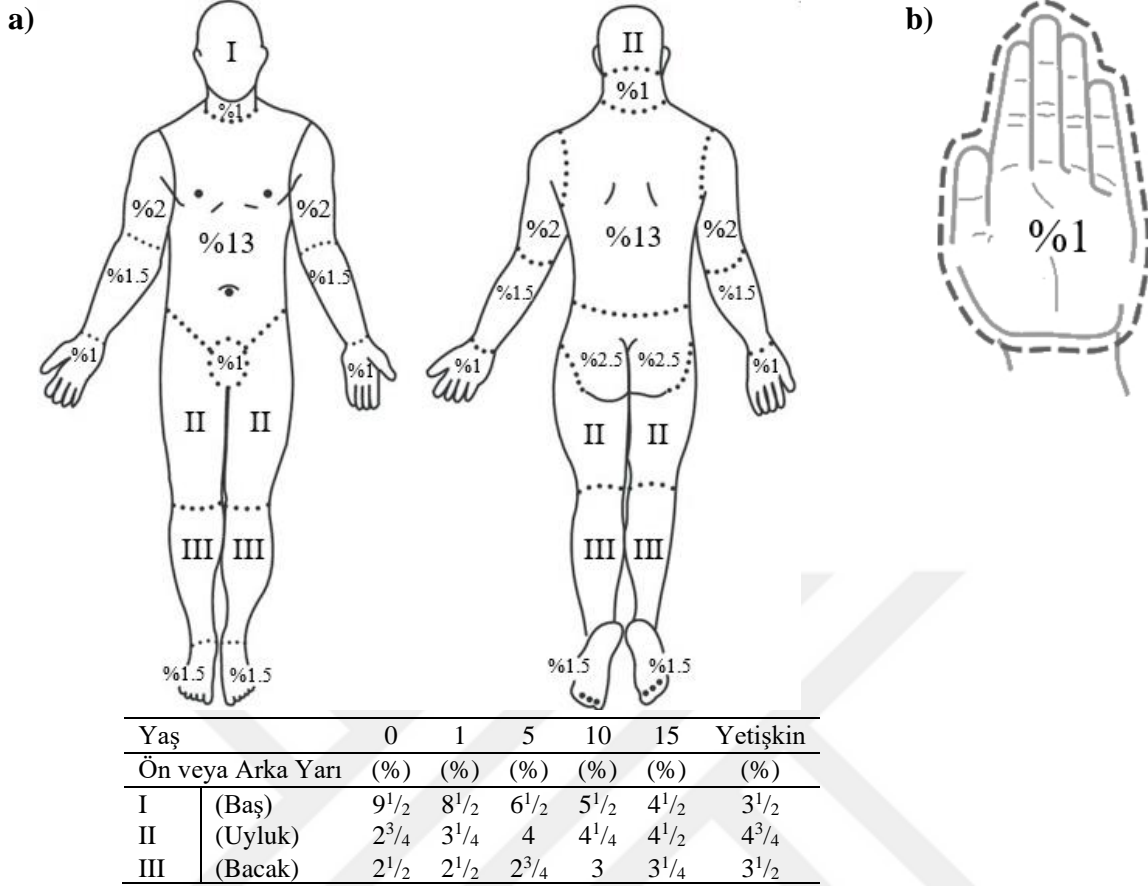
TVYA odaklı yanık değerlendirilmelerinde yaygın olarak kullanılan üç yöntem vardır:

Şekil 4'te görülen Wallece'nin Dokuzlar Kuralı (Wallace's Rule of Nines) [31] daha çok yetişkinler için kullanılmakta olan hatırlaması kolay ve hızlı tahminde bulunmak için kullanılabilen bir yöntemdir. Bu yöntem, yanık alanını gerçekte olandan yaklaşık %3 oranında daha fazla tahmin etme eğiliminde olduğu belirtilmiştir [32].



Şekil 4. Yetişkin, çocuk ve bebek yanık hastalarında 9'lar kuralı. Copyright 2003, Regents of the University of Michigan.

Şekil 5.a'da görülen Lund-Browder Çizelgesi [33], hastanın yaşını ve vücut oranlarını göz önünde bulundurduğundan Dokuzlar Kuralı'ndan daha doğru sonuçlar verebilmektedir.



Şekil 5. (a) Lund-Browder şemasına göre yüzey alanları [34] (b) Avuç içi kuralı

Bu nedenle, bu yöntem başta çocuklar olmak üzere, tüm yaş gruplarındaki hastalar için daha kullanışlıdır. Ancak, hesaplamaları yapmak için çizelgenin kullanılması gerekmektedir [32].

Şekil 5.b'de görülen Avuç İçi Kuralı (Rule of Palm), hastanın avuç içi ve parmaklarının TVYA'nın %1'ine denk geldiğini öngören, hızlı bir tahminde bulunabilmek için kullanılan bir yöntemdir. Nispeten ham bir yöntem olmasına rağmen, küçük veya büyük yanıkların (TVYA'nın <%15'i veya >%85'i) tahmin edilmesinde faydalıdır. Büyük yanıklarda yanık yüzdesi, zarar görmemiş derinin alanının bu yöntem ile tahmin edilmesi ve bu değerin 100'den çıkarılması ile hızlıca hesaplanabilir [30].

Günümüzde TYVA'nın hesaplanmasında bilgisayar ortamında oluşturan üç boyutlu modeller de kullanılabilmektedir. Bu modelleri içeren en tanınmış yazılımlar BurnCase 3D [35–37], EPRI 3D Burn Vision [38], BAI (Yanık yüzey alanı tahmini için 3D dijital model) [39] ve BurnCalc [40] şeklindedir. İlk üç yazılım hastanın cinsiyet, boy, kilo gibi özelliklerine göre hastaya ilişkin model oluştururken, BurnCalc yazılımı diğer yazılımlardan farklı olarak hastaya özel modeli Microsoft X-box™'ye bir eklenti olarak ortaya çıkan Kinect™ kamerasını kullanarak oluşturmaktadır. Bu yazılımlarda hastaya ilişkin yanık

bölgeler, oluşturulan model üzerinde bilgisayar faresi yardımı ile işaretlenmekte ve yanık yüzey alanı hesaplamaları buna göre yapılmaktadır.

1.8. Yanığın Şiddeti

Amerikan Yanık Birliği (American Burn Association-ABA) yanığın şiddetini; yanık alan yüzey genişliği, yanık derinliği, yanık bölgesi ve yanık nedenlerine göre küçük (minor), orta (moderate) ve büyük (major) şiddette yanıklar olmak üzere üç sınıfa ayırmaktadır [41; 42]. Yanığın şiddeti belirlenirken dikkat edilen hususlar Tablo 2’de verilmektedir.

1.9. Dijital Görüntülerden Deri ve Yanık Alanı Tespitine İlişkin Literatürde Yapılan Çalışmalar

Dijital görüntülerde deri ve yanık alanı tespitine ilişkin çalışmaların çoğu segmentasyon ve sınıflandırma konuları üzerinde durmaktadırlar. Bu nedenle sunulan tezde bu konular üzerine geniş bir literatür taraması yapılmıştır. İlgili alandaki her bir çalışma, veri

Tablo 2. Yanık şiddetinin sınıflandırılması [41; 42]

Yanık Kategorisi	Yanık Şiddeti Belirlenirken Dikkat Edilen Hususlar
Küçük Yanıklar	<ul style="list-style-type: none"> Erişkinlerde TYYA’nın %15’in altında olduğu 2. derece yanıklar Çocuklarda TYYA’nın %10’un altında olduğu 2. derece yanıklar Erişkin veya çocuklarda; gözler, kulaklar, yüz ve genital bölge dışında kalan ve TYYA’nın %2’nin altında olduğu 3. derece yanıklar
Orta Yanıklar	<ul style="list-style-type: none"> Erişkinlerde TYYA’nın %15-25 arasında olduğu 2. derece yanıklar Çocuklarda TYYA’nın %10-20 arasında olduğu 2. derece yanıklar Erişkin veya çocuklarda; gözler, kulaklar, yüz ve genital bölge dışında kalan ve yanık alanı %2-10 ve daha az olan üçüncü derece yanıklar
Büyük Yanıklar	<ul style="list-style-type: none"> Erişkinlerde TYYA’nın %25’in üzerinde olduğu 2. derece yanıklar Çocuklarda TYYA’nın %20’nin üzerinde olduğu 2. derece yanıklar Erişkin veya çocuklarda TYYA’nın %10’un üzerinde olduğu 3. derece yanıklar Yanık yüzdesi ne olursa olsun göz, yüz, kulaklar, ayaklar ve genital bölgeyi etkileyen bütün yanıklar Elektrik yanıkları İnhalasyon yanıkları Yanığa eşlik eden kompleks yaralanmaların beraber olduğu yanıklar

TYYA: Toplam yanık yüzey alanı

tabanı ve görüntü bilgisi, sınıflandırılmış bölgeler, çıkarılan öznelikler, kullanılan segmentasyon ve sınıflandırma yöntemleri bakımından detaylı bir biçimde incelenmiştir.

Yapılan araştırmalarda, segmentasyon ve sınıflandırma performansının görüntünün elde ediliş şekline bağlı olduğu görülmüştür. Acha ve diğerleri [43] aydınlatma ve kamera kalibrasyon sorunları ile başa çıkabilmek için özel bir görüntü alım protokolü geliştirmişlerdir. [44–49] sınıflandırma ve segmentasyon analizlerine başlamadan önce bu protokolü kullanmıştır. Bu çalışmaların en büyük dezavantajı, sistemin başarı oranının bu belirtilen protokole dayalı olmasıdır. Başka bir deyişle, tanımlanmış kurallara uymayan bir teknisyen tarafından çekilen bir görüntü için sistemin doğru sonuçları vermesi zor olabilmektedir. Öte yandan, [50–53]'deki veri tabanları, internet gibi çeşitli kaynaklardan toplanan, hastanelerden kişisel olarak çekilen ve kitaplardan taranan görüntüleri içermekle birlikte, [54–59]'deki veri tabanları belli hastanelerden elde edilen görüntüleri içermektedir. Tüm bu makalelerde herhangi bir görüntü alma protokolü izlenmediğinden, görüntüler üzerinde yapılan ön işleme, aydınlatma sorunuyla baş etmede önemli bir rol oynamıştır.

Segmentasyon algoritmasının türü, makalelerde değişiklik gösteren bir diğer faktördür. [43–46] çalışmalarında parlaklık ve renk değişimindeki birim değişikliklerin tekdüze olarak algılanabileceği $CIE L^*u^*v^*$ renk uzayına dayalı manuel bir segmentasyon gerçekleştirmiştir. Bu makalelerde segmentasyon işlemi, yanık yarasında küçük bir bölgenin bilgisayar faresi yardımıyla seçimi ile başlamış ve görüntü ön işleme ile devam etmiştir. Görüntüyü tek bir kanala dönüştürdükten sonra eşikleme (thresholding) ve son işleme (post processing) sonrasında segmentasyon tamamlanmıştır. Bu tür segmentasyon şeklinin en büyük dezavantajı, birden fazla yanık derecesine sahip alan içeren görüntülerde fare yardımıyla yanlış olarak seçilen bölgenin istenmeyen segmentasyon sonuçları ile sonuçlanabilmesidir. Manuel seçim, tasarlanan yanık türüne göre dikkatlice yapılmalıdır. Alonso Betanzos ve diğerleri [60] ile Castro ve diğerleri [61] çalışmalarında bulanık segmentasyon yöntemlerini, yanık görüntüler üzerindeki kümeleme yetenekleri açısından karşılaştırmışlardır. Her iki çalışmada da bulanık c-ortalamlar [62], bulanık k-en yakın komşu [63] ve modifiye bulanık c-ortalamlar [64] yöntemleri kullanılırken, [61]'ta uzamsal robust bulanık c-ortalamlar [65] ve çekirdekli bulanık c-ortalamlar [66] algoritmaları da analiz edilmiştir. Wantanajittikul ve diğerleri [67] yeni bir segmentasyon algoritması geliştirmiştir. Bu çalışmada görüntü önce RGB uzayından $YCbCr$ uzayının Cr kanalına dönüştürülmüş ve sonra deri bölgesini (yanık yarası dahil) arka plandan ayırmak için FCM yöntemi uygulanmıştır. Bundan sonra, elde edilen cilt bölgesine bir Luv dönüşümü

uygulanmıştır. Elde edilen u ve v değerleri, yanık yara bölgesini sağlıklı deriden ayırmak için girdi değerleri olarak FCM kümeleme algoritmasına verilmiştir. Tam otomatik özellikli bu segmentasyon yöntemi, [43]'deki yöntemden daha avantajlı görünmektedir. Bununla birlikte, yöntemin performansı, büyük ölçüde görüntünün arka planının karmaşıklığına bağlıdır. Bu yöntemin cilt rengine benzer bir arka plana sahip görüntülerde etkili olmasını beklemek çok iyimser olacaktır. Badea ve diğerleri [54], belli özellikleri sağlayan görüntüdeki herhangi bir alanın deri olarak kabul edilebileceğini belirtmişlerdir. Bu özellikler: belli bir $RGB-Yuv$ eşliğinin geçilmesi; kızılötesi görüntüde $32^{\circ}C$ 'den fazla bir sıcaklığa sahip olunması, alanın yanmış olarak işaretlenmiş olmaması. Tran ve diğerlerinin [57] çalışmasında görüntülerdeki yanmamış bölgelerin elimine edilmesi için görüntüler 4:3 oranında standart bir boyuta çekilerek normalleştirilmiştir. Görüntüler daha sonra Acha ve diğerleri [45] tarafından verilen renk bilgisine dayalı segmentasyon yöntemi kullanılarak bölümlere ayrılmıştır. Bu yönüyle, bu yöntem yarı otomatik olarak kabul edilmektedir. Kuan ve diğerleri [59] 2017'de bir hibrid segmentasyon yöntemi önermişlerdir. Bu yazarlar, Saranya ve diğerlerinin [68] çalışmasını deri bölgelerini arka plandan ayırabilmek için kullanmışlardır. Ardından segmente edilmiş deri bölgesi RGB 'den $L^*a^*b^*$ renk uzayına dönüştürülmüştür. Daha sonra yanık yarayı görüntüdeki vücut bölümünden ayırmak için a^* bileşeni üzerinde Otsu'nun [69] eşik değeri yöntemi kullanılmıştır. Segmentasyon sonrası yanık yara bölgelerinde bulunan küçük boşlukları doldurarak bölünmüş yanık yara bölgelerini pürüzsüzleştirmek için son işleme (post-processing) uygulanmıştır.

Öznitelik çıkarımı görüntü sınıflandırmada en önemli adımdır. Bu nedenle, birçok makalede farklı renk ve/veya doku temelli öznitelikler önerilmiştir. Acha ve diğerlerinin [43–45] ve Serrano ve diğerlerinin [46] çalışmalarında çıkarılan öznitelikler; ortalama parlaklık (L^*), renk tonunun (hue) ortalaması (h), renk doygunluğunun (chroma) ortalaması (c), ışık şiddetinin (lightness) standart sapması, renk tonunun standart sapması, renk doygunluğunun standart sapması, u^* 'in ortalaması, v^* 'in ortalaması, u^* 'in standart sapması, v^* 'in standart sapması, ışık şiddetinin çarpıklığı, ışık şiddetinin basıklığı, u^* 'in çarpıklığı, u^* 'in basıklığı, v^* 'in çarpıklığı ve v^* 'in basıklığı şeklinde belirtilmiştir. Öznitelik seçme algoritmasının uygulanmasından sonra elde edilen öznitelikler şu şekilde bulunmuştur: ışık şiddeti, renk tonu, renk tonunun standart sapması, u^* 'in renk doygunluğu bileşeni, v^* bileşeninin standart sapması ve ışık şiddetinin çarpıklığı. [67]'nin çalışmalarında kullandığı öznitelikler; renk tonunun (h) ortalaması, renk tonunun standart sapması, kontrast ve homojenlik olarak belirlenmiştir. Deepak ve diğerlerinin [50] ve Suvarna ve diğerlerinin [51];

53] çalışmalarında çıkarılan öznitelikler; ortalama ve $L^*a^*b^*$ renk uzayının V1 renk doygunluğu düzleminin Ayrık Kosinüs Dönüşümü (Discrete Cosinus Transform-DCT) fonksiyonunun (2,1). katsayısıdır. Acha ve diğerleri [48] çok boyutlu ölçeklemenin üç eksenini tahmin edebilmek için 8 öznitelik çıkarmışlardır. Bu öznitelikler: $C_{a^*b^*}^*$ (renkliliği ölçmeye çalışan sayısal parametre), o_{RGB} (RGB renk uzayında yer alan aykırı değer sayısı), h (renk tonu), v_{h^*} (renk tonunun açısasal varyansı), sk_{a^*} (a^* 'in çarpıklığı), sk_{b^*} (b^* 'in çarpıklığı), $SK_{0.25}$ (çarpıklığın çeyrek sapması), k_{a^*} (a^* 'in basıklığı). Serrano ve diğerleri [49], [48]'ye ait 8 özniteliği Sıralı İleri Seçme (SİS) ve Sıralı Geri Seçme (SGS) işlemlerinde kullanmışlardır. $C_{a^*b^*}^*$ (renkliliği ölçmeye çalışan sayısal parametre), h (renk tonu), k_{a^*} (a^* 'in basıklığı) öznitelikleri iki seçme yönteminde de ortak olarak seçilirken, v_{h^*} (renk tonunun açısasal varyansı) ve sk_{b^*} (b^* 'in çarpıklığı) SİS ve SGS'de seçilmiş 4. özniteliklerdir. Tran ve diğerleri [57] kırmızı, yeşil, mavi ve gri renk kanallarına dayalı hızlı öznitelik çıkarımı üzerine odaklanmışlardır. Kuan ve diğerlerinin [59] çalışmasında çıkarılan renge dayalı öznitelikler ışık şiddetinin ortalaması, renk tonunun ortalaması, renk tonunun standart sapması, a^* bileşeninin standart sapması, b^* bileşeninin standart sapması ve ışık şiddetinin çarpıklığı iken; çıkarılan doku öznitelikleri kontrast, korelasyon, enerji, homojenlik, ortalama, entropi, pürüzsüzlük, basıklık, çarpıklık ve ters fark momentidir.

Segmente edilmiş bölgelerin sınıflandırılabilmesi için çeşitli sınıflandırma yöntemleri önerilmiştir. Alonso Betanzos ve diğerleri [60], segmente edilmiş bir bölgeyi sınıflandırabilmek için bu bölgeyi eldeki tüm örneklerle karşılaştırmış ve bölgeyi, kendisine en benzer olan örneğin sınıfı ile ilişkilendirmişlerdir. Acha ve diğerleri [43; 45] ve Serrano ve diğerleri [46] yanıkları üç farklı derinliğe sınıflandırabilmek için bir Fuzzy-ARTMAP sinir ağı oluştururken, [44]'te bu yönteme ek olarak Destek Vektör Makinesinden (Support Vector Machine-SVM) de yararlanmıştır. Wantanajittikul ve diğerlerinin [67] çalışmaları, SVM'yi ikinci ve üçüncü derece yanıkları sınıflandırmak için kullanan diğer bir yöntemdir. Bu yöntemin sonuçları Bayes ve k-en yakın komşu sınıflandırıcılarının sonuçları ile karşılaştırılmıştır. Deepak ve diğerleri [50] ve Suvarna ve diğerleri [51], k-en yakın komşu sınıflandırıcısı, SVM ve şablon eşleştirme metotlarını yanıkları yüzeysel, kısmi kalınlıkta ve tam kat yanık olarak sınıflandırabilmek için kullanmışlardır. [51]'den farklı olarak [52]'de aynı amaç için YSA kullanılmıştır. Acha ve diğerleri [48] yanıkları yüzeysel, derin ve tam kat şeklinde sınıflandırabilmek için k-en yakın komşu sınıflandırıcısını ve SVM'yi kullanmışlardır. Aynı yöntemler, yanıkların doku nakline (grafting) ihtiyaç duyup duymamasına göre iki sınıfa ayrılması için de kullanılmışlardır. Suvarna ve Venkategowda

[53] çalışmalarında k-en yakın komşu sınıflandırıcısını kullanırken, Serrano ve diğerleri [49] SVM'yi uygulamışlardır. Badea ve diğerleri [54] piksel bazlı deri/yanık alanı tespit etme yöntemlerini ele almışlardır. Uzman cerrahlar tarafından yanık olarak işaretlenen görüntülerin bölümlerinden 32x32 boyutlu örnekler alınmış, bu örnekler, derin öğrenmeli Konvolüsyonel Sinir Ağlarının (Convolutional Neural Networks-CNN) [70] eğitimi aşamasında kullanılmışlardır. Elde edilen sonuçlar, renk-doku modeli [67], renk tonu-doygunluk modeli [71] ve *RGB-Yuv* modelinin sonuçları ile karşılaştırılmıştır. Badea ve diğerlerinin [55] diğer bir çalışmasında, yazarlar deri-yanık ayrımı için LeNet mimarisini [72] ele almışlardır. Yazarlar, bu mimariyi ve Ağ İçinde Ağ (Network in Network-NiN) mimarisini [73], deri, hafif yanık, ciddi yanık sınıflandırması için de kullanmışlardır. Tran ve diğerlerinin [56] çalışmasında CNN, yanığın dört dereceye sınıflandırılması için kullanılmıştır. Yerel ikili model operatörü (local binary pattern operator), CNN modelinin girdisi olarak uygulanmıştır. Despo ve diğerleri [58], deri/yanık sınıflandırması için Tamamen Konvolüsyonel Ağlar (Fully Convolutional Networks-FCN) [74] Koşullu Rastgele Alan (Conditional Random Field-CRF) [75] ile uygulamışlardır. Aynı yöntemler, bölgenin yüzeysel, yüzeysel/kısmi derin kalınlıkta, tam kalınlıkta ve yanmamış olarak sınıflandırılmasında da kullanılmıştır. WEKA [76] yazılımının yardımıyla, Kuan ve diğerleri [59] yanıkları üç derinlikte sınıflandırabilmek için lojistik regresyon, rastgele orman gibi 20 farklı sınıflandırma algoritmasını kullanmışlardır. Sabeena ve Kumar [77], görüntü bazlı gözetimli öğrenme yaklaşımı ve çok ölçekli bir süper piksel bilgisayar tabanlı SVM sınıflamasına dayanan bir modeli paralel yayılımla birlikte önermişlerdir.

Literatürde yer alan yöntemlerin yanığa ilişkin dijital görüntüleri sınıflandırmadaki performanslarına ilişkin bilgiler Bölüm 3'te verilmiştir.

1.10. Yanık Görüntü Veri Tabanı

Bu tezde kullanılan görüntüler KTÜ Farabi Hastanesi Yanık Ünitesi'nden temin edilmiştir. Görüntü alımı, dijital kameralar ve cep telefonları kullanılarak herhangi bir protokol takip edilmeden tıbbi hemşireler veya teknisyenler tarafından hastane koşullarında gerçekleştirilmiştir. Veri tabanı, yaşları geniş bir yelpazeyi kapsayan yanık hastalarının toplam 105 görüntüsünü içermektedir. Görüntüler farklı çözünürlüklerde olmakla beraber etik değerlendirmeler nedeniyle bazı görüntüler kırılarak 1024x768 piksel boyutuna getirilmiştir. Bu işlem sırasında herhangi bir ölçeklendirme işlemi uygulanmamıştır.

Her görüntünün deri ve yanık pikselleri uzman doktorlar tarafından manuel olarak işaretlenmiştir. Dolayısıyla, veri tabanında yer alan 105 görüntüye karşılık 105 adet de etiketli görüntü bulunmaktadır. Etiketlendirme sırasında, görüntülerde gölgede kalan veya çok fazla parlayan alanlar da bulunduğu bölgeye göre deri veya yanık olarak düşünülmüştür. İşaretli yanık görüntüsünün bir örneği Şekil 6’da gösterilmektedir.



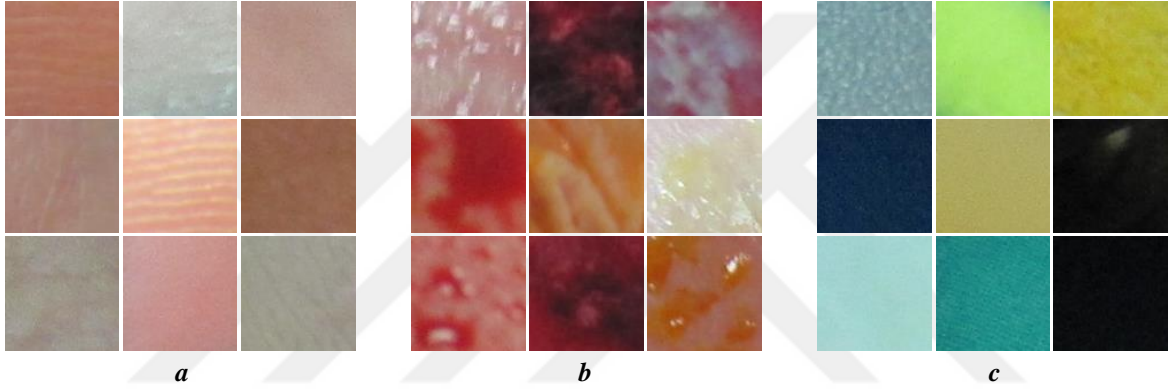
Şekil 6. Deri/yanık işaretleme örneği: soldaki orijinal renkli resim; sağdaki işaretli resim (mavi: deri, kırmızı: yanık)



Şekil 7. Tezde kullanılan alt veri seti örnekleri

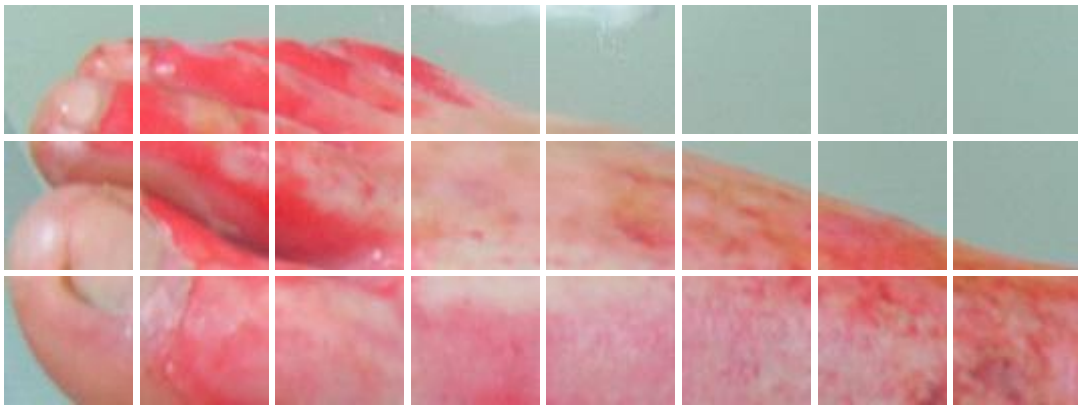
Tezde, performans sonuçlarının geçerliliğini artırmak için beş katmanlı bir çapraz doğrulama tasarımı geliştirilmiştir. Yüz beş görüntü rastgele olarak Veri Seti 1 (VS-1), VS-2, VS-3, VS-4, VS-5 olarak adlandırılan beş alt veri kümesine ayrılmıştır. Her bir veri seti Şekil 7’de görüldüğü gibi 21 görüntüden oluşmuştur.

Segmentasyon tabanlı sınıflandırma işleminde sınıflandırıcıların eğitimi aşamasında kullanılmak üzere, alt veri setlerindeki tüm görüntülerin deri, yanık ve arka plan bölgelerinden 64x64 piksel boyutlu doku örnekleri Şekil 8.a-c’dekine benzer şekilde çıkarılmıştır. İlgili veri setini temsil etmesi amacıyla, her bir alt veri setinden 450’şer adet deri, yanık ve arka plan örneği rastgele olarak seçilmiştir.



Şekil 8. Çıkarılan doku örnekleri. (a) Deri doku örnekleri, (b) Yanık doku örnekleri, (c) Arka plan doku örnekleri

Ayrıca, semantik segmentasyon işleminde, her bir yanık görüntüsü 32x32 ve 64x64 piksel boyutlu bloklara Şekil 9’daki gibi bölünmüş, SegNet ve U-Net mimarisinin eğitim basamağında ve görüntülerin test edilmesi işleminde bu bloklar kullanılmıştır.



Şekil 9. Semantik segmentasyonda kullanılacak doku örnekleri

2. YAPILAN ÇALIŞMALAR

2.1. Yanık Görüntülerinden Öznitelik Çıkarma İşlemi

Görüntü dokusu, bir görüntünün renk veya yoğunluğunun mekânsal düzlemdeki düzeni hakkında bilgi verir [78]. Deri ve yanıkların kendine özgü karakteristik özelliklere sahip oldukları bilinmektedir. Görüntülerdeki deri ve yanık olarak işaretlenen alanlardan çıkarılacak olan öznitelikler sınıflandırma aşamasında büyük yarar sağlayacaktır. Bu amaçla, her bir alt veri setinden 1350 (her sınıftan 450 tane) doku (toplamda 6750) örneklenmiştir ve bu dokuların her biri için, gri seviye histogram istatistiklerini [79], Haralick [80] istatistiklerini ve renk kanallarına ilişkin ortalama parlaklık istatistiklerini içeren öznitelikler çıkarılmıştır.

2.1.1. Gri Seviye Histogram İstatistikleri

Görüntüdeki bir bölgenin yoğunluk dağılımı hakkında faydalı bilgiler, ilgili bölgenin istatistiksel özelliklerine bakılarak elde edilebilir. Subashini ve diğerleri [79] gri seviye

Tablo 3. İstatistiksel histogram özniteliklerinin listesi

Ortalama: Ortalama piksel yoğunluğu $\mu = \frac{\sum_{ij} x_{ij}}{N}$	Tekdüzelik: Histogramın tekdüzeliği $U = \sum_{i=0}^{L-1} p(i)^2$
Standart Sapma: İlgilenilen bölgenin standart sapması $\sigma = \sqrt{\frac{\sum_{ij} (x_{ij} - \mu)^2}{N}}$	Basıklık: Histogram dağılımının düzgünlüğünü belirlemek için bir ölçüm $K = \frac{\sum_{ij} (x_{ij} - \mu)^4}{(N - 1)\sigma^4}$
Pürüzsüzlük: İlgilenilen bölgenin pürüzsüzlüğü $R = 1 - \frac{1}{(1 + \sigma^2)}$	Modifiye Standart Sapma: Ortalama kontrast $\sigma_m = \sqrt{\sum_{ij} (x_{ij} - \mu)^2 P(x_{ij})}$
Çarpıklık: Ortalama piksel yoğunluğu değerinin etrafında simetri ölçüsü $S = \frac{\sum_{ij} (x_{ij} - \mu)^3}{N\sigma^3}$	

x_{ij} : (i, j) konumundaki gri seviye piksel değeri; N : görüntüdeki piksel sayısı, L : gri seviyelerin sayısı, $p(i)$: histogramdaki yoğunluk seviyesi i 'nin olasılığı ve $P(x_{ij})$: x_{ij} 'deki yoğunluk seviyesinin olasılığı

histogramdan çıkarılan Tablo 3'teki dokuz istatistiksel özelliği destek vektör makinesinin eğitimi için kullanmış ve meme dokusunu üç farklı sınıfa ayırmışlardır. Benzer mantıkla, bu çalışmada, deri, yanık ve arka plan dokularından aynı özniteliklerin 7 tanesi çıkarılmıştır. Bu amaçla, *RGB* renk uzayının *R*, *G*, *B* kanalları ve *CIE L*a*b** [81] renk uzayının *L**, *a**, *b** kanalları kullanılmıştır. *RGB* renk uzayının gri kanalının da eklenmesiyle, her doku için toplam 49 öznitelik çıkarılmıştır.

2.1.2. Haralick Doku Öznitelikleri

Haralick ve diğerleri [80], çeşitli doku özniteliklerini çıkarmak için gri seviye birlikte görülme matrisi (Gray-Level Co-occurrence Matrix-GLCM) kullanımını başlatmıştır. GLCM, görüntüdeki komşu piksellerin göreceli konumu hakkında bilgi vermek için satırlar ve sütunların gri seviye yoğunluk değerlerini temsil ettiği bir kare matristir. GLCM, gri seviye değeri '*i*' olan bir pikselin, yatay (0°), dikey (90°) veya çapraz (45° veya 135°) konumdaki bitişik bir '*j*' pikseliyle ne sıklıkta birlikte görüldüğünü gösterir (*i* ve *j* görüntüdeki gri seviye değerleridir). Bir görüntünün GLCM'si, yarıçapı *d* ve yönü θ ile tanımlanan bir yer değiştirme vektörü (*t*) kullanılarak hesaplanır. GLCM matrisi Eşitlik 1'deki formül kullanılarak bulunur.

$$C_{\Delta x, \Delta y}(i, j) = \sum_{p=1}^m \sum_{q=1}^n \begin{cases} 1, & I(p, q) = i \text{ ve } I(p + \Delta x, q + \Delta y) = j \\ 0, & \text{diğer hallerde} \end{cases} \quad (1)$$

Eşitlik 1'de $I(p, q)$ görüntü fonksiyonudur, *m* ve *n* resim genişliği ve yüksekliğidir, Δx ve Δy , *d* ve θ ile tanımlanan uzaklık parametreleridir. Şekil 10, $d = 1$ olduğunda, farklı oryantasyonlardaki gri seviye birlikte görülme matrisinin hesaplanmasına ilişkin bir örneği göstermektedir.

GLCM hesaplandıktan sonra, matrisin her bir elemanı Eşitlik 2'deki formül kullanılarak normalize edilir.

$$p_{ij} = \frac{C(i, j)}{\sum_{k=1}^{N_g} \sum_{l=1}^{N_g} C(k, l)} \quad (2)$$

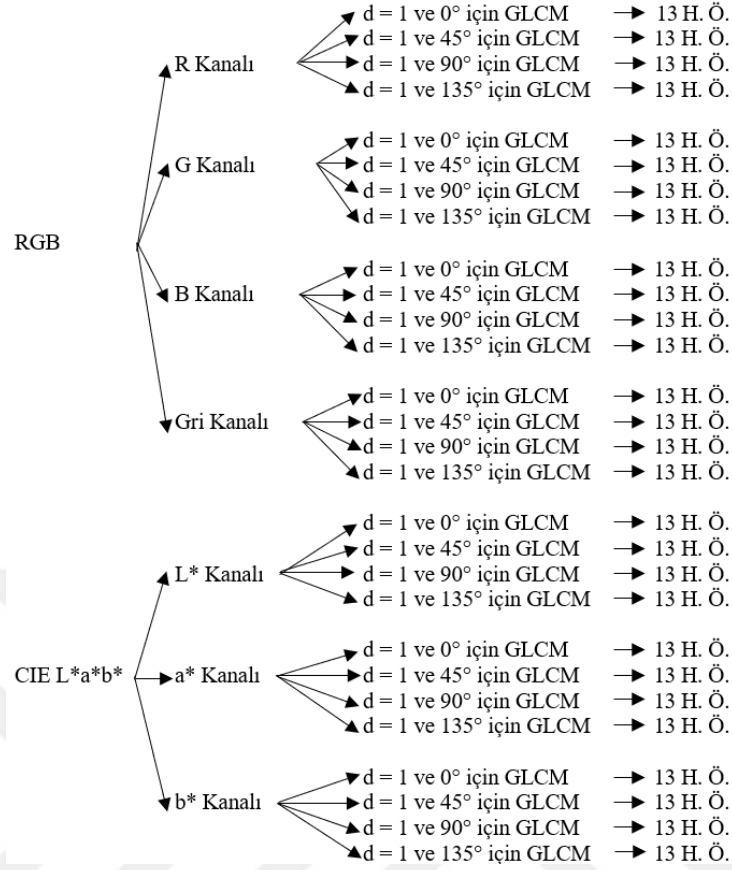
Eşitlik 2'de N_g görüntüdeki gri seviyelerin sayısını belirtmektedir. p_{ij} , normalize edilmiş GLCM'nin (*i, j*). elemanıdır.

Oryantasyon ($0^\circ, 45^\circ, 90^\circ, 135^\circ$)	4 gri seviyeye sahip orijinal görüntü (5×5 çözünürlüğünde)	$d = 1$ için GLCM																				
		<table border="1"> <thead> <tr> <th>1</th> <th>2</th> <th>3</th> <th>4</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>3</td> <td>0</td> <td>0</td> </tr> <tr> <td>2</td> <td>2</td> <td>0</td> <td>0</td> </tr> <tr> <td>3</td> <td>0</td> <td>0</td> <td>3</td> </tr> <tr> <td>4</td> <td>3</td> <td>0</td> <td>1</td> </tr> </tbody> </table>	1	2	3	4	1	3	0	0	2	2	0	0	3	0	0	3	4	3	0	1
1	2	3	4																			
1	3	0	0																			
2	2	0	0																			
3	0	0	3																			
4	3	0	1																			
		<table border="1"> <thead> <tr> <th>1</th> <th>2</th> <th>3</th> <th>4</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>2</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>3</td> <td>2</td> <td>0</td> <td>2</td> </tr> <tr> <td>4</td> <td>4</td> <td>0</td> <td>0</td> </tr> </tbody> </table>	1	2	3	4	1	0	0	1	2	1	0	0	3	2	0	2	4	4	0	0
1	2	3	4																			
1	0	0	1																			
2	1	0	0																			
3	2	0	2																			
4	4	0	0																			
		<table border="1"> <thead> <tr> <th>1</th> <th>2</th> <th>3</th> <th>4</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>4</td> <td>0</td> <td>0</td> </tr> <tr> <td>2</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>3</td> <td>0</td> <td>1</td> <td>3</td> </tr> <tr> <td>4</td> <td>2</td> <td>0</td> <td>2</td> </tr> </tbody> </table>	1	2	3	4	1	4	0	0	2	0	1	0	3	0	1	3	4	2	0	2
1	2	3	4																			
1	4	0	0																			
2	0	1	0																			
3	0	1	3																			
4	2	0	2																			
		<table border="1"> <thead> <tr> <th>1</th> <th>2</th> <th>3</th> <th>4</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>3</td> <td>1</td> <td>0</td> </tr> <tr> <td>2</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>3</td> <td>0</td> <td>0</td> <td>2</td> </tr> <tr> <td>4</td> <td>1</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	1	2	3	4	1	3	1	0	2	0	0	0	3	0	0	2	4	1	1	2
1	2	3	4																			
1	3	1	0																			
2	0	0	0																			
3	0	0	2																			
4	1	1	2																			

Şekil 10. $d = 1$ olduğunda farklı oryantasyonlardaki gri seviye birlikte görülme matrisinin hesaplanmasına ilişkin bir örnek (rastgele üretilmiş dört gri seviyeye sahip 5×5 çözünürlüğünde bir 2D gri-seviye görüntü kullanılmıştır)

Sunulan tezde, hesaplama istikrarsızlığı nedeniyle maksimum korelasyon katsayısı özneliği hesaplanmamış, Tablo 4'te gösterilen GLCM'den çıkarılan Haralick özneliklerinden sadece ilk 13'ü kullanılmıştır [82]. RGB (R , G ve B kanallarını kullanarak hesaplanan gri kanal da eklenerek) ve $L^*a^*b^*$ renk uzaylarının her bir kanalı için 0° , 45° , 90° ve 135° 'de bir GLCM oluşturulmuştur. Dolayısıyla, toplam 28 adet GLCM oluşturulmuş ve toplam 364 (28×13) Haralick özneliği çıkarılmıştır. Bu prosedür Şekil 11'de gösterilmektedir.

RGB renk uzayının her kanalı için, 256×256 boyutunda bir GLCM oluşturulmuştur. Bununla birlikte, $L^*a^*b^*$ renk uzayının renk skalaları RGB renk skalasından farklı olduğundan $L^*a^*b^*$ için GLCM üretiminde ilave hesaplamalar yapılmıştır. $L^*a^*b^*$ renk uzayının ilk bileşeni, parlaklık bilgisini içeren ve 0 ile 100 arasında değerleri alan L^* 'dir. Dolayısıyla, L^* kanalı için GLCM 101×101 boyutunda oluşturulmuş ve Haralick öznelikleri bu matris üzerinden hesaplanmıştır. $L^*a^*b^*$ renk uzayının b^* kanalı -128



Şekil 11. Haralick öznelik çıkarma prosedürü. H. Ö., Haralick özneliği için kısaltmadır

(mavimsi yeşil) ile 127 (pembemsi bir magenta) aralığında değer alırken a^* kanalı da -128 (mavi) ile 127 (sarı) aralığında değer almaktadır. Bu kanallar için 128 değeri her bir piksele eklenmiş ve böylelikle aralık 0 ile 255 aralığına kaydırılmıştır. Bu işlemden sonra, a^* ve b^* kanalları için ayrı ayrı 256x256 boyutunda birer GLCM oluşturulmuştur. Haralick öznelikleri bu matrisler kullanılarak hesaplanmıştır.

2.1.3. Ortalama Parlaklık Değerleri

Doku ve renk, görüntü segmentasyonunda kullanılan temel özelliklerdir. Bazı sınıflandırma işlemlerinde tek başına renk özelliğinin ayırt edicilikte yeterli olabileceğine inanılmaktadır. Bu mantıkla, rekabetçi bir öznelik matrisi oluşturabilmek için, RGB (gri kanal dahil), $CIE L^*a^*b^*$, HSV ve $YCbCr$ renk uzayına ait renk kanallarının ortalama piksel değerleri bir öznelik matrisi içinde toplanmıştır. Böylelikle, her bir doku örneği için toplamda 13 öznelik çıkartılmıştır.

Tablo 4. Haralick öznelikleri

<p>Açısal İkinci Moment (Tekdüzelik veya enerji olarak da bilinir):</p> $f_1 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{ij}^2$	<p>Toplam Entropi:</p> $f_8 = - \sum_{i=2}^{2N_g} \left\{ \sum_{j=1}^{N_g} \sum_{k=1}^{N_g} p_{jk} \right\} \log \left\{ \sum_{j=1}^{N_g} \sum_{k=1}^{N_g} p_{jk} \right\}$
<p>Kontrast:</p> $f_2 = \sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{\substack{i=1 \\ i-j =n}}^{N_g} \sum_{j=1}^{N_g} p_{ij} \right\}$	<p>Entropi:</p> $f_9 = - \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{ij} \log(p_{ij})$
<p>Korelasyon:</p> $f_3 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{(i - \mu_x)(j - \mu_y)p_{ij}}{\sigma_x \sigma_y}$ <p>μ_x ve σ_x GLCM matrisinin satır bazlı ortalama ve standart sapması; μ_y ve σ_y GLCM matrisinin sütun bazlı ortalama ve standart sapması.</p>	<p>Fark Varyansı:</p> $f_{10} = \sum_{k=0}^{N_g-1} \left[k - \sum_{m=0}^{N_g-1} m \left(\sum_{\substack{i=1 \\ i-j =m}}^{N_g} \sum_{j=1}^{N_g} p_{ij} \right) \right]^2 \left(\sum_{\substack{i=1 \\ i-j =k}}^{N_g} \sum_{j=1}^{N_g} p_{ij} \right)$
<p>Kareler Toplamı (Varyans):</p> $f_4 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - \mu)^2 p_{ij}$	<p>Fark Entropisi:</p> $f_{11} = - \sum_{i=0}^{N_g-1} \left\{ \sum_{\substack{j=1 \\ j-k =i}}^{N_g} \sum_{k=1}^{N_g} p_{jk} \right\} \log \left\{ \sum_{\substack{j=1 \\ j-k =i}}^{N_g} \sum_{k=1}^{N_g} p_{jk} \right\}$
<p>Ters Fark Momenti (Homojenlik):</p> $f_5 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{1}{1 + (i - j)^2} p_{ij}$	<p>Korelasyonun Bilgi Ölçüleri:</p> $f_{12} = \frac{f_9 - HXY1}{\max(HX, HY)}$ $f_{13} = 1 - e^{\sqrt{-2(HXY2 - f_9)}}$ <p>burada HX ve HY p_x ve p_y'nin entropileri ve</p> $HXY1 = - \sum_{ij} p_{ij} \log(p_x(i)p_y(j))$ $HXY2 = - \sum_{ij} p_x(i)p_y(j) \log(p_x(i)p_y(j))$ $p_x(i) = \sum_{j=1}^{N_g} p_{ij}, p_y(j) = \sum_{i=1}^{N_g} p_{ij}$
<p>Toplam Ortalama:</p> $f_6 = \sum_{i=2}^{2N_g} i \left\{ \sum_{\substack{j=1 \\ j+k=i}}^{N_g} \sum_{k=1}^{N_g} p_{jk} \right\}$	<p>En Büyük Korelasyon Katsayısı:</p> $f_{14} = \sqrt{Q'nun ikinci en büyük özdeğeri}$ <p>burada</p> $Q(i, j) = \sum_{k=1}^{N_g} i \frac{p_{ik}p_{jk}}{p_x(i)p_y(j)}$
<p>Toplam Varyans:</p> $f_7 = \sum_{i=2}^{2N_g} (i - f_8)^2 \left\{ \sum_{\substack{j=1 \\ j+k=i}}^{N_g} \sum_{k=1}^{N_g} p_{jk} \right\}$	

N_g : görüntüdeki gri seviye sayısı; p_{ij} : Normalize edilmiş GLCM'deki (i, j) . eleman.

2.2. Yanık Görüntülerinin Segmentasyonu

Sunulan tezde deri ve yanık sınıflandırma işlemi, segmente edilmiş görüntü bölgeleri üzerinden çıkartılan öznelikler kullanılarak yapılmaktadır. Dolayısıyla sınıflandırma işlemine geçmeden önce her bir yanık görüntüsü homojen piksel gruplarına bölünmekte, daha sonra ise eğitilmiş sınıflandırıcılar tarafından bu piksel grupları sınıflandırılmaktadır. Tezin bu bölümünde, tezde kullanılan segmentasyon ve sınıflandırma metotları hakkında teorik bilgilere yer verilmektedir.

Segmentasyon işlemi için, literatürde sıklıkla kullanılan Bulanık c-Ortalamlar, k-Ortalamlar, Beklenti Maksimizasyonu [83] ve Basit Lineer İteratif Kümeleme [84] yöntemleri yapılan bazı modifikasyonlarla yanık görüntülerine uygulanmıştır. Renk uzaylarının segmentasyon üzerindeki etkisini görmek için, her bir test görüntüsü *RGB* renk uzayından *L*a*b**, *HSV* ve *YCbCr* renk uzaylarına dönüştürülmüştür. Her bir segmentasyon yönteminin yapısına göre, *CIE L*a*b** uzayının *a** ve *b** kanalları, *HSV* uzayının *H* ve *S* kanalları ile *YCbCr* [85] renk uzayının *Cb* ve *Cr* kanalları kümeleme işleminde ayrı ayrı giriş verileri olarak kullanılmıştır. Veri tabanındaki görüntülerin arka planlarının karmaşık yapısı nedeniyle kümelerin sayısı 3 (deri, yanık, arka plan) olarak ayarlanamamıştır. k-Ortalamlar, FCM ve BM kümeleme yöntemlerinde, Calinski-Harabasz kriteri her bir test görüntüsünde en uygun küme sayısını belirlemek için kullanılmıştır. Bununla birlikte, SLIC yönteminde, segmente edilmiş her bir bölge yaklaşık olarak 50 pikselden oluşacak şekilde görüntü boyutuna göre kümeleme işlemi yapılmıştır.

2.2.1. k-Ortalamlar Yöntemi

Hiyerarşik olmayan kümeleme yöntemleri arasında yer alan k-ortalamlar yöntemi küme içi benzerlikler en yüksek, kümeler arası benzerlikler ise en düşük olacak şekilde n adet veriden oluşan veri setini önceden belirlenen k adet kümeye bölmeyi amaçlar. Bu yöntemde en iyi k adet küme Eşitlik 3'teki amaç fonksiyonunun minimize edilmesi ile bulunur.

$$J = \sum_{i=1}^n \sum_{j=1}^k D_{ij}^2 \quad (3)$$

Eşitlik 3'te, k küme sayısını ve n nesne sayısını göstermektedir. D_{ij}^2 , nesnelere küme merkezleri arasındaki uzaklık ölçüsüdür. Kümeleme analizlerinde uzaklık normu olarak genellikle Öklid uzaklıkları kullanılmakta (L2 normu) ve $\|x_{ij} - c_j\|^2$, $1 \leq j \leq k$ şeklinde hesaplanmaktadır. Burada x_{ij} , j 'inci kümedeki i 'inci nesneyi; c_j ise j 'inci küme merkezi prototipini ifade etmektedir. k-Ortalamlar algoritması aşağıdaki adımlardan oluşmaktadır [86].

- 1) X veri setinden rastgele olarak k adet küme merkezi seçilir.
- 2) Veri noktaları ile küme merkezleri arasındaki uzaklıklar hesaplanır.
- 3) Veri noktaları uzaklığı kendilerine en küçük olan merkezlerin ait olduğu kümelere atanır.
- 4) Küme merkezleri Eşitlik 4 ile güncellenir.

$$v_j = \sum_{i=1}^{n_j} \frac{x_{ij}}{n_j}, \quad 1 \leq j \leq k \quad (4)$$

- 5) Küme değiştiren veri noktaları yoksa ya da birbirini izleyen iki adımda hata kümelerindeki artış tanımlanmış bir yaklaşıma değerine eşit veya küçükse kümeleme sona erdirilir, değilse 2. adıma geçilerek işlemler tekrarlanır.

Sunulan tezde kullanılan k-Ortalamlar algoritmasında uzaklık hesaplamalarında Öklid uzaklığı kullanılmıştır. Küme merkezleri için başlangıç değeri olarak verilen değerlerin optimize edilebilmesi için ise k-Ortalamlar++ algoritması tercih edilmiştir. $D(x)$ ile veri noktası x 'ten seçilen merkez noktasına olan en kısa mesafe gösterilirse, bu algoritmanın adımları şu şekildedir:

- 1) Tekdüze dağılım olarak rastgele bir c_1 küme merkezi seçilir.
- 2) Bir sonraki küme merkezi olan $c_i = x' \in X$, $\frac{D(x')^2}{\sum_{x \in X} D(x)^2}$ olasılığı ile seçilir.
- 3) Adım 2 k adet merkez seçilene kadar tekrar edilir.
- 4) Standart k-Ortalamlar algoritmasındaki adımlar ile devam edilir [87].

2.2.2. Bulanık c-Ortalamlar Yöntemi

Bulanık c-Ortalamlar (Fuzzy c-Means-FCM) yöntemi, 1973'te Dunn [88] tarafından geliştirilen ve 1981'de Bezdek [89] tarafından modifiye edilen, veriyi iki veya daha fazla

kümeyle bölme için bir yöntemdir. $X = \{x_1, x_2, \dots, x_n\}$, $x_k \in \mathcal{R}^p$ ($1 \leq k \leq n$) olmak üzere n adet etiketsiz öznelik veri vektörünün bir kümesi; c ($2 \leq c \leq n$), X kümesinde tanımlanan bulanık kümelerin sayısı, $u_{ik} = \{u_i(x_k), 1 \leq i \leq c, 1 \leq k \leq n\}$, k 'inci vektörün i 'inci kümeyle olan üyelik fonksiyonunu göstermek üzere Eşitlik 5-7 ile belirtilen üç şart sağlandığı takdirde X 'te sınırlandırılmış bulanık c küme bulunmuş olur.

$$0 \leq u_{ik} \leq 1, \quad \forall i, k \quad (5)$$

$$0 < \sum_{k=1}^n u_{ik} < n, \quad \forall i \quad (6)$$

$$\sum_{i=1}^c u_{ik} = 1, \quad \forall k \quad (7)$$

Eşitlik 7'deki koşul sağlanmadığı takdirde, bulanık c kümeleme işlemi kısıtsızdır denmektedir. FCM, kısıtlı optimizasyon işlemi ile amaç fonksiyonunu minimize ederek c adet bulanık kümeyle saptayabilmektedir. FCM kümeleme algoritmasında gerçekleştirilen bu optimizasyon prosedürü Eşitlik 8'de gösterildiği şekildedir [90; 91].

$$\text{Minimum} \begin{cases} J_m(U, V; X) = \sum_{k=1}^n \sum_{i=1}^c (u_{ik})^m \|x_k - v_i\|_A^2 \\ \text{Kısıt: } \sum_{i=1}^c u_{ik} = 1, \quad \forall k \end{cases} \quad (8)$$

Eşitlik 8'de $U = \{[u_{ik}], 1 \leq i \leq c, 1 \leq k \leq n\}$ bölme (partition) matrisi; $V = \{[v_i], 1 \leq i \leq c\}$, $v_i \in \mathcal{R}^p$ elde edilen küme merkezleri vektörü; $m \in (1, \infty)$ üyelik derecesi ağırlık etkisini ayarlamak için bir indekstir ve $\| \cdot \|_A$ herhangi bir iç çarpım normdur.

Eşitlik 9 ve Eşitlik 10'da, Eşitlik 8'deki sınırlandırılmış optimizasyon problemini çözmek için kullanılan küme merkezleri ve ilgili üyelik fonksiyonlarına ilişkin bağıntılar verilmektedir.

$$v_i = \frac{\sum_{k=1}^n u_{ik}^m x_k}{\sum_{k=1}^n u_{ik}^m}, \quad 1 \leq i \leq c \quad (9)$$

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{\|x_k - v_i\|_A}{\|x_k - v_j\|_A} \right)^{2/(m-1)}}, \quad 1 \leq i \leq c, \quad 1 \leq k \leq n \quad (10)$$

Eşitlik 9 ve Eşitlik 10 tekrarlamalı bir optimizasyon prosedürü teşkil etmektedir [90]. Literatürde, iki çeşit iterasyon algoritması bulunmaktadır [92]. Bu algoritmaların birincisi, k'inci iterasyon için, sırayla V_{k-1} , U_k , V_k 'yi hesaplar ve $\|V_k - V_{k-1}\|_{err} \leq \mathcal{E}$ şartını doğrular. İkinci tip algoritma ise sırayla U_{k-1} , V_k , U_k 'yi hesaplar ve $\|U_k - U_{k-1}\|_{err} \leq \mathcal{E}$ koşulunu kontrol eder. İkinci algoritmada birinci algoritmaya göre yakınsaması gereken daha fazla parametre olduğu için, ikinci algoritma daha yavaştır [90; 92].

Birinci tip FCM kümeleme algoritmasının adımları aşağıdaki gibi özetlenebilir [90]:

- 1) c küme sayısı, m indeks değeri ve v_1, v_2, \dots, v_c küme merkezlerinin başlangıç değerleri seçilir.
- 2) Üyelik dereceleri olan u_{ik} ($1 \leq i \leq c, 1 \leq k \leq n$) Eşitlik 10 kullanılarak hesaplanır.
- 3) Eşitlik 9 kullanılarak $v_1^{yeni}, v_2^{yeni}, \dots, v_c^{yeni}$ güncellenmiş küme merkezleri elde edilir.
- 4) Eğer $\max_i \left\{ \|v_i - v_i^{yeni}\|_{err} \right\} < \mathcal{E}$ şartı sağlanırsa algoritma durdurulur, şart sağlanmazsa Adım 2'ye dönlür.

2.2.3. Beklenti Maksimizasyonu Yöntemi

Beklenti Maksimizasyonu (BM) algoritması, 1977 yılında ilk olarak Dempster ve diğerleri [83] tarafından öne atılmış bir yöntemdir. k-Ortalamlar algoritmasına benzer bir kümeleme algoritması olmasına rağmen, kümelere dahil edilecek nesnelere belirlenme biçimi olasılıksal bir yaklaşım içermektedir. İteratif bir yapıya sahip olan bu algoritma incelenmek istenen verinin yerel maksimum olasılığını bulmak için kullanılmaktadır. Maksimum olabilirlik hesabı Gauss Karışım Modeli (Gaussian Mixture Model-GMM) ile

modellenecek yapılmaktadır. Bir verinin Gauss karışım yoğunluk fonksiyonu Eşitlik 11’de gösterildiği gibi hesaplanmaktadır.

$$P(X, \theta) = \sum_{i=1}^M \alpha_i P_i(X, \theta_i) \quad (11)$$

Eşitlik 11’deki $X = [X_1, X_2, \dots, X_s]$, s boyutlu rastgele değişken bir vektör; s , veri histogramının gri seviyedeki maksimum değeri; α_i , verinin her bir bileşeninin ağırlıkları; $\theta_i = \{\mu_i, \sigma_i\}$ olmak üzere, μ_i , GMM’deki ortalama değer, σ_i ise standart sapmadır. P_i , GMM’deki dağılım fonksiyonudur ve Eşitlik 12’deki gibi ifade edilir.

$$P_i(X, \theta_i) = \left(\frac{1}{\sigma_i \sqrt{2\pi}} \right) e^{-\frac{\|x - \mu_i\|^2}{2\sigma_i^2}} \quad (12)$$

BM, iki adımdan oluşan bir algoritmadır. Birinci adım beklenti (E) adımı, ikinci adım ise maksimizasyon (M) adımıdır. Beklenti adımında ilk olarak olasılık yoğunluk fonksiyonu oluşturulur. $P(X, Y|\theta)$, olasılık yoğunluk fonksiyonudur. X , giriş verilerini, Y ise hesaplanmamış veya eksik verileri ifade eder. Olasılık yoğunluk fonksiyonu oluşturulduktan sonra logaritmik olasılık fonksiyonu elde edilir. Olasılık yoğunluk fonksiyonu, $L(\theta)$ ile gösterilir. Her bir veri için θ parametresi, t . iterasyona kadar devam ettirilir. Verinin tamamına ait olasılık fonksiyonlarının beklenen değerleri Eşitlik 13 ile hesaplanır. Böylelikle verilen gözlem değerleri ile eksik verilerin ve model parametrelerinin tahmini yapılmış olur.

$$\Phi(\theta|\theta^{(t)}) = E_{Y|X, \theta^{(t)}}[\log L(\theta; X, Y)] \quad (13)$$

Daha sonra M adımında, beklenti adımında bulunan her bir θ parametresi için Eşitlik 14 ile maksimum değerler hesaplanır.

$$\theta^{(t+1)} = \arg_{\theta} \max \Phi(\theta|\theta^{(t)}) \quad (14)$$

Bu işlem hesaplanan parametrelerdeki meydana gelen değişimlerin sona ermesine kadar devam eder. Hesaplanan logaritmik benzerlik fonksiyonuna göre giriş verilerinden

çıkış verileri elde edilir. Elde edilen çıkış verileri ilgili kümelere dahil edilir. Böylece görüntü segmentasyon işlemi tamamlanmış olur [93].

2.2.4. Basit Doğrusal İteratif Kümeleme

Süperpiksel, benzer özellikler sergileyen piksellerin birleştirilerek tek bir piksel ile ifade edilmesidir. Basit Doğrusal İteratif Kümeleme (Simple Linear Iterative Clustering-SLIC) [84], görüntüde süperpikseller oluşturmak amacıyla kullanılan ve k-Ortalamlar algoritmasının uyarlaması olan bir segmentasyon yöntemidir. Sahip olduğu iki önemli özellikle k-Ortalamlar yönteminden ayrılmaktadır.

- 1) Arama uzayını süper piksel boyutuyla orantılı olarak sınırlandırmak optimizasyon sürecindeki uzaklık hesaplama işlemlerinin sayısında yüksek oranda düşüşe neden olmaktadır.
- 2) Ağırlıklı bir mesafe ölçüsü, renk ve konumsal yakınlık bilgilerini birleştirirken, aynı zamanda süper piksellerin boyutu ve kompaktlığı üzerinde kontrol sağlar.

SLIC algoritmasının kullanımı ve anlaşılması kolaydır. Algoritmanın tek parametresi olan k , yaklaşık olarak eşit büyüklükteki süper piksellerin istenen sayısıdır. $CIE L^*a^*b^*$ renk uzayındaki renkli görüntüler için kümeleme prosedürü, k adet başlangıç küme merkezinin ($C_i = [l_i \ a_i \ b_i \ x_i \ y_i]^T$) S piksel aralıklı düzende bir kafes (grid) üzerinde örneklendiği bir başlangıç adımıyla başlar. Kabaca eşit boyutta süperpikseller üretmek için kafes aralığı $S = \sqrt{N/k}$ olarak alınır (Burada N görüntüdeki toplam piksel sayısıdır). Merkezler, 3x3 piksel komşuluktaki en düşük gradyan konumuna karşılık gelen kaynak konumlarına taşınır. Bu işlem, bir süperpikselin kenar üzerinde merkezlenmesini önlemek için yapılır. Daha sonra, atama adımında, her bir piksel i , arama bölgesi konumu ile çakışan en yakın küme merkezi ile ilişkilendirilir. Algoritmanın hızlanmasındaki anahtar nokta da budur, çünkü arama bölgesinin boyutunu sınırlamak, uzaklık hesaplamalarının sayısını önemli bir şekilde azaltmaktadır. Bu da her bir pikselin tüm küme merkezleriyle karşılaştırıldığı geleneksel k-Ortalamlar kümelemesine kıyasla önemli bir hız avantajı sağlamaktadır. Bu işlem, her bir piksel için en yakın küme merkezini belirleyen bir D uzaklık ölçüsü kullanılarak yapılmaktadır. Bu uzaklık $Labxy$ uzayında tanımlanmış 5 boyutlu (5D) bir Öklid uzaklığıdır. $CIE L^*a^*b^*$ renk uzayında bir piksele ait renk $[L \ a \ b]^T$ ile belirtilmektedir ve her bir renk kanalının alt-üst sınırları bilinmektedir. $[x \ y]^T$ ile ise piksele ait konum koordinatlarını göstermektedir ve bu değerler görüntünün boyutuna göre değişiklik gösterebilmektedir.

Dolayısıyla 5D Öklit uzaklığı kullanılırken geniş süperpikselerde konum parametresine renge oranla daha fazla ağırlık verilebilmektedir. Küçük süperpikseller için ise tam tersi bir durum söz konusudur. Konumsal ve renge ilişkin uzaklıkları tek bir ölçüde birleştirebilmek için renk yakınlığı, ilişkili küme içindeki maksimum renk uzaklığı (N_c) ile; konum yakınlığı ise ilişkili küme içindeki maksimum konum uzaklığı (N_s) ile normalize edilmelidir. Normalize edilmiş uzaklık ölçüsü D' Eşitlik 15'teki gibi yazılabilir.

$$d_c = \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2},$$

$$d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2},$$

$$D' = \sqrt{\left(\frac{d_c}{N_c}\right)^2 + \left(\frac{d_s}{N_s}\right)^2}$$
(15)

Belirli bir küme içinde beklenen maksimum konumsal mesafe, örnekleme aralığına karşılık gelmelidir, $N_s = S = \sqrt{N/k}$. Maksimum renk mesafesi olan N_c 'yi belirlemek ise o kadar kolay değildir, çünkü renk mesafeleri kümeden kümeye ve görüntüden görüntüye önemli ölçüde değişebilmektedir. Bu problem, N_c 'nin bir m değerine sabitlenmesiyle Eşitlik 16'daki şekilde önlenebilir.

$$D' = \sqrt{\left(\frac{d_c}{m}\right)^2 + \left(\frac{d_s}{S}\right)^2} = \sqrt{d_c^2 + \left(\frac{d_s}{S}\right)^2} m^2$$
(16)

D' 'yi bu şekilde tanımlamak, renk benzerliği ve konumsal yakınlık arasındaki göreceli önemin tartılmasına da izin vermektedir. m büyük olduğunda, konumsal yakınlık daha önemlidir ve ortaya çıkan süperpikseller daha kompakttır (Yani süperpiksellerin alan/çevre oranı daha düşüktür). m küçük olduğunda, elde edilen süperpikseller görüntü sınırlarına daha sıkı yapışır, ancak daha az düzenli boyut ve şekle sahiptirler. $CIE L^*a^*b^*$ renk uzayını kullanırken, m [1,40] aralığında alınabilir.

Bir süperpikselin beklenen konumsal genişliği yaklaşık $S \times S$ büyüklüğünde olduğundan, benzer piksellerin aranması, süperpiksel merkezinin etrafındaki $2S \times 2S$ 'lik bir

alandaki yapıdır. Her bir piksel en yakın küme merkezi ile ilişkilendirildikten sonra bir güncelleme adımı, küme merkezlerini, kümeye ait tüm piksellerin ortalama $[L \ a \ b \ x \ y]^T$ vektörü olarak ayarlar. Yeni küme merkezi konumları ile önceki küme merkezi konumları arasındaki artık hata E 'yi hesaplamak için L_2 normu kullanılır. Atama ve güncelleme adımları, hata yakınsayana kadar iteratif olarak tekrarlanabilir. İşlem sonrası bir adım olarak, ayrık kalmış pikseller yakınlarında bulunan süperpiksellere atanırlar.

2.2.5. Calinski-Harabasz İndeksi ile Optimum Küme Sayısının Belirlenmesi

Calinski ve Harabasz tarafından önerilen Calinski-Harabasz (CH) indeksi [94], küme içi ve kümeler arası kareler toplamını temel alarak k kümeye sahip bir kümelemenin geçerliliğini (kalitesini) ölçmekte ve optimal küme sayısının bulunmasını sağlamaktadır.

Kümeler arası kareler toplamı, SS_B ; küme içi kareler toplamı, SS_W ; küme sayısı, k ve gözlem sayısı n ile gösterildiğinde CH indeksi Eşitlik 17 kullanılarak hesaplanır.

$$CH = \frac{SS_B}{SS_W} \times \frac{(n - k)}{(k - 1)} \quad (17)$$

k , küme sayısını; m_i , k kümesine ilişkin küme merkezini; μ , tüm örneklerin ortalamasını ve $\|m_i - \mu\|$, kümenin merkezi ile tüm örneklerin ortalaması arasındaki Öklid uzaklığını göstermek üzere, ortalama kümeler arası kareler toplamı Eşitlik 17 ile bulunur.

$$SS_B = \sum_{i=1}^k n_i \|m_i - \mu\|^2 \quad (18)$$

x , bir örneği; p_i , i 'inci kümeyi; m_i , p_i kümesinin merkezini; $\|x - m_i\|$, örnek ile küme merkezi arasındaki Öklid uzaklığını göstermek üzere, ortalama küme içi kareler toplamı Eşitlik 18 ile bulunur.

$$SS_W = \sum_{i=1}^k \sum_{x \in p_i} \|x - m_i\|^2 \quad (19)$$

Büyük CH indeks değeri kümelemenin daha iyi yapıldığını gösterir. Bu nedenle, iyi tanımlanmış bir kümelemenin SS_B değerinin yüksek, SS_W değerinin ise düşük olması beklenir. CH indeksine göre maksimum CH değerine ulaşılan küme sayısı, optimal küme sayısı olarak seçilmektedir.

2.3. Yanık Görüntülerinin Sınıflandırılması

Bu bölümde sınıflandırma teknikleri, deri, yanık ve arka plan doku örneklerini sınıflandırmadaki performansları açısından karşılaştırılmıştır. İlk olarak, sınıflandırıcılar, Bölüm 2.1.1 ve Bölüm 2.1.2’de çıkarılan gri seviye histogram istatistikleri ve Haralick özniteliklerinin kombinasyonunu içeren Öznitelik Matrisi 1 (ÖM-1) ile eğitilmiştir. ÖM-1 toplamda 413 öznitelikten oluşmuştur. Her bir yöntem için, sınıflandırıcıların eğitilmesi ve test edilmesi aşamasında 10-katmanlı çapraz doğrulama uygulanmıştır. Eğitim modellerinin istatistiksel performansları, [95]’te tanımlanan metriklerin makro ve mikro ortalama versiyonları kullanılarak ölçülmüştür. Bir sonraki adımda, daha iyi veya eşdeğer performansı sağlayabilecek optimum sayıda özniteliğe sahip daha basit bir modelin bulunabilmesi için ileri seçim algoritmasından [96] yararlanılmıştır. Seçilen öznitelikler, öznitelik matrisi ÖM-2’nin oluşturulması için kullanılmıştır. Son olarak, sınıflandırıcılar Bölüm 2.1.3’te oluşturulan 13 öznitelikten oluşan öznitelik matrisi (ÖM-3) ile eğitilmiş ve eğitim modellerinin performansları karşılaştırılmıştır. 3 farklı öznitelik matrisi ile eğitilen sınıflandırıcılardan hangisinin bölümlere ayrılmış yanık görüntü bölgelerini sınıflandırmada en yüksek performansı vereceği tespit edilmiştir.

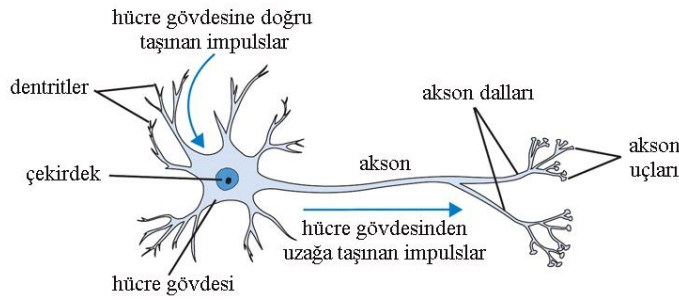
Bu bölümde, tezde kullanılan sınıflandırma yöntemlerinin teorik alt yapısına ilişkin bilgiler verilmektedir. Bu bilgiler verilirken Derin Öğrenme’nin temelini oluşturan YSA detaylı bir şekilde anlatılırken, diğer yöntemlerin işleyişi ana hatlarıyla ele alınmıştır.

2.3.1. Yapay Sinir Ağları

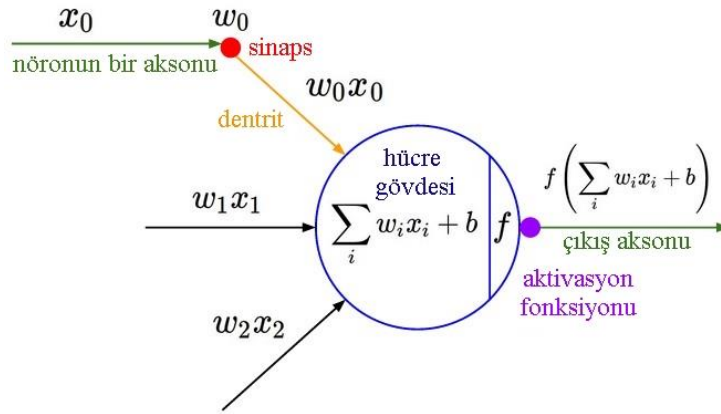
İlk olarak 1943 yılında nörofizikçi Warren McCulloch ve matematikçi Walter Pitts [97] tarafından yapay sinir modelinin sunulması ile gündeme gelen ve yıllar ilerledikçe gelişmeye devam eden YSA, insan beynindeki sinir hücrelerinin (nöron) çalışma yapısını taklit ederek mevcut verileri analiz edip öğrenme yolu ile yeni bilgiler üretebilen bilgi işlem sistemleridir [98].

YSA, yapay sinir hücrelerinden oluşmaktadır. Literatürde sinir hücreleri; nöron, düğüm, birim veya işlemci eleman gibi farklı şekillerde adlandırılabilir. İnsan sinir sisteminde yaklaşık 100 milyar nöron ve yaklaşık 1 katrilyon sinaps bulunmaktadır [99].

Yapay sinir sisteminde biyolojik sinir hücresinin yapısını oluşturan her bir elemana karşılık matematiksel bir karşılık bulunmaktadır. Bu karşılıklar Tablo 5’te, biyolojik bir nöronun ve ona karşılık gelen matematiksel modelin bir gösterimi ise Şekil 12’de verilmektedir. Her bir nöron, dendritlerinden giriş sinyallerini almakta ve aksonu boyunca çıkış sinyalleri üretmektedir. Akson sona doğru yaklaşırken dallanmakta ve diğer nöronların dendritlerine sinapslar yoluyla bağlanmaktadır. Nörona ilişkin matematiksel modelde, aksonlar boyunca hareket eden sinyaller (x_0), sinapstaki sinaptik kuvvete (w_0) bağlı olarak diğer nöronun dendritleri ile çarpımsal olarak (w_0x_0) etkileşime girmektedir. Buradaki fikir, sinaptik kuvvetlerin (w ağırlıkları) öğrenilebilmesi ve bir nöronun diğer bir nörona olan etkisinin gücünü (ve pozitif veya negatif ağırlıklara göre yönünü) kontrol edebilmesidir. Temel modelde dendritler sinyali hücre gövdesine taşımakta ve tüm sinyaller burada



Biyolojik sinir hücresine ait çizimsel bir gösterim



Sinir hücresinin matematiksel gösterimi

Şekil 12. Sinir ağının biyolojik ve matematiksel gösterimi

Tablo 5. Biyolojik sinir sistemi elemanları ve yapay sinir sisteminde karşılıkları

Biyolojik Sinir Sistemi	Yapay Sinir Sistemi
Nöron	İşlemci Elemanı
Dentrit	Toplama Fonksiyonu
Hücre gövdesi	Transfer Fonksiyonu
Aksonlar	Yapay Nöron Çıkışı
Sinapslar	Ağırlıklar

toplanmaktadır. Bu toplama son olarak bir de yanlılık değeri (b) eklenmektedir. Bu değer nörona haricen uygulanan bir ön değerdir. Oluşan son toplamın belirli bir eşğin üstünde olması durumunda, nöron uyarılmakta ve aksonu boyunca bir dürtü iletimine başlamaktadır. Matematiksel modelde, iletilen dürtüdeki bilginin frekansı önem arz ederken, dürtülerin kesin zamanlarının bir öneminin olmadığı varsayılmaktadır. Akson boyunca iletilen dürtüdeki uyarımın frekansı f aktivasyon fonksiyonu ile temsil edilmektedir. Gerçek değerli bir girdiyi (toplamdan sonraki sinyal gücü) alıp onu 0 ile 1 aralığında bir değere dönüştürebilme işlevine sahip olduğu için literatürde en yaygın olarak kullanılan aktivasyon fonksiyonu σ sigmoid fonksiyonudur. Eşitlik 20’de toplayıcı fonksiyonun ve yanlılık değerinin, Eşitlik 21’de ise toplayıcı fonksiyonun çıkışına uygulanan aktivasyon fonksiyonun denklemleri verilmektedir.

$$net = w_1x_1 + w_2x_2 + \dots + w_nx_n + b = \sum_{i=1}^n w_ix_i + b \quad (20)$$

$$y = f(net) = f\left(\sum_{i=1}^n w_ix_i + b\right) \quad (21)$$

YSA, doğrusal olmama, paralel çalışabilme, öğrenebilme, genelleme yapabilme, hata toleransına ve esnekliğe sahip olma, eksik verilerle çalışabilme, çok sayıda değişken ve parametre kullanabilme ve uyarlanabilme gibi birçok özelliğe sahip olmakla birlikte, paralel çalışabilmenin getirdiği donanım bağımlılığı, uygun ağ yapısının belirlenebilmesinde ve ağın davranışlarının açıklanabilmesindeki zorluklar, ağ eğitim süresinin uzun olması gibi bazı dezavantajlara da sahiptir [98]. Tablo 6’da geleneksel algoritmalar ile YSA’ların karşılaştırması yapılmaktadır.

YSA, başta sınıflandırma ve tahmin olmak üzere, ilişkilendirme veya örüntü eşleştirme, zaman serileri analizleri, sinyal filtreleme, veri sıkıştırma, örüntü tanıma, doğrusal olmayan sinyal işleme, doğrusal olmayan sistem modelleme, optimizasyon gibi birçok uygulamada kullanılmaktadır [98]. Bu uygulamaları içeren başlıca sektörler ise uzay, otomotiv, bankacılık, savunma, elektronik, eğlence, finans, sigortacılık, üretim, sağlık, petro-kimya, robotik, telekomünikasyon ve güvenlik olarak sayılabilir.

Yapay sinir ağları katman sayılarına göre tek katmanlı ve çok katmanlı, öğrenme yöntemine göre gözetimli, gözetimsiz ve yarı gözetimli, nöronlar arası bağlantı yapılarına göre ileri beslemeli ve geri beslemeli olarak sınıflandırılırlar [101].

2.3.1.1. Tek Katmanlı Algılayıcı (Perceptron)

İlk kez 1958 yılında Cornell Üniversitesi'nden psikolog Frank Rosenbatt tarafından şekil sınıflandırma amacıyla geliştirilen en basit tek katmanlı sinir ağı modelidir. Bu ağ sadece girdi ve çıktı katmanlarından oluşur. Giriş sinyalleri bütün çıkış nöronlarına bağlanmaktadır. Her bağlantının her çıkış nöronu için ayrı bir ağırlık değeri vardır [98].

Doğrusal sınıflandırıcı olarak kullanılan tek katmanlı algılayıcı ağlarda, çıktı fonksiyonu ikili değer üreten bir eşik değer fonksiyondur. Sınıfları temsil eden çıktı fonksiyonu doğrusaldır ve kullanılan nöron modeline (Perceptron, Adaline (Adaptif Lineer Nöron (Adaptive Linear Neuron)) bağlı olarak $\{0, 1\}$ veya $\{-1, 1\}$ değerlerini alabilmektedir.

Tablo 6. Geleneksel algoritmalar ile YSA'ların karşılaştırması [100]

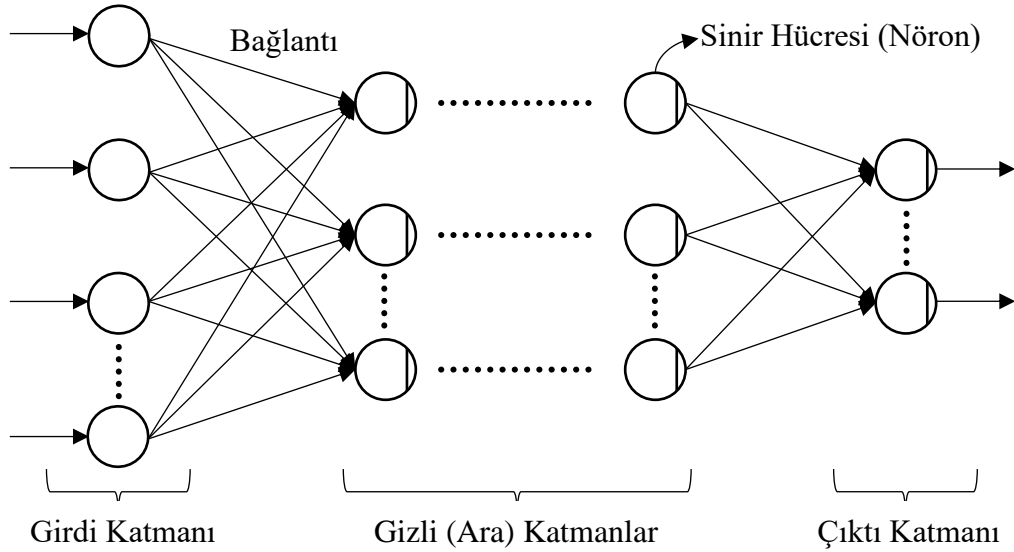
Geleneksel Algoritmalar	Yapay Sinir Ağları
Çıkışlar, koyulan kurallara girişlerin uygulanması ile elde edilir.	Öğrenme esnasında giriş çıkış bilgileri verilerek, kurallar koyulur.
Bilgiler ve algoritmalar kesindir.	Deneyimden yararlanır.
Hesaplama; merkezi, eş zamanlı ve ardışıktır.	Hesaplama; toplu, eş zamansız ve öğrenmeden sonra paraleldir.
Bellek paketlenmiş ve hazır bilgi depolanmıştır.	Bellek ayrılmış ve ağa yayılmıştır.
Hata toleransı yoktur.	Hata toleransı vardır.
Nispeten hızlıdır.	Yavaş ve donanıma bağımlıdır.

Tek katmanlı algılayıcılar girdi değerlerini ikiye ayıracak bir doğru veya düzlem bulmaya çalışmaktadırlar. İki'den fazla sınıfın algılanabilmesi için, birden çok yapay sinir hücresi kullanılması gerekmektedir [98].

2.3.1.2. Çok Katmanlı Algılayıcı (Multilayer Perceptron)

Yapay sinir ağlarının günümüzde en yaygın olarak kullanılan modeli çok katmanlı algılayıcı ağlarıdır. Tek katmanlı algılayıcılarda eğitilebilecek tek bir ağırlık matrisinin olması, ağa basitlik ve hızlilik kazandırmaktadır. Fakat bu tür ağlar sadece karmaşık olmayan lineer problemleri çözebilmektedirler. Bu nedenle, karmaşık ve lineer olmayan problemlerin çözümü için çok katmanlı yapay sinir ağlarına ihtiyaç duyulmaktadır [98; 102].

Çok katmanlı yapay sinir ağları temel olarak Şekil 13'teki gibi Girdi Katmanı, Gizli Katmanlar ve Çıktı Katmanı olmak üzere üç katmandan oluşur [98; 103].



Şekil 13. Çok katmanlı bir yapay sinir ağının genel yapısı

Girdi Katmanı (Input Layer), veri setinde yer alan örneklere ilişkin öğrenilmesi istenen özelliklerin her birinin girdi olarak verildiği katmandır. Girdi katmanındaki nöron sayısı öğretilecek örneklerin özellik sayısı ile eşit olmalıdır. Her bir girdi bir ağırlık değerine sahiptir ve gizli katmandaki nöronlara da bu ağırlıklar ile bağlıdır. Ağırlıklar girdilerin çıktılar üzerindeki etkisini ayarlayabilmek için kullanılan parametrelerdir. Ağırlığı sıfır olan nöronların çıktı üzerinde herhangi bir etkisi olmamaktadır [104].

Gizli Katmanlar (Hidden Layers), girdi katmanı ile çıktı katmanı arasında yer almaktadırlar. Bu katmanlarda girdi katmanından gelen bilgiler işlenir ve bir çıktıya dönüştürülür. Gizli katman sayısı ve bu katmanlar üzerindeki nöron sayısı problemin zorluğuna göre değişebilmektedir. Bu katmanlarda ileri yönlü hesaplamalar ve geri yönlü hata yayılımı yapılmaktadır. Katman sayısı arttıkça hesaplama yükü ve hesaplama süresi de artmaktadır, fakat karmaşık problemlerin çözümü için genelde katman sayıları ve katmanlardaki nöron sayıları fazla tutulmaktadır [105].

Çıktı Katmanı (Output Layer), girdi verisine ilişkin sınıf bilgisi, etiket değeri gibi çıktı değerlerinin tutulduğu katmandır. Seçilen hata hesaplama fonksiyonuna göre, sinir ağı tarafından üretilen çıktı değeri ile beklenen çıktı değeri arasındaki fark hesaplanır. Elde edilen hata değerine göre ağıdaki ağırlıkların güncellenmesi sağlanır. Hata fonksiyonunun değerini minimum yapacak şekilde ağırlıkların güncellenmesi yapay sinir ağının öğrenmesi anlamına gelmektedir. Bu işlem bir optimizasyon işlemidir ve ağırlıkların güncellenmesi için literatürde farklı yöntemler kullanılmaktadır [106].

YSA'da transfer fonksiyonu bir nörona gelen net girdiyi hesaplamaktadır. En yaygın olarak kullanılan transfer fonksiyonunda nörona bağı olan girdi değerleri ile ilgili ağırlıklar çarpılmakta ve sonrasında bu çarpımlar toplanmaktadır. Aktivasyon fonksiyonu nörona gelen net değeri işlemekte ve bu değere karşılık aktivasyon çıktısını üretmektedir. Yapay sinir ağını oluşturan elemanların mevcut doğrusal yapısı, kullanılan aktivasyon fonksiyonu ile doğrusal olmayan bir yapıya dönüştürülmektedir. Aktivasyon fonksiyonunun probleme göre uygun seçilmesi ağı performansını ve başarı oranını önemli derecede etkilemektedir [107].

2.3.1.3. İleri Beslemeli Yapay Sinir Ağları (Feedforward Networks)

İleri beslemeli yapay sinir ağında nöronlar genellikle katmanlara ayrılmıştır. Sinyaller girdi katmanından çıktı katmanına doğru tek yönlü bağlantılarla iletilir. Bir katmandaki nöronlar sadece bir önceki katmanın nöronları tarafından beslenirler. Aynı katmanda bulunan nöronlar arasında bağlantı bulunmaz. İleri beslemeli bağlantıların ağırlıkları eğitim sırasında değiştirilebilir ama geri dönüşümlerin bağlantı ağırlıkları değiştirilemez, sabittirler. Bilgi, gizli katmanlarda ve çıktı katmanında işlendikten sonra ağ çıktısı belirlenir. Bu yapı ile ileri beslemeli ağlar doğrusal olmayan statik bir işlevi gerçekleştirir. Geriye yayılım öğrenme algoritması bu tip yapay sinir ağlarının eğitiminde en çok kullanılan yöntemlerden biridir. Ağ, örneklere ve bu örneklere ilişkin beklenen çıktılara göre eğitilir ve problem

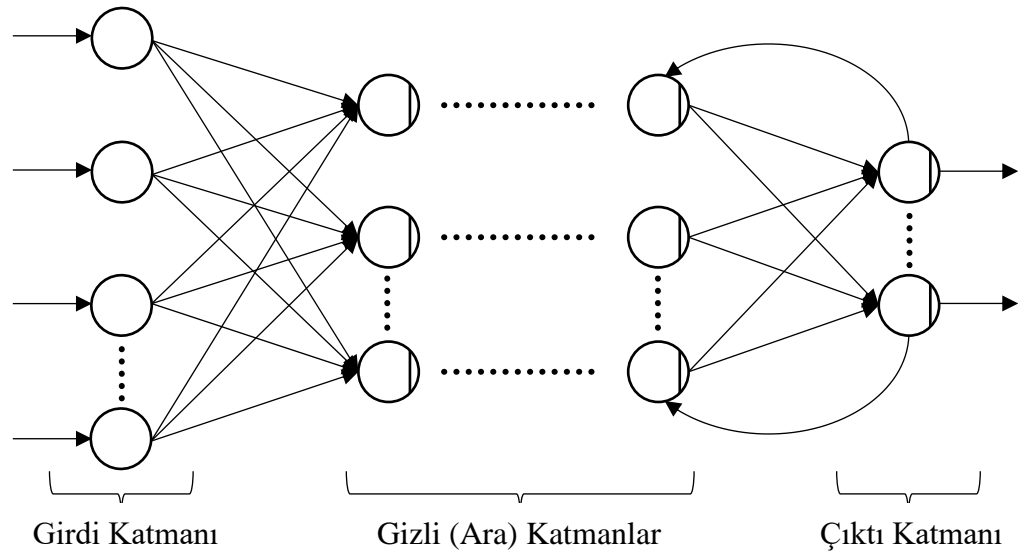
uzayını temsil eden bir çözüm uzayı üretir. Daha sonra gösterilen benzer örnekler için bu çözüm uzayı sonuçlar ve çözümler üretebilmektedir [98]. Şekil 13'te gösterilen Çok Katmanlı Algılayıcılar ileri beslemeli ağlara örnek olarak verilebilir.

2.3.1.4. Geri Beslemeli Yapay Sinir Ağları

Geri beslemeli sinir ağı (Feedback or Recurrent Networks), çıktı katmanındaki ve/veya gizli katmanlardaki çıkışlardan giriş nöronlarına veya önceki gizli katmanlardaki nöronlara geri beslenmenin yapıldığı bir ağ yapısına sahiptir. Girişler hem ileri yönde hem de geri yönde aktarılmaktadır. Geri beslemeli ağda bilgi alışverişi bir katmandaki hücreler arasında olabileceği gibi katmanlar arasındaki hücreler arasında da olabilir. Geri beslemeli ağ bu yapısı ile dinamik bir davranış özelliği göstermekte ve zaman serilerinin tahmini gibi problemlerde başarılı sonuçlar vermektedirler. Geri beslemenin yapılaş sekline göre farklı yapıda ve davranışta geri beslemeli YSA yapıları elde edilebilmektedir. Geri beslemeli ağlara örnek olarak Hopfield, Elman ve Jordan ağları verilebilir [108]. Şekil 14'te geri beslemeli YSA için bir örnek gösterilmektedir

2.3.1.5. Gözetimli Öğrenme

Gözetimli (Danışmanlı/Denetimli) öğrenme, en sık kullanılan öğrenme türlerindedir. Eğitim sırasında sisteme girdi ve hedef çıktı verisi verilir. İstenilen çıktı ile gerçek çıktı



Şekil 14. Geri beslemeli YSA örneği

arasındaki farka (hata) göre, nöronlar arası bağlantıların ağırlıkları, en uygun çıktıyı elde edebilmek için bir öğrenme algoritmasıyla güncellenir. Hata toleransına göre iterasyonlar devam eder. İstenilen çıktı değerlerine yakınsayan değerler elde edilince öğrenme biter. Öğrenme bittikten sonra ağırlıklar değişmez [109].

Gözetimli öğrenmede, test sürecinde eğitim sürecinde kullanılmamış veriler ağırlıklı olarak girer. Çıkan sonuçlar eldeki verilerle karşılaştırılarak ağırlıklı test süreci de tamamlanmış olur. Test aşamasında çıkan sonuçların istenilen seviyede olmaması durumunda ağırlıklı iterasyon sayısı, ara katmanlar, ara katmanlardaki nöron sayıları ya da ağırlıklar değiştirilir. Öğrenme süreci yeni modifikasyonlarla tekrardan başlatılır [101]. Widrow-Hoff tarafından geliştirilen ‘Delta Kuralı’, Rumelhart ve McClelland tarafından geliştirilen ‘Genelleştirilmiş Delta Kuralı’ ve ‘Geri Yayılım (Back-Propagation) Algoritması’ gözetimli öğrenme algoritmalarına örnek olarak verilebilir [110; 111].

2.3.1.6. Gözetimsiz (Danışmansız/Denetimsiz) Öğrenme

Gözetimli öğrenmeden farklı olarak ağırlıklı sadece girer verisi verilir, ancak hedef çıktılar belirtilmez. Sistem, girdiler içerisinde birbirine en çok benzeyenleri gruplar ve her bir grup için farklı bir örüntü tanımlar. Yani girer verileri arasındaki matematiksel ilişkiye göre bağlantı ağırlıkları ayarlanır. Grossberg tarafından geliştirilen ART (Adaptif Rezonans Teorisi (Adaptive Resonance Theory)) ve Kohonen tarafından geliştirilen SOM (Öz-Düzenlemeli Harita Ağı (Self Organizing Map)) öğrenme kuralı danışmansız öğrenmeye örnek olarak verilebilir [110; 111].

2.3.1.7. Yarı Gözetimli (Takviyeli (Reinforcement)) Öğrenme

Gözetimli öğrenmenin özel bir formudur. Bu algoritmada, girer değerlerine karşı istenilen çıktı değerinin bilinmesine gerek yoktur. YSA’ya bir hedef verilmemekte fakat elde edilen çıktının verilen girerdiye karşılık uygunluğunu değerlendiren bir kriter kullanılmaktadır [100]. Genetik algoritmalar yarı gözetimli öğrenmeye örnek olarak verilebilir.

2.3.1.8. YSA’nın Eğitilmesi

YSA’nın en önemli özelliği öğrenebilme yeteneğine sahip olmasıdır. Hedef çıktılar ile tahmin edilen çıktılar arasındaki farkı en aza indirecek şekilde nöronlar arası optimum

ağırlık değerlerinin belirlenmesi işlemine “ağın eğitilmesi” denir. YSA’lar, kendilerine örnekler gösterildikçe bu ağırlık değerlerini yenileyerek amaca ulaşmaya çalışırlar. Öğrenme işleminde hesaplanan hata değerine göre ağırlıklar sürekli olarak güncellenmektedir. Bu güncelleme işleminin nasıl yapılacağına eğitim algoritmaları ile karar verilir. Ağırlıkların sürekli yenilenmesi sırasında istenilen sonuca ulaşılan kadar geçen zamana “öğrenme” denir. Ağırlık değerlerinin değişmesi belirli kurallara göre yürütülmektedir. Bu kurallara “öğrenme kuralları” denir [112].

YSA’da eğitim süreci veri setinin Eğitim, Doğrulama ve Test veri seti olmak üzere rastgele seçimle 3 alt veri setine bölünmesiyle başlar. Genellikle veri örneklerinin %50-70 arası eğitim setinde, %30-50 arası doğrulama ve test setinde yer alır. Doğrulama ve test setine de verilerin rastgele ve eşit oranlarda dağıtılmasına dikkat edilir. Ağ ağırlıklarının güncellenmesi ve nihai değerlerinin kestirimi, eğitim aşamasında sadece eğitim veri örnekleri dikkate alınarak gerçekleştirilir. Eğitim sürecinin ne zaman sonlandıracağına karar vermek ezberleme probleminin önüne geçmek ve en etkili modeli oluşturmak için önemlidir. Ağın eğitimini sonlandırmaya karar vermek için doğrulama veri setindeki hata değişimi dikkate alınır. İterasyon sayısı arttıkça hatanın belirli bir noktaya kadar düştüğü, daha sonra da aynı düzeyde sabit kaldığı, artık hatada azalmanın olmadığı noktanın optimum eğitim noktası olduğu düşünülür [113].

YSA modellerinin öğrenme süreçleri tıpkı insanların öğrenme süreçleri gibi farklılık göstermektedirler. Sinir ağları, bir modelin öğrenme sürecini tanımlayan ve öğrenme denklemleri olarak adlandırılan matematiksel ifadelerle tanımlanmış, öğrenme kurallarına göre işlem yaparlar [114]. Öğrenme kuralı, öngörülmuş bir hata ölçüsünü minimize etme adına ağda kullanılan ağırlıkların nasıl güncellenmesi gerektiğini belirler.

Günümüzde YSA’ların eğitilmesinde kullanılan algoritmaların çoğunluğu Hebb, Delta, Kohonen ve Hopfield isimindeki dört farklı öğrenme kuralından esinlenerek geliştirilmiştir.

1949 yılında psikolog Donald Olding Hebb [115] tarafından geliştirilen ve en eski öğrenme kuralı olarak bilinen Hebb kuralında eğer iki nöron birbirlerinden bilgi alıyorsa ve ikisi de aktifse (matematiksel olarak aynı işaretli) o iki nöron arasındaki bağın kuvvetlenmesi gerekir. Eğer hücrelerin matematiksel işaretleri farklı ise, bu durumda da hücreler arası bağlantılar zayıflatılır, diğer bir ifade ile ağırlıklar azaltılır.

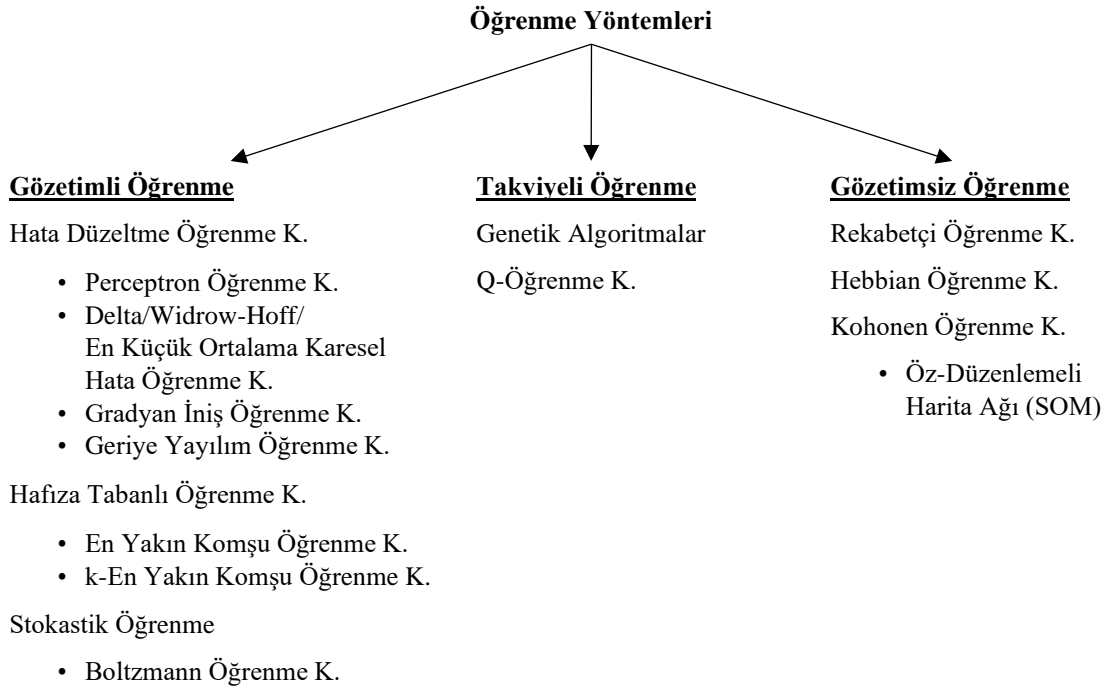
Bernard Widrow ve Ted Hoff [116] tarafından 1959 yılında geliştirilen Delta kuralı, sinir hücresinin hesaplanan çıktı değeriyle beklenen çıktı değeri arasındaki farkı azaltmak

üzere bağlantı ağırlıklarının sürekli olarak güncellenmesi düşüncesine dayanmaktadır. Delta öğrenme kuralı çıktı değerine ait ortalama karesel hatayı minimize etmeyi amaçlar. Hata çıktı katmanından girdi katmanına doğru geri yayılarak azaltılmaya çalışılır. Bu sebepten dolayı Delta kuralı, geri yayılım veya en küçük ortalama karesel öğrenme kuralı olarak da adlandırılmaktadır.

1982 yılında John Hopfield [117] tarafından popüler hale getirilen Hopfield kuralı Hebb kuralına benzemektedir. Bu kurala göre eğer istenilen giriş ve çıkışın her ikisi de aktif ise nöronlar arası bağlantı öğrenme katsayısı kadar artırılırken, her ikisi de pasif ise, bağlantı ağırlıkları öğrenme katsayısı kadar azaltılmaktadır. Öğrenme katsayısı genel olarak 0-1 arasında kullanıcı tarafından seçilen bir değerdir.

Teuvo Kohonen tarafından geliştirilen Kohonen kuralı [118] biyolojik sistemlerdeki öğrenmeden esinlenmiştir. Bu kuralda ağırlıkların, ağırlıklarını değiştirmek için birbirleriyle yarışır. En büyük ağırlığa sahip nöron kazanan nöron olur ve çevresindeki nöronları sınırlama yetkisine sahiptir.

Uygulamalarda en yaygın olarak kullanılan öğrenme algoritması olmasından ve sunulan tezde Bölüm 3.5'te yanık görüntülerini sınıflandırmak için kullanılmasından dolayı geri yayılım algoritması detaylı bir şekilde incelenmiştir. YSA'da kullanılan başlıca öğrenme kuralları Şekil 15'te gösterilmektedir.



Şekil 15. YSA'da kullanılan öğrenme kuralları

2.3.1.9. Geri Yayılım Algoritması (Back-Propagation)

İlk olarak 1986 yılında [119] tarafından tanıtılan geri yayılım algoritması, YSA öğrenme modelleri arasında en çok kullanılanlardandır. Genelleştirilmiş delta kuralı olarak da adlandırılan bu öğrenme kuralının uygulandığı ağlar, gözetimli öğrenme stratejisi kullanan, çok katmanlı, tam bağlantılı, ileri beslemeli ağlardır. Diğer bir deyişle, ağa verilen eğitim verisi hem girdi değerlerini hem de girdiye ilişkin hedef çıktı değerlerini içerir. Bir katmandaki her nöron alt katmandaki bütün nöronlara bağlanır. Geri besleme bağlantısı içermemekle birlikte aynı katmandaki nöronlar arasında da hiçbir bağlantı yoktur.

Geri yayılım öğrenme modeli, çıktı değerleri ile hedef değerlerin durumlarına göre hatayı çıkıştan geriye doğru azaltmaya çalıştığından dolayı bu ismi almıştır. Ağ çıkışında hesaplanan hata, ağırlıkların yeni değerlerinin hesaplanmasında kullanılmaktadır. p sayıda beklenen çıktının olduğu bir ağda, eğitimin k 'inci iterasyonunda YSA'nın çıkışındaki j 'inci nöronun \mathcal{S} aktivasyon fonksiyonu ile hesaplanmış çıkış değeri $\mathcal{S}(y_j^k)$, hedef çıktı d_j^k ve j nöronuna ilişkin hata e_j^k ile gösterilirse, hata değerinin hesaplanması Eşitlik 22 ile yapılır [120].

$$e_j^k = (d_j^k - \mathcal{S}(y_j^k)) \quad (22)$$

Geri yayılım algoritmasında hata tespiti için ölçüt olarak Eşitlik 23'teki hata kareler ortalaması kullanılır.

$$\mathcal{E}_k = \frac{1}{2} \sum_{j=1}^p (d_j^k - \mathcal{S}(y_j^k))^2 \quad (23)$$

Hesaplanan ve hedef değerler arasındaki fark, her bir çıktı nöronu için bir hata sinyali olarak hesaplanır. Hesaplanan hata sinyalleri, her çıktı nöronuna karşı gelen ara katmandaki nöronlara aktarılır. Böylece, ara katmandaki nöronların her biri toplam hatanın sadece hesaplanan bir kısmını içerir. Bu süreç her katmandaki nöronlar toplam hatanın belirli bir kısmını içerecek şekilde giriş katmanına kadar tekrarlanır. Elde edilen hata sinyalleri temel alınarak, bağlantı ağırlıkları her nöronda yeniden düzenlenir. Bu düzenleme tüm verilerin kodlanabileceği bir duruma ağın yakınsamasını sağlar [121].

v_{ih}^k ile k 'inci iterasyonda i 'inci ($i = 1, \dots, n$) girdi katmanı nöronunu, h 'inci ($h = 1, \dots, q$) gizli katman nöronuna bağlayan ağırlık değeri; w_{hj}^k ile k 'inci iterasyonda h 'inci

gizli katman nöronunu, j 'inci ($j = 1, \dots, p$) çıktı katmanı nöronuna bağlayan ağırlık değeri; η ile öğrenim hızı gösterilsin. Bu durumda ağırlık değerlerinin değişme miktarı, çıktı katmanı ile gizli katmanlar arasındaki nöronlar için gradyan açılma yöntemi olarak adlandırılan Eşitlik 24'ten, gizli katman ile gizli katman veya gizli katman ile girdi katmanı arasındaki nöronlar için ise Eşitlik 25'ten yola çıkılarak bulunur.

$$\Delta v_{ih}^k = -\eta \left(\frac{\partial \mathcal{E}_k}{\partial w_{ih}^k} \right), \quad i = 1, \dots, n; \quad h = 1, \dots, q \quad (24)$$

$$\Delta w_{hj}^k = -\eta \left(\frac{\partial \mathcal{E}_k}{\partial w_{hj}^k} \right), \quad h = 1, \dots, q; \quad j = 1, \dots, p \quad (25)$$

Geri yayılım algoritmasında her iterasyon, ileri yayılım ve geri yayılım olmak üzere iki aşamadan oluşmaktadır. İleri yayılım aşamasında YSA'nın o andaki durumunda YSA'ya uygulanan giriş işaretlerine karşı YSA'nın çıkışlarında oluşan değerler bulunur. Geri yayılım aşamasında, çıkışlarda oluşan hatalardan yola çıkılarak devredeki ağırlıkların yeniden düzenlenmesi yapılmaktadır [122]. x_i ile girdi nöronu, z_h ile gizli nöron, y_j ile çıktı nöronu gösterilirse, YSA'daki ağırlık değişimleri Eşitlik 26 ve Eşitlik 27 kullanılarak yapılmaktadır. Bu düzenlemeler yanlılık terimleri için de gerçekleştirilmektedir.

$$\Delta w_{hj}^k = \eta \delta_j^k \mathcal{S}(z_h^k), \quad \delta_j^k = (d_j^k - \mathcal{S}(y_j^k)) \mathcal{S}'(y_j^k), \quad h = 1, \dots, q; \quad j = 1, \dots, p \quad (26)$$

$$\Delta v_{ih}^k = \eta \delta_h^k x_i^k, \quad \delta_h^k = \left(\sum_{j=1}^p \delta_j^k w_{hj}^k \right) \mathcal{S}'(z_h^k), \quad i = 1, \dots, n; \quad h = 1, \dots, q \quad (27)$$

α momentum katsayısının da eklenmesiyle ağırlık katsayılarının ve yanlılık değerlerinin güncelleştirilmesi Eşitlik 28 ve Eşitlik 29 kullanılarak yapılır.

$$v_{ih}^{k+1} = v_{ih}^k + \Delta v_{ih}^k + \alpha \Delta v_{ih}^{k-1}, \quad i = 0, \dots, n; \quad h = 1, \dots, q \quad (28)$$

$$w_{hj}^{k+1} = w_{hj}^k + \Delta w_{hj}^k + \alpha \Delta w_{hj}^{k-1}, \quad h = 0, \dots, q; \quad j = 1, \dots, p \quad (29)$$

Tablo 7. Geri yayılım algoritması adımları [123]

Verilen	$X_k \in \mathbb{R}^n$ vektörlerinden oluşan bir \mathcal{T} eğitim setini ve $D_k \in \mathbb{R}^p$ beklenen çıktı vektörlerini içeren $n - q - p$ mimarisine sahip \mathcal{N} sinir ağı
Adım 0	Ağırlıklara (v_{ih}^1) rastgele küçük değerler atanır, $\Delta v_{ih}^0 = 0$, $i = 0, \dots, n$; $h = 1, \dots, q$ Ağırlıklara (w_{hj}^1) rastgele küçük değerler atanır, $\Delta w_{hj}^0 = 0$, $h = 0, \dots, q$; $j = 1, \dots, p$ $k = 1$; η öğrenim hızı, α momentum katsayısı, τ hata tolerans değerleri atanır.
Adım 1	Tüm girdi eğitim vektörü ve hedef çıkış çiftleri için Adım 2-9 arasındaki işlemleri durdurma kriteri sağlanana kadar gerçekleştirilir.
Adım 2	Her bir eğitim verisi çifti $(X_k, D_k) \in \mathcal{T}$ için Adım 3-8 takip edilir.
İleri besleme (Feedforward)	
Adım 3	Her bir girdi nöronu ($X_i, i = 1, \dots, n$) x_i girdi sinyalini alır ve bu sinyali bir üst katmandaki tüm nöronlara (gizli katmandaki nöronlar) aktarır.
Adım 4	Her bir gizli nöron ($Z_h, h = 1, \dots, q$), ağırlıklandırılmış girdi sinyallerini toplar, $z_h^k = v_{0h}^k + \sum_{i=1}^n x_i^k v_{ih}^k, \quad h = 1, \dots, q$ çıkış sinyalini hesaplamak için aktivasyon fonksiyonunu kullanır, $\mathcal{S}(z_h^k) = \frac{1}{1 + e^{-z_h^k}}, \quad h = 1, \dots, q$ ve bu sinyali üst katmandaki tüm birimlere (çıkış katmanındaki nöronlar) iletir.
Adım 5	Her bir çıkış nöronu ($Y_j, j = 1, \dots, p$), ağırlıklandırılmış girdi sinyallerini toplar, $y_j^k = w_{0j}^k + \sum_{h=1}^q \mathcal{S}(z_h^k) w_{hj}^k, \quad j = 1, \dots, p$ çıkış sinyalini hesaplamak için aktivasyon fonksiyonunu kullanır. $\mathcal{S}(y_j^k) = \frac{1}{1 + e^{-y_j^k}}, \quad j = 1, \dots, p$
Hatanın geri yayılımı (Backpropagation of error)	
Adım 6	Her bir çıkış nöronu ($Y_j, j = 1, \dots, p$), girdi eğitim verisine karşılık gelen hedef çıktı değerini kullanarak hata terimini hesaplar. $\delta_j^k = (d_j^k - \mathcal{S}(y_j^k)) \mathcal{S}'(y_j^k), \quad j = 1, \dots, p$ Sonra, w_{hj} katsayısını güncellemede kullanılacak ağırlık düzeltme terimini hesaplar. $\Delta w_{hj}^k = \eta \delta_j^k \mathcal{S}(z_h^k), \quad h = 1, \dots, q; \quad j = 1, \dots, p$ Daha sonra, w_{0j} yanlılık değerini güncellemede kullanılacak olan yanlılık düzeltme terimini hesaplar ve δ_j^k 'leri alt katmandaki nöronlara iletir. $\Delta w_{0j}^k = \eta \delta_j^k, \quad j = 1, \dots, p$
Adım 7	Her bir gizli nöron ($Z_h, h = 1, \dots, q$), yukarı katmandan gelen delta girdilerini toplar ve oluşan toplamı aktivasyon fonksiyonunun türevi ile çarpar. $\delta_h^k = \left(\sum_{j=1}^p \delta_j^k w_{hj}^k \right) \mathcal{S}'(z_h^k), \quad h = 1, \dots, q$ Daha sonra, v_{ih} katsayısını güncellemede kullanılacak ağırlık düzeltme terimini hesaplar. $\Delta v_{ih}^k = \eta \delta_h^k x_i^k, \quad i = 1, \dots, n; \quad h = 1, \dots, q$ Son olarak, v_{0h} yanlılık değerini güncellemede kullanılacak yanlılık düzeltme terimini hesaplar. $\Delta v_{0h}^k = \eta \delta_h^k, \quad h = 1, \dots, q$
Ağırlık katsayılarını ve yanlılık değerlerini güncelleştir	
Adım 8	Her bir çıkış nöronu ($Y_j, j = 1, \dots, p$) yanlılık değerlerini ve ağırlık katsayılarını ($h = 0, \dots, q$) günceller. $w_{hj}^{k+1} = w_{hj}^k + \Delta w_{hj}^k + \alpha \Delta w_{hj}^{k-1}, \quad h = 0, \dots, q; \quad j = 1, \dots, p$ Her bir gizli nöron ($Z_h, h = 1, \dots, q$) yanlılık değerlerini ve ağırlık katsayılarını ($i = 0, \dots, n$) günceller. $v_{ih}^{k+1} = v_{ih}^k + \Delta v_{ih}^k + \alpha \Delta v_{ih}^{k-1}, \quad i = 0, \dots, n; \quad h = 1, \dots, q$ Anlık hata hesaplanır. $\mathcal{E}_k = \frac{1}{2} \sum_{j=1}^p (d_j^k - \mathcal{S}(y_j^k))^2, \quad j = 1, \dots, p$
Adım 9	Durdurma kriteri kontrol edilir. $\mathcal{E}_{ort} = \frac{1}{Q} \sum_{k=1}^Q \mathcal{E}_k < \tau$

Tüm girdi eğitim vektörü ve hedef çıktı çiftleri için ileri besleme ve hatanın geri yayılımı adımları durdurma kriteri sağlanana kadar gerçekleştirilir. Belirli bir iterasyon sayısına ulaşıncaya veya ortalama hata belirli bir tolerans değerinin altına düşünce eğitimi durdurmak en çok kullanılan iki durdurma kriteridir. Geri yayılım algoritması tüm adımlarıyla açıklamalı olarak Tablo 7’de verilmiştir. Algoritmaya eklenen momentum katsayısı, hatanın en küçük olduğu noktayı bulmaya, doğrultunun ayarlanmasına ve geri yayılım algoritmasının öğrenme hızının artırılmasına yardımcı olmaktadır.

Eşlenik gradyan veya quasi newton gibi yöntemler gradyan azaltma (gradient descent) yönteminden daha çabuk yakınsar ancak bu yöntemlerde genellikle hata yüzeyinin karesel bir fonksiyonla modellenebileceği varsayılır, bu varsayımın tutmadığı durumlarda bu yöntemler çok başarılı sonuçlar üretmez [124; 125]. Yavaş olması ve basit bir problemin çözümünde bile YSA’nın eğitiminin binlerce iterasyon gerektirmesi geriye yayılım algoritmasının dezavantajlarıdır. Bu algoritmanın başarısı ağırlık katsayılarının ilk değerlerine, momentum ve öğrenme katsayısı değerlerinin seçimine bağlıdır.

2.3.2. Karar Ağaçları

Bir karar ağacı, kök düğüm adı verilen bir değişkenden başlayan ağaç benzeri bir yapıda hiyerarşik bir ilişki grubudur. Hem sınıflandırma hem de regresyon problemlerinin çözümü için kullanılabilir. Karar ağacının yapısında her bir düğüm bir özneliği temsil eder. Dallar ve yapraklar ağaç yapısının elemanlarıdır. En üstteki eleman kök, en alttaki eleman yaprak ve bunların arasında kalan elemanlar ise dal olarak isimlendirilir. İç karar düğümleri giriş verilerinin test edildiği, soruların sorulduğu ve hangi yöne yönelmesi gerektiğine karar verilen düğümlerdir. Dallar bu soruların cevaplarını temsil eder. Uç yapraklar ise kategorinin bulunduğu sınıf etiketlerini belirtir [126]. Karar ağacında kurallar kökten yaprığa doğru inilerek (IF-THEN rules) yazılır. Eğitim verileri ve bu verilere ait öznelik bilgilerinden yararlanılarak bir karar ağacı yapısı oluşumunda temel prensip verilere ilişkin bir dizi sorular sorulması ve elde edilen cevaplar doğrultusunda karar kuralları oluşturulmasıdır. Ağacın ilk düğümü olan kök düğümünde verilerin sınıflandırılması ve ağaç yapısının oluşturulması için sorular sorulmaya başlanır ve dalları olmayan düğümler ya da yapraklara gelene kadar bu işlem devam eder [127; 128].

Karar ağaçlarında en önemli sorunlardan birisi bölümlenmenin veya dallanmanın hangi kökten itibaren hangi kritere göre yapılacağıdır. Ele alınan kritere göre karar ağacı

algoritması da değişiklik göstermektedir. Literatürde kullanılan başlıca karar ağacı algoritmaları şu şekilde sıralanabilir: CHAID (Otomatik Ki-Kare Etkileşim Belirleme (Chi-Squared Automatic Interaction Detector)), CART (Sınıflandırma ve Regresyon Ağaçları (Classification and Regression Trees)), ID3, Kapsamlı (Exhaustive) CHAID, C4.5, MARS (Çok Değişkenli Uyarlanabilir Regresyon Uzanımları (Multivariate Adaptive Regression Splines)), QUEST (Hızlı-Yansız-Etkin İstatistiksel Ağaç (Quick, Unbiased, Efficient Statistical Tree)), C5.0, SLIQ (QUEST Algoritmasında Gözetimli Öğrenme (Supervised Learning in QUEST)), SPRINT (Ölçeklenebilir, Parellenebilir Tümevarım Karar Ağacı (Scalable Parallelizable Induction of Decision Trees))

Karar ağaçlarının dallara bölünmesinde kullanılan başlıca kriterler şu şekildedir:

- Bilgi kazancı (information gain): Bu kriterde tüm özniteliklerin entropileri hesaplanır ve en düşük entropiye sahip öznitelik bölünme için seçilir. Bu metot çok sayıda değer içeren öznitelikleri seçme yanlısıdır.
- Kazanç oranı (gain ratio): Bilgi kazancının bir versiyonu olan bu kriter, öznitelik seçimi yaparken dal sayısı ve büyüklüklerini de hesaba katarak bilgi kazancındaki yanlılığı yok etmeye çalışır.
- Gini indeksi (gini index): Etiket karakteristiklerinin dağılımındaki eşitsizliği en aza indirmek için kullanılan bir ölçüttür. Belirli bir değişkenin rastgele seçildiğinde yanlış sınıflandırılmasının derecesini veya olasılığını ölçer. En düşük gini indeksini veren ayrıma sahip öznitelik bölünme için seçilir.
- Doğruluk (accuracy): Tüm ağacın doğruluğunu maksimize eden öznitelik bölünme için seçilir.

Bu ölçütlerin dışında olabirlik-oranı (likelihood ratio) ki-kare istatistikleri, Twoing ölçütü, DKM (Dietterich-Kearns-Mansour) kriteri ve Kolmogorov-Smirnov ölçütü gibi kriterler de kullanılmaktadır.

Karar ağaçlarında aşırı uyum (overfitting) problemini engellemek ve yanlış sınıflandırma hatasını minimize etmek için kullanılan erken budama (pre-pruning) ve geç budama (post-pruning) işlemi olarak adlandırılan iki yöntem vardır. Budama işlemi, ağacın gerçek büyüklüğüne karar vermede kullanılır. Erken budamada bir düğüme ulaşan örnek sayısı, eğitim verilerinin belirli bir yüzdesinden daha küçükse o düğüm artık bölünmez. Hızlı bir çözüm sunmasına rağmen, az sayıda örneğe bağlı olarak alınan kararlar genelleme hatasını artırabilmektedir. Geç budamada ağacın veriye oturmasına izin verilir [129]. Ağaçta

aşırı öğrenme problemi çapraz doğrulama verileri ile test edilir ve sonuca göre budama gerçekleştirilir [130].

2.3.3. Rastgele Orman Sınıflandırıcısı

Toplu sınıflandırma (ensemble classification) yöntemleri, bir sınıflandırıcı yerine birden çok sınıflandırıcı üreten ve sonrasında onların tahminlerinden alınan oylar ile yeni veriyi sınıflandıran öğrenme algoritmalarıdır. En yaygın olarak kullanılan toplu sınıflandırıcılar; Torbalama (Bagging), Hızlandırma (Boosting) ve Rastgele Orman (Random Forest) yöntemleridir [131].

Torbalama algoritmasında, bir sınıflandırıcıyı eğitmek için yer değiştirmeli olarak orijinal eğitim veri setinden birden çok önyüklemeli eğitim veri setleri oluşturulur ve her bir önyüklemeli eğitim veri seti için bir ağaç tasarlanır. Art arda gelen ağaçlar bir öncekinden bağımsız bir şekilde davranır ve tahmin etme amacı için en büyük oy baz alınır [132]. Diğer taraftan, Hızlandırma iteratif olarak tekrarlı eğitimi kullanır ve yanlış sınıflandırılmış örneklerin ağırlıkları, önemlerini yükseltmek için bir sonraki iterasyonda artırılır. Hızlandırma genel olarak sınıflandırmanın hem varyansını hem de sapmasını azaltır ve çoğu durumda Torbalama'dan daha doğru sonuç verir. Ancak yavaşlık ve gürültü hassasiyeti gibi dezavantajlara sahiptir [133].

Rastgele Orman, ağaç tipi sınıflandırıcılar topluluğu olarak tanımlanabilir. Torbalama yönteminin rastgelelik özelliği eklenmiş bir versiyonudur [134]. Rastgele Orman, tüm değişkenler arasından en iyi dalı kullanarak her bir düğümü dallara ayırmak yerine, her bir düğümde rastgele olarak seçilen değişkenler arasından en iyisini kullanarak her bir düğümü dallara ayırır. Her bir veri seti orijinal veri setinden yer değiştirmeli olarak üretilir. Sonra rastgele öznitelik seçimi kullanılarak ağaçlar geliştirilir. Geliştirilen ağaçlar budanmaz [134; 135].

Rastgele orman algoritmasında, diğer karar ağaçlarına benzer şekilde, dallanma kriterlerinin belirlenmesi ve uygun bir budama yönteminin seçilmesi önemlidir. Dallanma kriterlerinin belirlenmesinde Gini indeksi kullanılmaktadır [136].

Rastgele Orman algoritmasını başlatmak için kullanıcı tarafından 2 parametre tanımlanmalıdır. Bu parametreler, en iyi bölünmeyi belirlemek için her bir düğümde kullanılan değişkenlerin sayısı (m) ve geliştirilecek ağaçların sayısı N 'dir.

Rastgele Orman yönteminde, üretilen modeli test etmek için ayrı bir veri seti yoksa ya da orijinal veri setinden test veri seti ayrılmamış ise, sınıf dağılımı dikkate alınmak suretiyle orijinal veri setinin $2/3$ 'ü eğitim verisi (inBag), $1/3$ 'ü ise test verisi (Out-Of-Bag (OOB)) olarak ayrılır. Eğer test verisi mevcut ise ya da orijinal veri setinden test için veri ayrılmış ise eğitim verilerinin $2/3$ 'ü (inBag) eğitim için, $1/3$ 'ü (OOB) ise test için kullanılır. Bu yöntemde karar ormanını oluşturulmasında kaç karar ağacı kullanılacaksa, o sayıda bootstrap tekniği kullanılarak örneklem oluşturulur ve her bir örneklem için inBag ve OOB verisi ayrılır. Tüm ağaçlar ayrılan OOB verisi ile test edilerek hata oranı hesaplanır ve ardından bu hata oranlarının ortalaması alınarak karar ormanının OOB hatası bulunur. Hesaplanan OOB hata oranına göre tüm ağaçlara bir ağırlık verilir. Hata oranı en yüksek olan karar ağacı en düşük ağırlığı, hata oranı en düşük olan karar ağacı ise en yüksek ağırlığı alır. Belirlenen ağırlıklara göre tüm ağaçlar sınıflandırma işlemi için bir oylama işlemine tabi tutulurlar. En çok oyu alan sınıf nihai sınıf tahmini olarak belirlenmiş olur [137].

2.3.4. Gradyan Artırma Karar Ağacı

Friedman tarafından geliştirilen Gradyan Artırma Karar Ağacı (Gradient Boosted Tree-GBT) algoritması, geleneksel açgözlü (greedy) ağaç tabanlı yöntemlerden daha yüksek tahmin doğruluğu elde edebilmek amacıyla artırma (boosting) ve gradyan iniş (gradient descent) fikirlerini kullanır [138].

Rastgele Orman algoritmasında bağımsız karar ağaçları oluşturulmakta ve sadece bu ağaçların ürettiği tahminlerin ortalaması nihai tahmin olarak alınmaktadır. Öte yandan, gradyan artırma karar ağacı algoritmasında ise iteratif bir yaklaşım benimsenmiştir. Ağaçlar üretilir ve sıralı bir şekilde topluluğa eklenir. İlk ağaç, Rastgele Ormanlarda yapıldığı gibi üretilir. Buradaki temel fark, ilk ağacın ürettiği tahmin hatalarını en aza indirmeyi amaçlayan ikinci ağacın üretilmesidir. Hem birinci hem de ikinci ağaçlar topluluğa eklenir ancak her birine farklı ağırlıklar verilir. Bu süreç birden çok kez tekrarlanır: her adımda, öğrenilen tüm topluluğun hatalarına göre yeni bir ağaç eğitilir ve daha sonra topluluğa eklenir. Son topluluk, yeni girdilerin sonucunu kestirmek için bir model olarak kullanılır [139].

2.3.5. Naive Bayes Sınıflandırıcısı

Naive Bayes sınıflandırıcı her bir özneliğin istatistiksel açıdan bağımsız olduğu varsayımını kullanan olasılık tabanlı bir sınıflandırma işlemidir. Bu varsayımda, bir

durumun meydana gelmesi, diğer durumların meydana gelmesini etkilememektedir. Bu kabullenme birçok problemde doğru olmamasına rağmen, Naive Bayes sınıflandırıcısı uygulamada çok iyi sonuçlar verebilmektedir [140].

Birbirinden bağımsız ve rastgele iki olayın (X ve Y) birbiri ardı sıra gerçekleştiği durumlarda bu iki olaydan birinin gerçekleşmesi durumunda ikinci olayın gerçekleşme olasılığı $P(X \cap Y)$ ifadesi ile gösterilebilir. Değişme özelliği sayesinde çarpım kuralı Eşitlik 30'daki gibi iki farklı ifade ile yazılabilir.

$$P(X \cap Y) = P(X|Y)P(Y) = P(Y|X)P(X) \quad (30)$$

Bayes teoremi, rassal bir sürece bağlı olarak ortaya çıkan rasgele bir X olayı ile diğer bir rastgele Y olayı için koşullu olasılıklar ve marjinal olasılıklar arasındaki ilişkiyi tanımlar. Bu ilişkiyi ilk kez 17. yüzyılda yaşamış olan İngiliz matematikçi Thomas Bayes ortaya atmıştır. $P(X)$ ve $P(Y)$ ile X ve Y olaylarının marjinal olasılıkları, $P(X|Y)$ ile Y olayının gerçekleştiği bilindiğinde X olayının gerçekleşme olasılığı gösterilirse, Bayes teoremi Eşitlik 31'deki gibi yazılabilir.

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \quad (31)$$

$X = \{x_1, x_2, \dots, x_n\}$, sınıf üyeliği bilinmeyen veri örneğini; C_i ($i = 1, \dots, q$), X 'in ait olabileceği sınıfları; $P(C_i|X)$, X 'in C_i sınıfında olma olasılığını (sonsal olasılık); $P(C_i)$, C_i sınıfının gözlem yapılmadan önceki ilk olasılığını (önsel olasılık); $P(X)$, X girdi vektörünün gözlemlenmesi olasılığını; $P(X|C_i)$, C_i sınıfı bilindiğinde X 'in gözlemlenmesi olasılığını gösterdiğinde sınıflandırma problemi için Bayes kuralı Eşitlik 32'deki gibi yazılabilir [141].

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \quad (32)$$

Hesaplamalardaki işlem yükünü azaltmak için, $P(X|C_i)$ olasılığını basitleştirmek amacıyla her örneğe ait x_i değerlerinin birbirinden bağımsız olduğu kabul edilerek Eşitlik 33 elde edilir [142].

$$P(X \setminus C_i) = \prod_{k=1}^n P(X_k \setminus C_i) \quad (33)$$

Bilinmeyen örnek X 'i sınıflandırmak için Eşitlik 32'de $P(C_i \setminus X)$ içinde yer alan paydalar birbirine eşit olduğundan sadece pay değerlerinin karşılaştırılması yeterlidir. Bu değerler içinden en büyük olanı Eşitlik 34 ile seçilerek bilinmeyen örüntünün bu sınıfa ait olduğu belirlenir.

$$\arg \max_{C_i} \{P(X \setminus C_i)P(C_i)\} \quad (34)$$

Sonsal olasılıkları kullanan Eşitlik 34, En Büyük Sonsal Sınıflandırma Yöntemi (Maximum A Posteriori Classification-MAP) olarak da bilinir. Eşitlik 33'ten dolayı Bayes sınıflandırıcısı olarak Eşitlik 35 kullanılabilir.

$$C_{MAP} = \arg \max_{C_i} \left\{ \prod_{k=1}^n P(X_k \setminus C_i) \right\} \quad (35)$$

Bayes sınıflandırıcıda nitelik değerleri sayısal olduğunda, koşullu olasılıkları modellemek için sayısal verilerin dağılımının normal dağıldığı varsayılarak Eşitlik 36'daki olasılık yoğunluk fonksiyonu kullanılır. Eşitlikte μ_{C_i} ortalama, σ_{C_i} ise standart sapma değerleridir.

$$P(x_k \setminus C_i) = f(x_k, \mu_{C_i}, \sigma_{C_i}) = \frac{1}{\sqrt{2\pi\sigma_{C_i}}} e^{-\frac{(x_k - \mu_{C_i})^2}{2\sigma_{C_i}^2}} \quad (36)$$

2.3.6. k-En Yakın Komşu Sınıflandırıcısı

k-En yakın komşu algoritması (kNN), en temel örnek tabanlı öğrenme algoritmaları arasındadır. Örnek tabanlı öğrenme algoritmalarında, öğrenme işlemi eğitim setinde tutulan verilere dayalı olarak gerçekleştirilmektedir. Yeni karşılaşılan bir örnek, eğitim setinde yer alan örnekler ile arasındaki benzerliğe göre sınıflandırılmaktadır [143]. Bu algoritma verilen k değeri kadar en yakın komşunun sınıfına göre sınıflandırma işlemi yapmaktadır. kNN algoritmasında bir vektörün sınıflandırılması, sınıfı bilinen vektörler kullanılarak yapılmaktadır. Test edilecek örnek, eğitim kümesindeki her bir örnek ile tek tek işleme alınır.

Test edilecek örneğin sınıfını belirlemek için eğitim kümesindeki o örneğe en yakın k adet örnek seçilir. Seçilen örneklerden oluşan küme içerisinde hangi sınıfa ait en çok örnek varsa test edilecek olan örnek bu sınıfa aittir denilir [144].

kNN algoritmasının performansı için büyük öneme sahip noktalardan birisi örnekler arası yakınlığın nasıl ölçümleneceğidir. Öklid (Euclidean), Manhattan, Mahalanobis ve Chebyshev yöntemleri uzaklık hesaplamasında kullanılan başlıca yöntemlerdir [143].

kNN algoritması, eğitim verilerinin çok olması durumunda etkili sonuçlar verebilmektedir. Basit ve kolay anlaşılır bir yapıda olması ve gürültüye sahip öğrenme verilerinde de güçlü sonuçlar ortaya çıkarabilmesi kNN'nin en önemli avantajlarıdır. k sayısının ne olacağına ve hangi uzaklık ölçütünün kullanılacağına dair belirsizliğin olması ve büyük veri setlerinde test edilen verinin eğitim kümesindeki tüm örneklerle karşılaştırılacak olması nedeniyle sınıflandırma işleminin hesaplama zamanının oldukça uzun olması kNN'nin en önemli dezavantajlarıdır [145].

2.3.7. Genelleştirilmiş Lineer Modeller

Genelleştirilmiş Lineer Modeller (Generalized Linear Models-GLM), alışlagelen normal kurama dayalı doğrusal regresyon modelleri ile lojistik ve Poisson regresyonu gibi doğrusal olmayan modelleri birleştirerek regresyon ve deneysel tasarım modellerini bir araya getiren bir yaklaşımdır [146].

GLM'de anahtar varsayım, yanıt değişkeninin dağılımının normal, binom, Poisson, ters normal, üstel ve gamma dağılımlarını içeren üstel dağılımlar ailesinin bir üyesi olmasıdır. Üstel ailenin üyeleri olan dağılımlar Eşitlik 37'deki genel biçime sahiptirler.

$$f(y_i, \theta_i, \phi) = e^{\frac{y_i \theta_i - b(\theta_i)}{a(\phi)} + h(y_i, \phi)} \quad (37)$$

Eşitlik 37'de, ϕ ölçek parametresi, θ_i ise doğal konum parametresidir. Üstel ailenin üyeleri için beklenen değer Eşitlik 38, varyans ise Eşitlik 39 ile bulunur.

$$E(Y) = \mu = b'(\theta) = \frac{db(\theta_i)}{d(\theta_i)} \quad (38)$$

$$Var(Y) = b''(\theta)a(\phi) = \frac{d^2b(\theta_i)}{d(\theta_i^2)}a(\phi) = \frac{d\mu}{d\theta_i}a(\phi) \quad (39)$$

Eşitlik 40 incelendiğinde $Var(\mu)$, yanıtın varyansının ortalamaya bağımlılığını gösterir. Bu, normal dağılım dışındaki üstel ailenin üyesi olan tüm dağılımların bir özelliğidir [146].

$$Var(\mu) = \frac{Var(Y)}{a(\phi)} = \frac{d\mu}{d\theta_i} \quad (40)$$

GLM'de temel görüş yanıt değişkeninin beklenen değerinin uygun bir fonksiyonu için doğrusal bir model gerçekleştirmektir. Genel olarak GLM üç bileşenden oluşmaktadır. Bu bileşenler; yanıtın dağılımı, lineer kestiricilerin bulunduğu sistematik kısım ve link fonksiyonudur. Yapının daha açık biçimi aşağıdaki gibi verilebilir [147]:

- 1) y_1, y_2, \dots, y_n sırasıyla $\mu_1, \mu_2, \dots, \mu_n$ ortalamalı bağımsız yanıt gözlemleridir.
- 2) y_i gözlemleri üstel aile üyesi olan bir dağılıma sahiptir.
- 3) Modelin sistematik kısmı x_1, x_2, \dots, x_k açıklayıcı değişkenlerini içermektedir.
- 4) Model $\eta = x'\beta = \beta_0 + \sum_{i=1}^k \beta_i x_i$ lineer kestirici civarında inşa edilir.
- 5) Model bir link (bağ) fonksiyonu aracılığı ile bulunur (Eşitlik 41).

$$\eta_i = g(\mu_i), \quad i = 1, 2, \dots, n \quad (41)$$

Link terimi, ortalama ve lineer kestirici arasında bağlayıcı fonksiyon gerçeğinden türetilmiştir. Beklenen yanıt $E(y_i) = g^{-1}(\eta_i) = g^{-1}(x'\beta)$ şeklindedir. Çoklu lineer regresyonda model Eşitlik 42'deki gibi özel bir durumu göstermektedir. Yani $g(\mu_i) = \mu_i$ şeklindedir ve bu durumda birim link fonksiyonu kullanılmaktadır.

$$\mu_i = \eta_i = x_i'\beta, \quad i = 1, 2, \dots, n \quad (42)$$

- 6) Link fonksiyonu diferansiyellenebilen monoton bir fonksiyondur.
- 7) $\sigma_i^2 (i = 1, 2, \dots, n)$ varyansı, μ_i ortalamasının bir fonksiyonudur.

Link fonksiyonu için birçok seçim şekli mevcuttur. Kanonik ve kanonik olmayan olmak üzere iki tip link fonksiyonu bulunmaktadır. Eğer $\eta_i = \theta_i$ olacak şekilde seçilirse, bu durumda η_i 'ye kanonik link denir. Tablo 8'de bazı dağılımların kanonik link fonksiyonları gösterilmektedir.

GLM ile birlikte kullanılabilirler diğerk link fonksiyonları Őu Őekildedir:

- 1) Φ ile birikimli standart normal dağılım fonksiyonu gösterilirse Probit link EŐitlik 43'teki gibi olur.

$$\eta_i = \Phi^{-1}[E(y_i)] \quad (43)$$

- 2) Tamamlayıcı (complimentary) log-log link EŐitlik 44'teki gibidir.

$$\eta_i = \ln \{ \ln [1 - E(y_i)] \} \quad (44)$$

- 3) Güç ailesi linki (power family link) EŐitlik 45'teki gibidir.

$$\eta_i = \begin{cases} (E(y_i))^\lambda, & \lambda \neq 0 \\ \ln[E(y_i)], & \lambda = 0 \end{cases} \quad (45)$$

GenelleŐtirilmiŐ lineer modellerde link fonksiyonu bağımlı değıŐkenin dağılımının taŐıdıđı özellikleri üstlenmektedir. Probit ve tamamlayıcı log-log link fonksiyonları sadece Bernoulli ve Binom dağılımları ile birlikte kullanılmaktadır. Özellikle link fonksiyonun yanlış seğıimi ile yapılan analizler yanlış sonuçlara götürmektedir. Link fonksiyonun belirlenmesinde genellikle EŐitlik 45 ile verilen güç ailesi link fonksiyonları kullanılmaktadır. GLM hakkında daha ayrıntılı bilgiye Nelder ve Wedderburn [148], McCullagh ve Nelder [149], Myers ve diğerklerinin [147] yayınlamıŐ olduđu kitaplardan ulaŐılabilir.

Tablo 8. GLM için kanonik bağılar

Dağılım	Kanonik Link
Normal	$\eta_i = \mu_i$ (Özdeş link)
Binom	$\eta_i = \ln \left(\frac{\pi_i}{1-\pi_i} \right)$ (Lojistik link)
Poisson	$\eta_i = \ln(\lambda)$ (Log link)
Üstel	$\eta_i = \frac{1}{\lambda_i}$ (Ters bağı)
Gamma	$\eta_i = \frac{1}{\lambda_i}$ (Ters bağı)

En çok olabilirlik yöntemi, GLM'deki parametre kestirimi için kuramsal temeldir. Bununla birlikte, algoritmadaki en çok olabilirlik sonuçlarının gerçek uygulaması İteratif Olarak Yeniden Ağırlıklandırılmış En Küçük Kareler (Iteratively Reweighted Least Squares-IRLS) algoritmasına dayanır. Eğer $\hat{\beta}$, IRLS algoritmasından elde edilen regresyon katsayısının son değeri ise ve eğer link fonksiyonunun seçimini içeren model varsayımları doğru ise o zaman Eşitlik 46'daki ifadeler asimtotik olarak gösterilebilir. Eşitlikte V matrisi, $a(\phi)$ dışında, lineer yanıt fonksiyonunda kestirilen parametrelerin varyanslarından oluşan köşegen matristir [146].

$$E(\hat{\beta}) = \beta, \quad Var(\hat{\beta}) = a(\phi)(X'VX)^{-1} \quad (46)$$

2.3.8. Otomatik Çok Katmanlı Algılayıcı

Bölüm 2.3.1.2'de detaylı şekilde anlatılan Çok Katmanlı Algılayıcı (Multi-Layer Perceptron-MLP) sınıflandırıcıları doğru bir şekilde eğitildiklerinde, genellikle iyi bir sınıflandırma performansına sahip, kompakt ve hızlı sınıflandırıcılar sunmalarından dolayı günümüzde yaygın olarak kullanılmaktadır. Ayrıca çok büyük eğitim setleri ve çok sayıda sınıfa da ölçeklenebilirler. Sahip olduğu bu özelliklerinden dolayı optik karakter tanıma işlemlerinde MLP kullanılabilir, fakat eğitim sürecinde performans artışı için el ile müdahale gereksinimi doğmaktadır. Literatürdeki mevcut öğrenme oranı ayarlama yöntemleri çok çeşitli sorunlara uygulandığında tutarsız ve öngörülemeyen performanslar gösterebilmektedir. Otomatik MLP (AutoMLP), sinir ağlarının eğitim sırasında ideal öğrenme oranını ve boyut ayarlamasını belirleyebilmek için 2010 yılında geliştirilmiş bir algoritmadır. Genetik algoritmalar [150; 151] ve stokastik optimizasyondan [152; 153] alınan fikirler bu algorithmada birleştirilmektedir. Algoritmanın işleyişi şu şekildedir: Önce farklı öğrenme oranları ve farklı sayıda gizli birim ile paralel olarak eğitilmiş küçük bir ağlar topluluğu oluşturulur. Önceden belirlenmiş az sayıdaki epok sonrasında, bir doğrulama seti kullanılarak hata oranı belirlenir ve en kötü performans gösteren ağların gizli birim sayısı ve öğrenme oranı, en iyi performans gösteren ağlarınkiyle değiştirilirler. Gizli birim sayıları ve öğrenme oranları, başarılı oranlardan ve boyutlardan elde edilen olasılık dağılımlarına göre belirlenirler.

2.3.9. Kural Türetme

Kural türetme (rule induction), makine öğrenmesinin en önemli tekniklerinden biridir. Verilerde saklı olan düzenlemeler kurallarla ifade edilebildiğinden, kural türetme aynı zamanda Veri Madenciliğinin de temel araçlarından biridir.

Kural türetme algoritmaları kurallardan oluşan bir sınıflandırma modeli üretir. Kuralların amacı, veri kümesi örneklerini sınıf etiketleriyle eşleştirerek sınıflandırmaktır. İki çeşit kural vardır: Önerme kuralları (propositional rules) ve birinci dereceden mantık kuralları (first order logic rules). Önerme kuralları, bağlanmış terimler ve kural tarafından kapsanan bir örneğe atanan bir sınıf etiketini içerir [154]. Terimler, girdi özneliği olan x_i 'nin kesikli ya da sürekli olmasına göre $x_i = v$, $x_i < \theta$ veya $x_i \geq \theta$ şeklindedir. x_0 sürekli ve x_1 kesikli özneliklerini ve iki önerme içeren C_1 sınıfı için örnek bir kural şu şekilde yazılabilir:

EĞER ($x_0 < 2$) **VE** ($x_1 = \text{"mavi"}$) **O ZAMAN** *Sınıf* = C_1

Bir kural kümesi, kural türetme algoritmaları tarafından türetilen kurallar listesidir. Örnek bir kural kümesi şu şekildedir:

Eğer ($\bar{O}zn_1 < 2$) **ve** ($\bar{O}zn_2 \geq 6$) **O zaman** *Sınıf* = C_1

Değilse

Eğer ($\bar{O}zn_1 < 5$) **O zaman** *Sınıf* = C_2

Değilse *Sınıf* = C_3

Bu örnekte, kural kümesi iki kural içerir ve her kural bir sınıfı sınıflandırır. Hiçbir kural tarafından kapsanmayan örnekler, varsayılan kural tarafından C_3 olarak ele alınmaktadır.

İki temel kural türetme algoritması grubu vardır: Ayır ve fethet (separate and conquer) algoritmaları (sıralı kapsama algoritmaları olarak da bilinir) ve böl ve fethet (ele al) (divide and conquer) algoritmaları. Ayır ve fethet algoritmaları [154], eğitim setinin ilgili kısmını açıklayan en iyi kuralı özyinelemeli bir şekilde bulur. Her bir iterasyon, bu kural tarafından kapsanan örnekleri eğitim setinden ayırır. Bu prosedür, hiçbir örnek kalmayana kadar devam eder.

Böl ve fethet algoritmaları (ağaç türetme algoritmaları), veri kazanımı, entropi, Gini indeksi vb. gibi önceden tanımlanmış bazı safsızlık (impurity) ölçütlerine göre verileri ayıran

en iyi ayrımı bulurlar. Örnekleri en iyi ayrıma göre böldükten sonra, elde edilen bölümleri özyinelemeli bir şekilde fethederler. C4.5, bilgi kazanımı açısından her düğümde en iyi bölünme ve en iyi özneliği arayan en çok bilinen ağaç türetme algoritmalarından biridir [155].

Ayır ve fethet algoritmaları bazı ortak işlemlere sahiptir. Her bir C_i sınıfı için başlangıç kuralı belirlenir ve arama yönüne göre koşullar kurula eklenir. Ardından, kural basitleştirilir ve bu basitleştirmeye ön budama (pre-pruning) denir. Ön budamadan sonra kural, arama stratejisine göre kural kümesine eklenir. Son olarak, oluşturulmuş kural kümesi basitleştirilmiştir ve bu basitleştirmeye geç budama (post-pruning) denir.

Sıralı kapsama algoritmaları sınıflandırma amaçlı kullanılan en yaygın kural türetme algoritmalarıdır. PRISM [156], PREPEND [157], AQ [158], CN2 [159] ve RIPPER bu algoritmalar arasında en bilinenleridir. Sunulan tezde yanık görüntülerinin sınıflandırılması için RIPPER algoritması tercih edilmiştir. Kurula şartlar eklenirken “bilgi kazancı” kriterinden yararlanılmıştır. Kural geliştirme ve budama süreçlerinde toplam verinin %90’ı eğitim verisi olarak kullanılmıştır. Hedeflenen saflık oranı %90 olarak alınmıştır.

2.3.9.1. Hata Azalması İçin Tekrarlanan Artımlı Budama Algoritması

Hata Azalması İçin Tekrarlanan Artımlı Budama (Repeated Incremental Pruning to Produce Error Reduction-RIPPER) algoritması, Fürnkranz ve Widmer’in geliştirmiş olduğu Artımlı Azaltılmış Hata Budama (Incremental Reduced Error Pruning-IREP) algoritmasının [160] Cohen tarafından iyileştirilmiş bir halidir [161]. Cohen, IREP kural türetme algoritmasının farklı problem kümeleri üzerindeki başarımını değerlendirmiş ve bu algoritmanın C4.5 algoritma kurallarına göre daha yüksek hata oranları ürettiğini göstermiştir. IREP algoritmasının eksikliklerini gidermek için geliştirilen RIPPER algoritmasında, budama fazında kullanılmak üzere alternatif bir metrik, kural setine yeni bir kural eklemenin ne zaman durdurulacağına karar vermek için yeni bir yöntem ve IREP tarafından öğrenilen kuralları optimize etmek için yeni bir teknik önerilmiştir. Bununla birlikte, RIPPER algoritması çoklu sınıfları (mümkün sınıf sayısının ikiden fazla olduğu durumlar) desteklemektedir [162].

RIPPER algoritması, kullanılan eğitim kayıtlarının sayısı ile doğrusal olarak ölçeklenebilmekte ve dengesiz sınıf dağılımına sahip modeller için de kullanılabilir. Ayrıca, modelin aşırı öğrenmesini önlemek amacıyla bir doğrulama seti kullanılmaktadır.

RIPPER sınıfları frekanslarına göre sıralar. (y_1, y_2, \dots, y_c) ile sınıf etiketleri gösterilirse ve y_1 en düşük frekansa, y_c en yüksek frekansa sahip sınıflar ise RIPPER önce y_1 dışında kalan tüm kayıtları negatif kayıt olarak nitelendirip y_1 için kurallar üretir. Sonrasında ise RIPPER y_2 için kural çıkarır. Bu işlem geriye sadece varsayılan (default) sınıf olarak etiketlenen y_c kalana kadar devam eder.

Kural büyümesi için, RIPPER genelden özele stratejisini kullanır. Başlangıçta her kural boştur, sonrasında seri olarak şartlar eklenerek kurallar oluşturulur. Kurala şartların eklenmesi için FOIL'in bilgi kazancı kriteri [163] kullanılır. $R: A \rightarrow p_0$ pozitif kayıt ve n_0 negatif kayıt kapsayan *sınıf* şeklinde bir kural olsun. Yeni bir B şartı eklendiğinde, kural $R': A \wedge B \rightarrow p_1$ pozitif kayıt ve n_1 negatif kayıt kapsayan *sınıf* şeklinde olsun. Bu durumda FOIL'in bilgi kazancı Eşitlik 47'deki gibi hesaplanır.

$$FOIL'in\ bilgi\ kazancı = p_1 \times \left(\log \frac{p_1}{p_1 + n_1} - \log \frac{p_0}{p_0 + n_0} \right) \quad (47)$$

Şartlar kural negatif örnekleri kapsamaya başlayana kadar eklenir. Sonrasında kural, doğrulama setindeki performansına göre budanır. Bunun için Eşitlik 48'deki metrik kullanılır. Eşitlikte p doğrulama setindeki kural tarafından kapsanan pozitif kayıtların sayısı, n ise doğrulama setindeki kural tarafından kapsanan negatif kayıtların sayısıdır. Eğer bu metrik bir şartın silinmesi sonrası artarsa, o zaman kural budanır.

$$Kural\ değer\ metriği = \frac{p - n}{p + n} \quad (48)$$

Bir kural oluşturulduktan sonra, kuralın kapsadığı tüm kayıtlar elimine edilir. Algoritma daha sonra yeni bir kural oluşturmaya devam eder. Kurallar, Minimum Açıklama Uzunluğu (Minimum Description Length-MDL) ilkesini ihlal etmediği ve doğrulama kümesindeki hata %50'den az olduğu sürece oluşturulur. RIPPER algoritması için daha ayrıntılı bilgiye [161; 162; 164; 165]'den ulaşılabilir.

2.4. Derin Öğrenme ve Semantik Segmentasyon

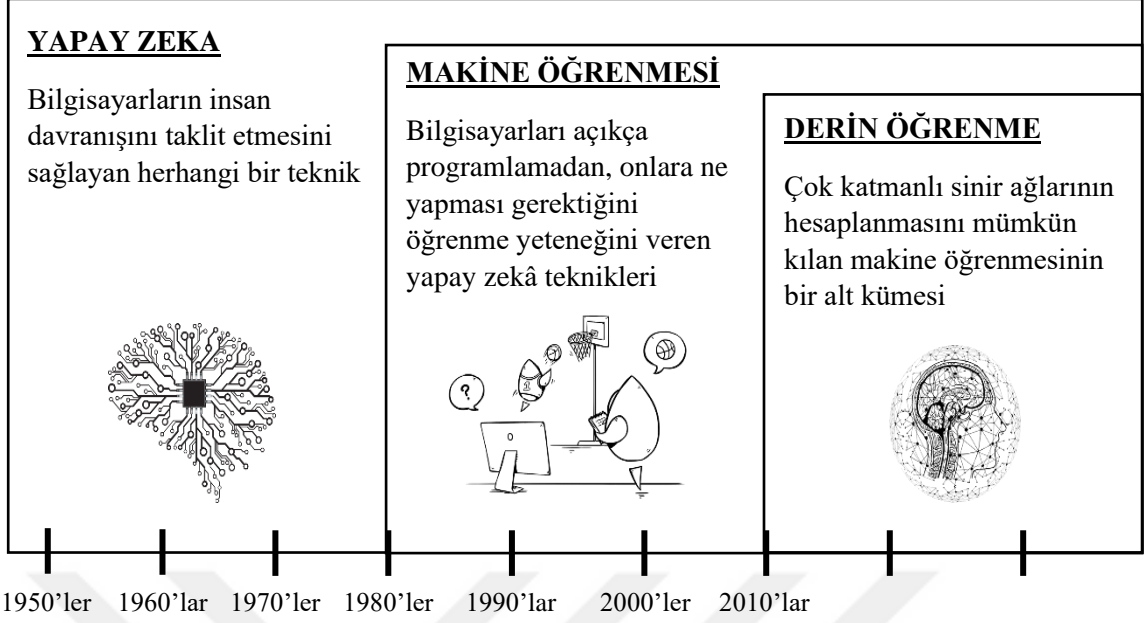
Tezin bu bölümünde, yanık görüntülerinin sınıflandırılması için önerilen ikinci yaklaşım olan Derin Öğrenme ve Semantik Segmentasyon yöntemleri ile ilgili teorik bilgiler verilmiştir.

2.4.1. Derin Öğrenme

Günümüzde neredeyse her gün duyar hale geldiğimiz “Yapay Zekâ (Artificial Intelligence)” terimi ilk olarak 1956 yılında bilgisayar bilimcisi John McCarthy ve arkadaşlarının organize ettiği “Yapay Zekâ Üzerine Dartmouth Yaz Araştırma Projesi” isimli bir çalıştayda detaylı olarak tanıtılmış ve literatürde bir alan olarak yerini almıştır [166]. Genel bir tanım yapmak gerekirse, yapay zekâ insan gibi düşünebilen bir makinedir. Bir başka deyişle, yapay zekâ, bilgisayar sistemlerinin insan zekâsına ilişkin süreçleri veya faaliyetleri yerine getirebilme kabiliyetidir. Bu süreçler arasında öğrenme (bilginin edinilmesi ve bilginin kullanımı için kurallar), muhakeme (yaklaşık veya kesin sonuçlara ulaşmak için kuralların kullanılması) ve kendini düzeltme yer almaktadır. Makineyle görme (machine vision), uzman sistemler (expert systems) ve konuşma tanıma (speech recognition) en popüler yapay zekâ uygulamaları arasında yer almaktadır [167].

Yapay zekâ kavramı çoğu zaman “Makine Öğrenmesi (Machine Learning)” kavramı ile karıştırılabilmektedir. Makine öğrenmesi, 1980’li yıllarda ortaya çıkan, makinelerin açıkça programlanmadan, sadece verileri kullanarak otomatik olarak öğrenebilmesine, doğru tahminler yapabilmesine ve deneyimlerine göre gelişebilmesine imkân sağlayan bir yapay zekâ uygulamasıdır. Bu kapsamda makine öğrenmesi, yapay zekânın bir alt dalı olarak görülmektedir.

Makine öğrenmesi algoritmaları, Gözetimli Öğrenme (Supervised Learning), Gözetimsiz Öğrenme (Unsupervised Learning) ve Takviyeli Öğrenme (Reinforcement Learning) ana başlıkları altında toplanmaktadırlar. Gözetimsiz öğrenme sistemleri, çıktı verilerini kullanmadan sadece girdi verileri üzerinden öğrenme işlemini gerçekleştirmektedir. Kümeleme ve temel bileşenler analizi bu öğrenmede kullanılan temel iki yöntemdir. Gözetimli öğrenme sistemleri ise girdi verileri ile etiketlenmiş çıktı verileri arasında eşleme yaparak öğrenme yapar. Bu öğrenmenin uygulamaları genellikle sınıflandırma ve regresyon olmak üzere iki kategoriye ayrılır. Takviyeli öğrenme, etkenin bulunduğu ortamı algılaması ve kendi başına aldığı kararlarla ortamda en yüksek ödül miktarına ulaşabilmek için hangi eylemleri yapması gerektiğini öğrenmesi ile ilgilenen bir yaklaşımdır. Takviyeli öğrenmede bir eğitmen bulunur, fakat gözetimli öğrenmedeki gibi sisteme çok bilgi vermez veya veremez. Sistem, doğru kararları için ödüllendirilirken, yanlış kararları için ise cezalandırılır. Bu tür öğrenme genellikle oyun kuramı, yöneylem araştırması, benzetim tabanlı en iyileme ve istatistik gibi alanlarda kullanılmaktadır [168].



Şekil 16. Yapay zekâ, makine öğrenmesi ve derin öğrenmenin zaman içerisindeki gelişimi

Makine öğrenmesinin bir alt kümesi olarak 2010'lu yıllarda ortaya çıkan Derin Öğrenme (Deep Learning) kavramı makinelerin dünyayı algılamada ve anlamada kullanılan bir yaklaşım biçimidir. Makine öğrenmesinde sınıflandırma için kullanılacak özelliklerin manuel olarak sisteme girilmesi gerekirken derin öğrenmede özellikler sistem tarafından otomatik olarak bulunabilmektedir [169]. Bununla birlikte, derin öğrenme sistemleri, makine öğrenmesi sistemlerine oranla yüksek donanımlı bilgisayarlara ve oldukça büyük miktarda eğitim verisine ihtiyaç duymaktadırlar. Makine öğrenmesi bu özelliklerin manuel olarak sağlanmasını gerektirir. Pek çok derin öğrenme metodu yapay sinir ağı mimarilerini kullanmaktadır. Klasik bir sinir ağı 2-3 katmandan oluşurken, derin öğrenmede gizli katman sayısı çok yüksekler çıkabilmektedir. Derin öğrenme ifadesindeki “derin” kelimesi de çok sayıda gizli katmana sahip olmayı ifade etmektedir. Günümüzde ses tanıma, yüz tanıma, sürücüsüz araç kullanımı, savunma sistemleri ve sağlık sektörü gibi birçok alanda derin öğrenmeden yararlanılmaktadır. Şekil 16’da yapay zekâ, makine öğrenmesi ve derin öğrenmenin zaman içerisindeki gelişimi gösterilmektedir.

Bugün itibarıyla, bilgisayar donanımlarında, özellikle GPU’da (Grafik İşleme Ünitesi (Graphics Processing Unit)) meydana gelen gelişmeler sayesinde çok fazla gizli katmandan oluşan yapay sinir ağlarının doğurduğu hesaplama maliyeti giderek düşmüştür. Bunun sonucunda farklı türlerde derin öğrenme mimarileri ortaya çıkmıştır. Literatürde en sık kullanılan mimariler CNN, Tekrarlayan Sinir Ağları (Recurrent Neural Network-RNN),

Uzun-Kısa Süreli Hafıza Ağları (Long-Short Term Memory-LSTM), Sınırlı Boltzman Makineleri (Restricted Boltzmann Machines-RBM) ve Derin Oto-Kodlayıcılar (Deep Autoencoders)'dır. Görüntü işlemede ve özellikle tıbbi uygulamalarda çok başarılı sonuçlar vermesi nedeniyle bu çalışmada CNN altyapısına sahip semantik segmentasyon yöntemleri kullanılmıştır. Bu yöntemler açıklanmadan önce CNN alt yapısının çalışma prensiplerine ilişkin detaylı bilgiler Bölüm 2.4.1.1'de, literatürde en çok kullanılan derin öğrenme tabanlı segmentasyon algoritmalarına ilişkin bir özet ise Tablo 9'da verilmiştir.

2.4.1.1. Konvolüsyonel Sinir Ağları (CNN)

CNN ağı ilk olarak Lecun ve arkadaşları tarafından [72; 170] yüksek boyutlu görüntü analizi için önerilmiştir. İnsanların görme sisteminden esinlenerek ortaya çıkan konvolüsyonel sinir ağları ile nesne tanımlama, görüntü sınıflandırma ve görüntü segmentasyonu gibi işlemler başarılı bir şekilde yapılabilmektedir. Bu ağların temel yapısı, bir veya daha fazla konvolüsyonel katman, alt örnekleme (subsampling) katmanı ve sonrasında gelen bir veya daha fazla tamamen bağlı sinir ağından oluşmaktadır [171]. Klasik sinir ağlarında olduğu gibi tüm ağda tek bir türevlenebilir skor fonksiyonu ve en sondaki tamamen bağlı katmanda da bir kayıp fonksiyonu bulunmaktadır. CNN'leri klasik ağlardan farklı kılan en belirgin özellik daha az miktarda eğitime ve daha az parametreye ihtiyaç duymalarıdır.

Klasik sinir ağlarını özellikle büyük boyutta görüntüler ile kullanmak bazı sorunlara yol açabilmektedir. Örneğin Alex Krizhevsky tarafından oluşturulan CIFAR-10 (Kanada İleri Araştırma Enstitüsü (Canadian Institute For Advanced Research)) [172] veri setinde, görüntüler yalnızca 32x32x3 boyutundadır (32 genişlik, 32 yükseklik, 3 renk kanalı) ve bu da sinir ağının ilk gizli katmanındaki tek bir tamamen bağlı nöronun $32 \times 32 \times 3 = 3072$ ağırlığa sahip olacağını göstermektedir. Bu miktar her ne kadar sorun çıkarmayacak düzeyde olsa da, örneğin 240x240x3 boyutunda bir görüntüde ağırlık sayısı bir anda 172800'e çıkacaktır. Ağdaki nöron sayısı arttıkça da parametre sayısı hızlı bir şekilde yükselmeye devam edecektir. Dolayısıyla tam bağlı klasik sinir ağlarında parametre sayısı yüksek çözünürlüklü görüntülerde oldukça büyük sorunlara yol açacak ve nihayetinde ezberleme (overfitting) gibi durumlar da ortaya çıkacaktır. CNN'de ise görüntülerin ele alınış şekli tamamen farklıdır.

Tablo 9. En çok kullanılan derin öğrenme tabanlı segmentasyon algoritmalarının özeti [173]

Yöntem	Yıl	Gözetim					Eğitim			Tip			Modüller		Açıklama
		G	Z	S	İ	P	TH	ÇH	ÇÖ	SS	SÖ	Ö	RNN	K-K	
Global Ortalama Havuzlama	2013	✓					✓				✓				Nesneye özgü yumuşak segmentasyon
DenseCRF	2014				✓	✓				✓					Segmentasyonu yükseltmek için CRF kullanmak
FCN	2015	✓				✓				✓				Tamamen konvolüsyonel katmanlar	
DeepMask	2015	✓					✓			✓				Segmentasyon ve sınıflandırma için eş zamanlı öğrenme	
U-Net	2015	✓				✓				✓			✓	Çok ölçekli öznitelik birleştirmeli kodlayıcı-kod çözücü	
SegNet	2015	✓				✓				✓			✓	Havuzlama indislerinin aktarıldığı kodlayıcı-kod çözücü	
RNN tipinde CRF	2015	✓					✓			✓		✓		CRF'leri eğitilebilir RNN modülleri olarak simüle etmek	
Derin Ayrıştırma Ağı	2015	✓					✓							Yüksek dereceli bağımlılığı içerebilmek için paylaşılmamış filtreleri kullanma	
BoxSup	2015	✓								✓				Zayıf denetim için sınırlayıcı kutu kullanma	
SharpMask	2016	✓					✓				✓		✓	Çok katmanlı öznitelik füzyonuna sahip geliştirilmiş DeepMask	
Ölçeklendirmeye Dikkat	2016	✓					✓			✓				Çok ölçekli girişlerden özniteliklerin füzyonu	
Semantik Segmentasyon	2016	✓						✓		✓				Görüntü segmentasyonu için çekişmeli eğitim	
Conv LSTM ve Konumsal İnhibisyon	2016	✓					✓				✓	✓		Örnek segmentasyonu için konumsal inhibisyon kullanma	
JULE	2016		✓				✓			✓		✓		Segmentasyon için birleştirilmiş gözetimsiz öğrenme	
ENet	2016	✓					✓			✓				Gerçek zamanlı segmentasyon için kompakt ağ	
Örnek Bazlı Segmentasyon	2016	✓					✓				✓			Örnek segmentasyonu için çok görevli yaklaşım	
Mask R-CNN	2017	✓					✓			✓				Segmentasyon için bölge teklif ağı kullanma	
Büyük Filtre Önemlidir	2017	✓					✓			✓			✓	Karmaşık öznitelikleri öğrenmek için daha büyük filtrelerin kullanılması	
RefineNet	2017	✓					✓			✓			✓	Hassas segmentasyon için çok yollu iyileştirme modülü	
PSPNet	2017	✓					✓			✓				Ölçeğe bağımlı olmayan çok ölçekli havuzlama	
Tiramisu	2017	✓					✓			✓			✓	DenseNet 121 öznitelik çıkarıcı	
Görüntüden Görüntüye Çeviri	2017	✓						✓		✓			✓	Görüntüyü segment haritalarına dönüştürmek için koşullu GAN	
Dikkat ile Örnek Segmentasyonu	2017	✓					✓				✓	✓		Görüntü segmentasyonu için dikkat modülleri	
W-Net	2017		✓				✓			✓			✓	Normalleştirilmiş kesim kaybı kullanarak gözetimsiz segmentasyon	
PolygonRNN	2017		✓				✓			✓			✓	RNN kullanarak konturlar oluşturma	
Derin Katman Basamaklandırma	2017	✓					✓			✓				Farklı karmaşıklıkta pikselleri işlemek için çok seviyeli yaklaşım	
Konumsal Yayılma Ağı	2017	✓					✓			✓				Doğrusal etiket yayılımı kullanarak geliştirme	
DeepLab	2018	✓					✓			✓				Genişlemiş/boşluklu (atrous) konvolüsyon, konumsal havuzlama piramidi, DenseCRF	
SegCaps	2018	✓					✓			✓				Segmentasyon için kapsül ağları	
Çekişmeli İş Birliği	2018		✓				✓			✓				Birden fazla ağ arasında çelişmeli iş birliği	
Süper Piksel Denetimi	2018		✓				✓			✓				Süper piksel iyileştirme işlevini denetleyici sinyaller olarak kullanma	
Derin Ekstremum Kesim	2018		✓				✓			✓				Etkileşimli segmentasyon için ekstremum noktaları kullanma	
İki Akış Füzyonu	2019		✓				✓			✓				Görüntü akışı ve etkileşim akışını aynı anda kullanma	
SegFast	2019	✓					✓			✓			✓	SqueezeNet kodlayıcıda derinlemesine ayrılabilir konvolüsyon kullanma	

G, gözetimli; Z, zayıf gözetimli; S, gözetimsiz; İ, interaktif; P, parçalı gözetimli; TH, tek hedefli optimizasyon; ÇH, çok hedefli optimizasyon; ÇÖ, çekişmeli öğrenme; SS, semantik segmentasyon; SÖ, sınıfa özgü segmentasyon; IN, örnek segmentasyonu; RNN, tekrarlayan sinir ağı modülleri; K-K, kodlayıcı-kod çözücü mimarisi; FCN, tamamen konvolüsyonel ağ; LSTM, uzun kısa süreli hafıza; GAN, üretici çekişmeli ağ; CRF, koşullu rastgele alan.

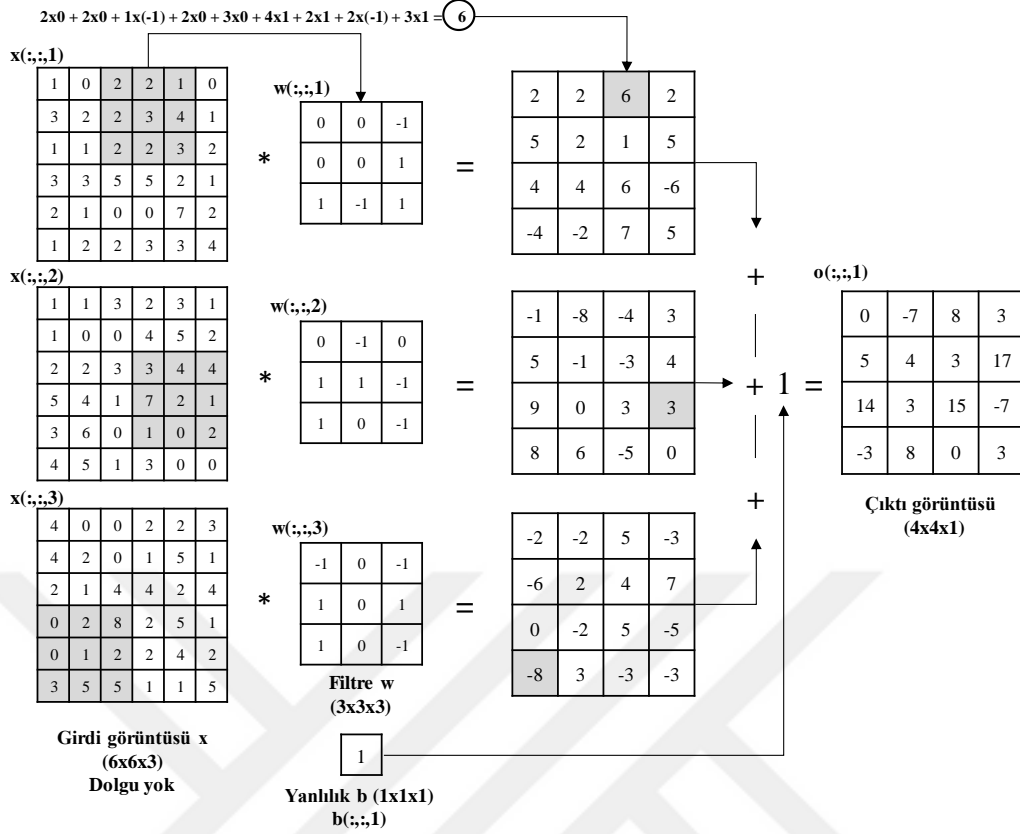
Katmanlar; genişlik, uzunluk ve derinlik (kanal sayısı) olmak üzere üç boyutlu bir düzendedir ve bir katmandaki nöronlar, tamamen bağlı ağlardakinin aksine kendinden bir önceki katmanın sadece küçük bir alanına bağlıdır (yerel bağlantı). Alıcı alan (receptive field) olarak adlandırılan bu alan bir hiperparametredir ve bu alanın boyutu filtre boyutu ile eşdeğerdir. Örneğin CIFAR-10 ağında alıcı alan (veya filtre boyutu) 5×5 ise, konvolüsyon katmanındaki her bir nöron girdi görüntüsündeki $[5 \times 5 \times 3]$ bir bölgeye, toplam $5 \times 5 \times 3 = 75$ (ve +1 yanlılık) ağırlıkla bağlı olacaktır. CNN'deki yerel bağlantı anlayışı sayesinde parametre sayısında büyük bir azalma meydana gelmektedir.

Basit bir CNN bir katmanlar dizisinden oluşmaktadır ve her bir katman türevlenebilir fonksiyonlar ile bir aktivasyon hacmini başka bir aktivasyon hacmine dönüştürmektedir.

CNN mimarilerinde temel olarak üç çeşit katman kullanılmaktadır: Konvolüsyon katmanı, havuzlama katmanı, tam bağlı katman (klasik sinir ağlarındaki gibi).

2.4.1.1.1. Konvolüsyon Katmanı

CNN'in temel katmanı olan bu katmanda hesaplama yükü oldukça fazladır. Bu katmanın parametreleri bir dizi öğrenilebilir filtre içerir. Genelde 3×3 , 5×5 gibi boyutlara sahip olan bu filtreler, üç renk kanalının bulunduğu görüntülerde $3 \times 3 \times 3$ veya $5 \times 5 \times 3$ (5 piksel genişlik, 5 piksel yükseklik ve 3 renk kanalı) boyutlarındadırlar. Konvolüsyon işlemi sırasında filtreler görüntünün genişliği ve yüksekliği boyunca kaydırılır, filtrelerdeki değerler ile görüntünün filtreye karşılık gelen bölümündeki değerler bire bir nokta çarpımı ile çarpılır ve elde edilen değerler toplanır. Bu işlem tüm görüntü boyunca uygulandıktan sonra yeni bir görüntü (aktivasyon haritası) meydana gelmiş olur. Konvolüsyon işlemine tabi tutulacak görüntünün üç kanallı olması durumunda kullanılacak filtrenin de üç kanallı olması gerekmektedir. Görüntünün her bir kanalındaki değerler ile filtrenin o kanalındaki değerler konvolüsyon işlemine girecek, dolayısıyla üç adet 2 boyutlu aktivasyon haritası elde edilecektir. Bu haritaların toplama işlemine tabi tutulmasıyla 2 boyutlu sonuç haritası meydana gelecektir. Konvolüsyon işleminde ikinci bir filtrenin daha kullanılması demek, 2 boyutlu bir sonuç haritasının daha oluşması anlamına gelecektir. Dolayısıyla, örneğin $6 \times 6 \times 3$ boyutundaki bir görüntünün $3 \times 3 \times 3$ 'lük iki adet filtre ile konvolüsyonu sonucunda $4 \times 4 \times 2$ 'lik bir aktivasyon haritası elde edilmiş olacaktır (3 kanallı örnek bir görüntü ile 3 kanallı örnek bir filtrenin konvolüsyonuna ilişkin işlemler Şekil 17'de gösterilmektedir). Konvolüsyon sonrası $6 \times 6 \times 3$ 'lük boyutun $4 \times 4 \times 2$ 'ye düşmesinde yani çıktı görüntüsünün boyutunun

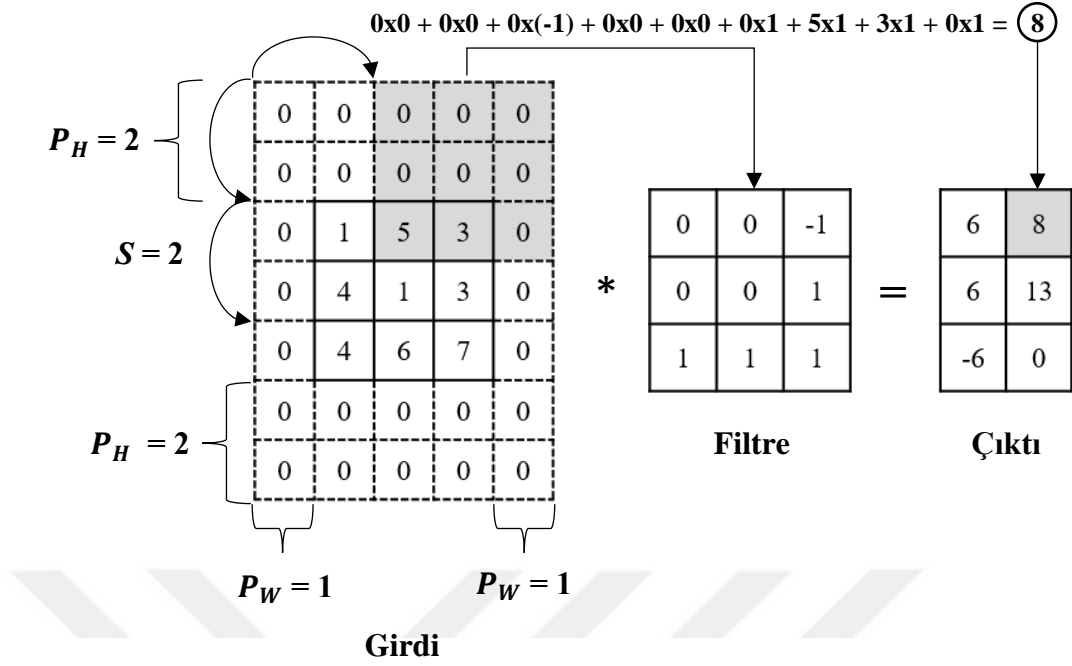


Şekil 17. Üç kanallı bir görüntü ile üç kanallı bir filtrenin konvolüsyonuna ilişkin işlemler. Girdi görüntüsündeki 3x3'lük taralı alanlar ile o alanlara karşılık gelen filtrelerin konvolüsyonu sonucunda oluşan değerler de taralı şekilde gösterilmiştir. Girdi görüntüsüne dolgu eklenmemiş, adım uzunluğu 1 olarak alınmıştır.

belirlenmesinde üç hiperparametre rol oynamaktadır. Bunlar kullanılan filtrenin boyutu, adım uzunluğu ve girdi görüntüsüne uygulanacak olan sıfır dolgusunun miktarıdır.

Adım uzunluğu, konvolüsyon sırasında filtrenin kaydırma işleminin her bir adımda kaç değer atlayarak yapılacağını gösterir. Örneğin, adım uzunluğu 2 olarak alındığında ilk konvolüsyon işleminden sonra filtre, girdi görüntüsünde ilk pikselden 2 piksel sonrasına kaydırılır. Adım uzunluğu seçilirken girdi görüntüsünün genişliği, yüksekliği ve kullanılan sıfır dolgulama miktarı göz önünde bulundurulmalıdır. Aksi takdirde kaydırma işlemi sırasında filtrenin görüntü boyutuyla uyuşmaması gibi durumlar ortaya çıkabilmektedir.

Sıfır dolgu, özellikle girdi görüntüsü ile çıktı görüntüsünün boyutlarının aynı kalması istendiğinde girdi görüntüsünün kenar bölümlerine eklenen sıfırlardır. Girdi görüntüsünün genişliği W , yüksekliği H , derinliği (kanal sayısı) D , uygulanacak filtre sayısı K , alıcı alan boyutu (filtre boyutu) F (birbirine eşit olan satır ve sütun sayısından herhangi biri), adım



Şekil 18. Yükseklik için 2, genişlik için 1 kademe sıfır dolgulama uygulandıktan sonra 2 piksellik adım uzunluğuna göre iki boyutlu konvolüsyon işleminin yapılışı. P_W ve P_H ile genişlik ve yükseklik için kullanılan sıfır dolgulama miktarları, S ile adım uzunluğu miktarı gösterilmektedir. Çıktıdaki taralı değerler, girdideki taralı alanlarla 3x3'lük filtrenin konvolüsyonu sonucunda hesaplanmıştır.

uzunluğu S (sıra ve sütun için aynı) ile gösterildiğinde, girdi görüntüsü ile çıktı görüntüsünün konvolüsyon işlemi sonrasında aynı boyutlarda kalabilmesi için görüntünün soluna ve sağına eklenmesi gereken sıfır dolgulama miktarı (P_W) Eşitlik 49 ile, görüntünün altına ve üstüne eklenmesi gereken sıfır dolgulama miktarı (P_H) ise Eşitlik 50 ile bulunur. Sıfır dolgulama ve adım uzunluğuna göre konvolüsyon işlemi tamamlandıktan sonra oluşacak çıktı görüntüsünün genişliği (W_{son}) ve yüksekliği (H_{son}) Eşitlik 51'deki gibi olacaktır. Çıktı görüntüsünün derinliği (D_{son}) ise uygulanacak filtre sayısına yani K 'ye eşit çıkacaktır. İki boyutlu konvolüsyon işleminde sıfır dolgulama ve adım uzunluğunun kullanımına ilişkin işlemler Şekil 18'de gösterilmektedir.

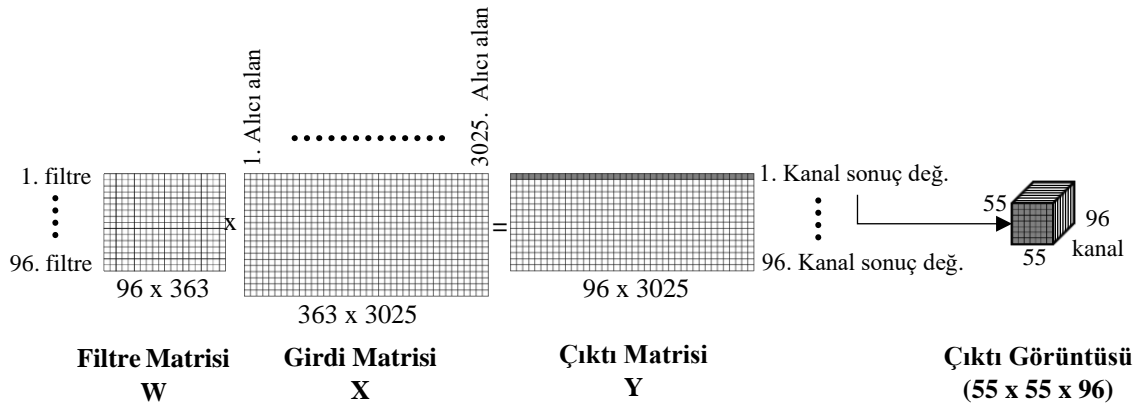
Filtre boyutu, konvolüsyon işleminde çıktı boyutunun belirlenmesinde rol oynayan bir diğer faktördür. Çıktı boyutunun girdi boyutuyla aynı olması istendiği durumlarda sıfır dolgu miktarı ve adım uzunluğu filtre boyutuna göre ayarlanmaktadır. Çıktı görüntüsünün derinliği (kanal sayısı) konvolüsyon işleminde kullanılan filtre sayısına eşittir.

$$P_W = \frac{(S - 1).W - S + F}{2} \quad (49)$$

$$P_H = \frac{(S - 1).H - S + F}{2} \quad (50)$$

$$H_{son} = \frac{(H - F + 2P_H)}{S} + 1, \quad W_{son} = \frac{(W - F + 2P_W)}{S} + 1 \quad (51)$$

Konvolüsyon işlemlerinin hesaplanmasında yaygın olarak kullanılan yöntemlerden biri büyük boyutlu iki matrisinin çarpılmasıdır. Bu yöntemin uygulanabilmesi için girdi görüntüsünün ve uygulanacak filtrelerin matrislerle ifade edilmesi gerekmektedir. Örneğin $(227 \times 227 \times 3)$ boyutundaki bir girdi, $(11 \times 11 \times 3)$ boyutlu 96 adet filtre ile adım aralığının 4 olduğu ve sıfır dolgulamanın uygulanmadığı bir konvolüsyon işlemine tabi tutulsun. Bu durumda ilk adım her bir filtreyi $11 \times 11 \times 3 = 363$ elemanlı bir sütun vektörüne dönüştürmek olacaktır. Bu işlem 96 filtrenin hepsi için uygulandığında (96×363) boyutlu bir filtre matrisi (W) elde edilecektir. İkinci adımda, adım uzunluğu 4 olduğunda girdi görüntüsünde kaç bölgenin filtreler ile birebir çarpım işlemine gireceği Eşitlik 14 ve Eşitlik 15 yardımıyla hesaplanır $((227-11)/4+1 = 55$ bölge). 55 satırda, 55 sütunda olmak üzere toplam 3025 adet alıcı alan ve her bir alıcı alana ilişkin 363 girdi değeri bulunmaktadır. Dolayısıyla girdi matrisinin (X) boyutu (363×3025) olacaktır. Bu iki matrisin çarpımı ile (96×3025) boyutunda çıktı matrisi (Y) elde edilecektir. Çıktı matrisindeki her bir satır, çıktı görüntüsündeki her bir kanala ait sonuç değerlerini göstermektedir. Bu satırların her biri (55×55) boyutuna

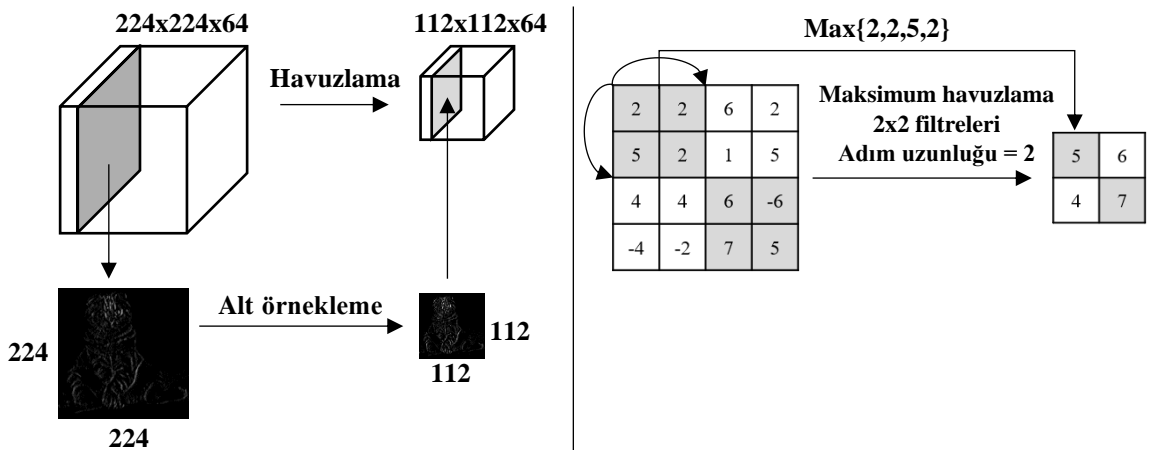


Şekil 19. Üç kanallı bir görüntü ile üç kanallı bir filtrenin konvolüsyonuna ilişkin işlemler. Girdi görüntüsündeki 3×3 'lük taralı alanlar ile o alanlara karşılık gelen filtrelerin konvolüsyonu sonucunda oluşan değerler de taralı şekilde gösterilmiştir. Girdi görüntüsüne dolgu eklenmemiş, adım uzunluğu 1 olarak alınmıştır.

çevrildiğinde derinliği 96 olan yani (55x55x96) boyutundaki çıktı görüntüsünü oluşturacaktır. Konvolüsyon işlemine ilişkin matris işlemleri Şekil 19’da gösterilmektedir.

2.4.1.1.2. Havuzlama Katmanı

CNN mimarisinde art arda gelen konvolüsyon katmanları arasına periyodik olarak bir havuzlama katmanı eklemek yaygındır. Ağın bu katmanında öğrenilen herhangi bir parametre yoktur. Bu katmanın görevi, ağıdaki parametre sayısını ve hesaplama yükünü azaltabilmek için girdi matrisinin boyutunu aşamalı olarak düşürmek ve böylece ezberlemeyi kontrol altına alabilmektir. Özellikle lokasyon bilgisinin çok önemli olmadığı problemlerde bu katman sayesinde iyi sonuçlar alınabilmektedir. Havuzlama Katmanı, girdinin her bir kanalında bağımsız olarak çalışmakta ve maksimumu veya ortalamayı bulma yöntemlerini kullanarak konumsal boyutlandırmayı sağlamaktadır. En yaygın olarak kullanılan havuzlama katmanında 2x2’lik filtreler adım uzunluğu 2 alınarak uygulanmakta ve her bir kanalın boyutu bu sayede genişlik ve yükseklik boyunca yarıya düşürülmektedir. Dolayısıyla %75 oranında bir aktivasyon işlemi işlem dışı tutulmuş olmaktadır. Bu şekilde uygulanan havuzlama işleminde her bir kanalda girdiye ait 2x2’lik alandaki 4 değer maksimumu (veya ortalaması) alınarak, bu 4 değer yerine tek bir değer elde edilmektedir. Girdinin boyutu yarıya düşmekle beraber kanal sayısı değişmemektedir. Maksimum havuzlamaya ilişkin işlemler Şekil 20’de gösterilmektedir.



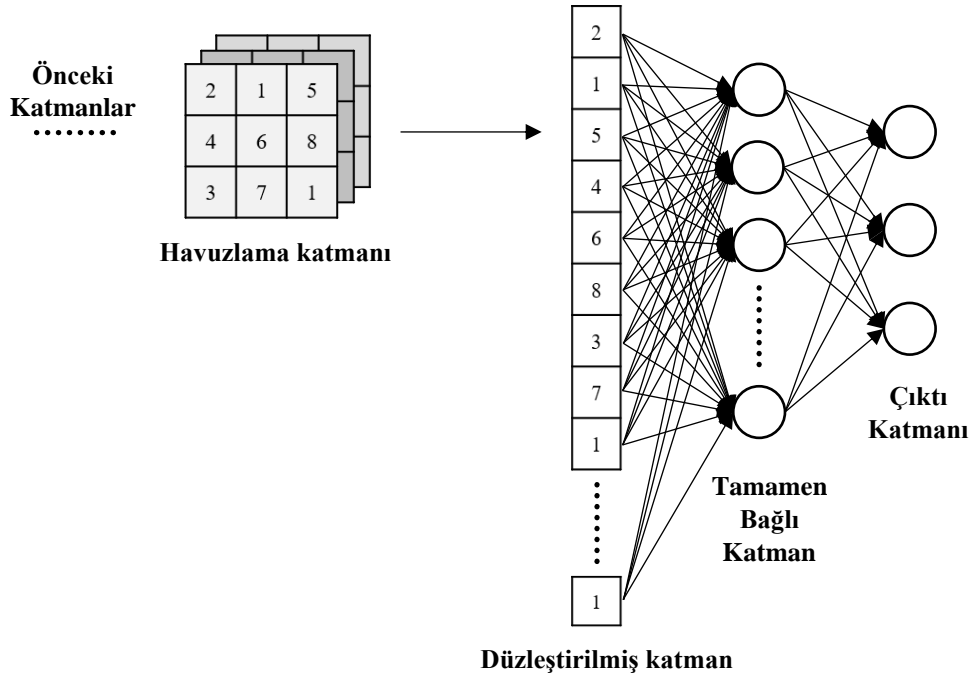
Şekil 20. Maksimum havuzlama işlemleri. Solda, (224x224x64) boyutlu girdi 2x2’lik filtreler ile adım uzunluğu 2 alınarak havuzlanıyor ve boyutu (112x112x64)’e düşüyor. Sağda, örnek bir 2 boyutlu matriste aynı koşullarda maksimum havuzlamanın nasıl sonuçlar verdiği gösteriliyor.

2.4.1.1.3. Tamamen Bağlı Katman

Tamamen bağlı katmanlar genellikle CNN mimarisinin sonuna doğru bulunurlar. Kendisinden önceki katmanın (tamamen bağlı, havuzlama veya konvolüsyon) çıktısına tam olarak bağlanırlar ve o katmandaki nöronları alarak kendisinin sahip olduğu her bir nörona bağlarlar. Bu katmanlar sayesinde konvolüsyon ve havuzlama katmanlarında öğrenilen özelliklerin doğrusal olmayan kombinasyonlarının da öğrenilebilmesi ve sınıf skorları gibi hedeflerin optimize edilmesi sağlanabilmektedir. Şekil 21’de tamamen bağlı katman için bir örnek gösterilmektedir.

2.4.1.1.4. Aktivasyon Fonksiyonunun Bulunduğu Katman

Aktivasyon fonksiyonu bir nöronun aktif olup olmayacağına karar vermek için kullanılmaktadır. Aktivasyon fonksiyonunun kullanılmadığı durumda çıktı sinyali doğrusal bir fonksiyondan ibaret olmakta, bu durumda sinir ağı sınırlı öğrenme gücüne sahip bir lineer regresyon gibi davranmaktadır. Doğrusal olmayan aktivasyon fonksiyonları sayesinde ağların daha güçlü öğrenmesi sağlanabilmektedir. Geriye yayılım algoritması ile öğrenme işlemi gerçekleştirilirken türev işlevine ihtiyaç duyulduğu için seçilecek aktivasyon

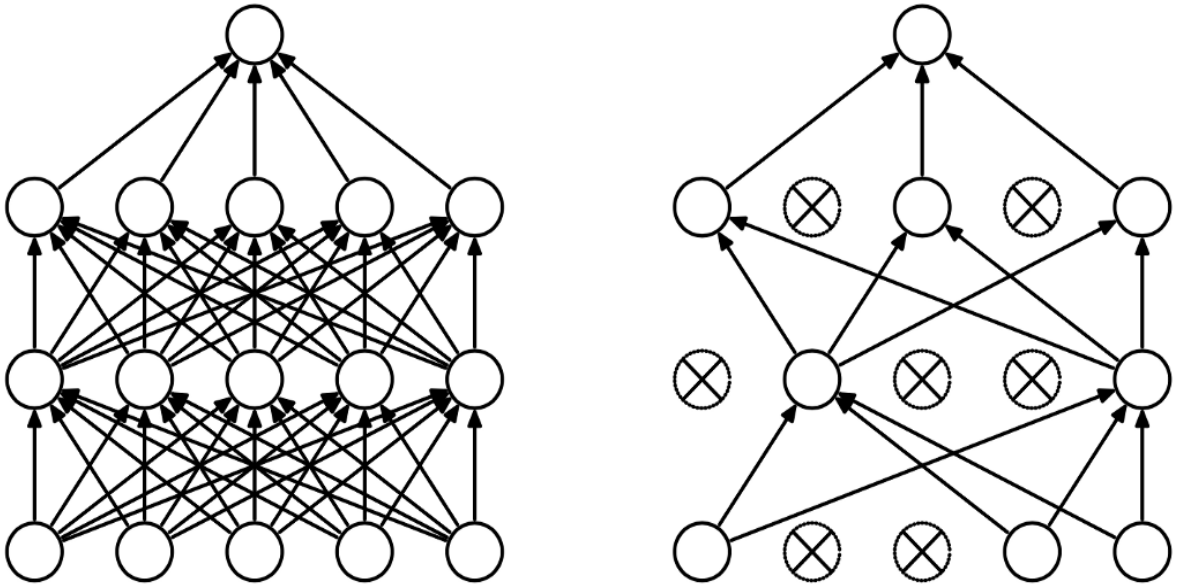


Şekil 21. Havuzlama katmanının düzleştirilmesinden sonra tamamen bağlı katmanlara bağlantısı

fonksiyonunun türevlenebilir olması öğrenme işleminin gerçekleşebilmesini sağlayacaktır. Literatürde en yaygın kullanılan aktivasyon fonksiyonları Tablo 10’da gösterilmektedir.

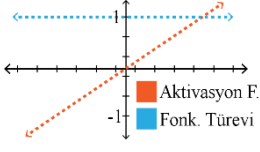
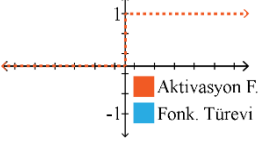
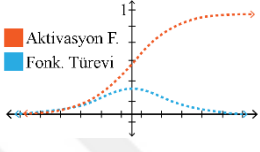
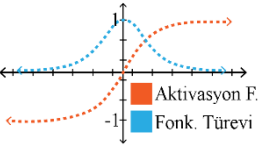
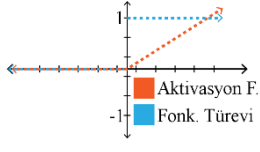
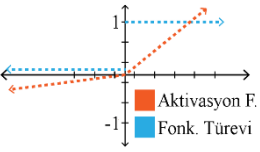
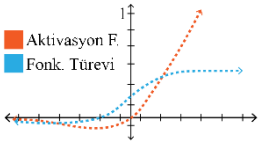
2.4.1.1.5. Düğüm Seyreltme (Dropout) Katmanı

Büyük bir ağda çok uzun süre eğitim yapıldığında veya kullanılan veri sayısının az olduğu durumlarda ezberleme sorunu ortaya çıkabilmektedir. Düğüm seyreltme katmanı [174] bir ağda aşırı öğrenmeyi ortadan kaldırmak için kullanılan bir katmandır. Sinir ağı içerisinde belli kurallara göre (eşik değeri kullanarak ya da rastgele) belli nöronların kullanılmaması mantığıyla çalışmaktadır. Konvolüsyon katmanlarında kullanılan parametre sayısının az olması, öznitelik haritalarındaki konumsal ilişkiler ve aktivasyonlar arası yüksek derece ilişki ihtimali; düğüm seyreltme katmanlarının konvolüsyon katmanları arasında iyi başarımlar elde edememesine yol açabilmektedir. Bu nedenle, düğüm seyreltme katmanları genellikle parametre sayısının çok yüksek olmasından dolayı ezberleme riskinin yüksek olduğu tamamen bağıli katmanlardan sonra kullanılmaktadırlar. Bağıli katmanlardaki bazı bağların koparılması nöronların birbirleri hakkında daha az bilgiye sahip olması ve dolayısıyla birbirlerinin ağırlık değişimlerinden daha az etkilenmelerini sağlamaktadır. Böylece daha tutarlı (robust) modeller oluşturulabilmektedir. Şekil 22’de normal bir yapay sinir ağı bağlantıları ve düğüm seyreltme uygulanmış ağ yapısı görülmektedir.



Şekil 22. Düğüm seyreltme sinir ağı modeli. Solda, standart sinir ağı; sağda, düğüm seyreltme sonrasında sinir ağı

Tablo 10. Literatürde en yaygın kullanılan aktivasyon fonksiyonları

Fonksiyon	Grafik	Denklem	Türev	Açıklama
Doğrusal		$f(x) = cx$	$f'(x) = c$	Bu fonksiyon girişi c ile ölçeklendirir. Eğer $c=1$ ise, giriş her zaman çıktıya eşittir. Fonksiyonun türevi sabit olduğundan sinir ağı her zaman sabit bir miktarda düzeltilir.
İkili Basamak		$f(x) = \begin{cases} 0, & x < 0 \text{ için} \\ 1, & x \geq 0 \text{ için} \end{cases}$	$f'(x) = \begin{cases} 0, & x \neq 0 \text{ için} \\ ?, & x = 0 \text{ için} \end{cases}$	İkili sınıflandırıcı olarak kullanılabildiği için daha çok çıkış katmanlarında tercih edilir. Gizli katmanlarda türevi öğrenme değeri temsil etmediği için kullanılması tavsiye edilmez.
Lojistik (Sigmoid)		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$	En yaygın kullanılan aktivasyon fonksiyonlarından birisidir. [0,1] aralığında çıktı üretir. Sınıflandırma problemleri için kullanışlıdır. Fonksiyonun uçlarına doğru y değerleri x'teki değişikliklere çok az tepki vermektedir. Bu bölgelerde türev değerleri çok küçük olur ve 0'a yakınsar. Buna gradyanların ölmesi/kaybolması (vanishing gradient) denir ve öğrenme olayı minimum düzeyde gerçekleşir.
TanH (Hiperbolik Tanjant)		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$	Sigmoid fonksiyonuna çok benzerdir fakat fonksiyonun aralığı (-1,1)'dir. Türevinin daha dik olması daha hızlı öğrenme ve sınıflama işlemi için daha geniş aralığa sahip olabilme imkânı doğurur. Fonksiyonun uçlarında gradyanların ölmesi problemi devam etmektedir.
Düzleştirilmiş Lineer Birim (ReLU)		$f(x) = \begin{cases} 0, & x < 0 \text{ için} \\ x, & x \geq 0 \text{ için} \end{cases}$	$f'(x) = \begin{cases} 0, & x < 0 \text{ için} \\ 1, & x \geq 0 \text{ için} \end{cases}$	Negatif ekseninde 0 değerlerini alması ağına daha hızlı çalışmasını sağlar. Fakat bu bölgede türevin sıfır olması (ölen ReLU) öğrenmenin de gerçekleşmediği anlamına gelmektedir. Hesaplama yükünün sigmoid ve hiperbolik tanjant fonksiyonlarına göre az olması çok katmanlı ağlarda daha çok tercih edilmesine sebep olmuştur.
Sızıntı (Leaky) Düzleştirilmiş Lineer Birim (PReLU)		$f(x) = \begin{cases} ax, & x < 0 \text{ için} \\ x, & x \geq 0 \text{ için} \end{cases}$	$f'(x) = \begin{cases} a, & x < 0 \text{ için} \\ 1, & x \geq 0 \text{ için} \end{cases}$	0.01 gibi küçük alfa değerleri kullanılarak fonksiyonun eksi değerler de alabilmesi sağlanmıştır. Bu sayede ReLU'daki ölen gradyan sorunu çözülmüş ve negatif bölgedeki değerler için de öğrenme sağlanmıştır.
Kendinden Geçitli (Swish) Fonksiyon		$f(x) = \frac{x}{1 + e^{-x}}$	$f'(x) = \frac{1}{1 + e^{-x}} + \frac{e^{-x}x}{(1 + e^{-x})^2}$	Sızıntı ReLU'ya benzer şekilde negatif bölgede değerler almaktadır, fakat aldığı değerler doğrusal değildir. Gradyanların ölmesi sorunu giderilmiştir. ReLU'ya oranla daha fazla işlem gücü gerektirmektedir.

2.4.1.1.6. Yığın Normalizasyonu (Batch Normalization) Katmanı

Yığın normalizasyon katmanı, her bir giriş kanalını mini yığın boyunca normalleştirmektedir. Konvolüsyonel sinir ağlarında öğrenimi hızlandırabilmek için konvolüsyon katmanları ile ReLU (Düzleştirilmiş Lineer Birim (Rectified Linear Unit)) katmanları gibi doğrusal olmayan katmanlar arasına yığın normalizasyon katmanı konabilmektedir. Bu katman mini yığın içindeki veriyi her bir öznitelik boyutu boyunca ortalama ve standart sapmayı tahmin edebilmek için kullanır. Daha sonra bu parametreler, sıfır ortalamalı ve bir standart sapmalı normalleştirilmiş öznitelikler elde etmek için kullanılırlar. Yığın normalizasyon katmanı normalleştirilmiş değerlerin kaydırma ve ölçeklendirmelerini yapmak üzere gamma ve beta olmak üzere iki adet eğitilebilir parametre içermektedir. Bu katman sayesinde ağ içindeki eğitim sürecini aşırı derecede etkileyecek büyüklükteki ağırlıkların dengelenmiş olması sağlanmakta ve aynı zamanda eğitim hızı artırılmaktadır [175].

2.4.1.1.7. Sınıflandırma Katmanı

Öznitelikler belirlendikten sonra probleme uygun bir sınıflandırıcı ile sınıflandırma işleminin yapıldığı katmandır. Çok sınıflı sınıflandırma problemlerinde en yaygın kullanılan sınıflandırıcılar Softmax ve SVM'dir [176]. Softmax fonksiyonu, sinir ağı tarafından çıktı olarak elde edilen skor değerlerini normalize ederek test girdisinin hangi sınıfa daha yakın olduğuna dair olasılıksal değerler üretmektedir. Skor değerlerini içeren girdi vektörü x , sınıf aitlik olasılıklarını içeren çıktı vektörü p , sınıf sayısı K ile gösterildiğinde Softmax fonksiyonu Eşitlik 52'deki gibidir.

$$p_j = \frac{e^{x_j}}{\sum_{i=1}^K e^{x_i}} ; j = 1, \dots, K \quad (52)$$

SVM ve Softmax'ın çıktı olarak ürettikleri değerler birbirinden farklılık göstermektedir. SVM, test girdisinin hangi sınıfa daha benzer olduğuna dair benzerlik skoru üretirken, Softmax aynı amaca yönelik olasılık değerleri üretilmektedir. Kayıp (loss) fonksiyonunun değerinin hesaplanması SVM'de "hinge loss" ile yapılırken, Softmax'ta "cross entropy" kullanılmaktadır. SVM, loss değerini "hinge loss" ile hesaplarken, Softmax

cross entropy ile hesaplamaktadır. Loss fonksiyonu SVM’de $[0, \infty)$, Softmax’ta ise $[0, 1]$ aralığında değer almaktadır [177].

2.4.1.2. Derin Öğrenmede Kullanılan Hiperparametreler

Hiperparametreler ilgilenilen veri setine bir öğrenme algoritması uygulanmadan önce modeli tasarlayan kişi tarafından belirlenmesi gereken değişkenlerdir. Hiperparametreler, optimizasyona yönelik ve modele özel hiperparametreler olmak üzere iki grup altında toplanabilirler. Optimizasyona yönelik hiperparametreler; öğrenme hızı, mini-yığın boyutu, epok sayısı ve momentum iken, modele özel hiperparametreler ise gizli katman sayısı ve gizli katmanlardaki nöron sayısı, ağırlık azalımı ve başlangıç ağırlık değerlerinin belirlenmesi şeklindedir [178].

2.4.1.2.1. Öğrenme Hızı (Learning Rate)

Derin öğrenmede geri yayılım ile parametreler güncellenirken yeni ağırlıkların hesaplanması sırasında “öğrenme hızı” parametresi kullanılmaktadır. Bu değer eğitim öncesinde sabit bir değer olarak verilebilir, belli adımlarla artacak şekilde ayarlanabilir veya momentum değerine göre belirlenebilir.

Model öğrenme hızı optimal değerlerden çok daha küçükse, ideal duruma ulaşmak normalden çok daha uzun zaman alacaktır. Ayrıca lokal optimum değere takılarak, global optimum değere hiç ulaşamaması gibi sorunlar da çıkabilecektir. Öte yandan, eğer öğrenme hızı optimal değerden çok daha büyükse, öğrenme hızlanmakta fakat yakınsama olamama sorunu oluşabilmektedir. Öğrenme hızı olarak başlangıçta 0.1 veya 0.01 gibi büyük değerler seçilmekte, belli bir epoktan sonra bu değerler kademeli olarak düşürülmektedir [179].

2.4.1.2.2. Mini-Yığın (Mini Batch) Boyutu

Mini-yığın boyutu, parametre güncellemesinin gerçekleştiği ağa verilen alt örneklerin sayısıdır. Bu değer, eğitim sürecinin kaynak gereksinimlerini, hızını ve iterasyonların sayısını direkt olarak etkilemektedir.

Mini-yığın değerini en küçük 1, en yüksek ise eğitim kümesindeki tüm verilerin sayısı kadar alabilmek mümkündür. Mini-yığın değerinin 1 olduğu durumda “stochastic gradient descent” ile aynı işlev yerine getirilmiş olur ve bu durum “stokastik eğitim” olarak

adlandırılır ve her iterasyonda sadece tek bir veri üzerinde işlem yapılır. Bu eğitimde model gürültüleri öğrenebilir fakat yerel optimumda takılıp global optimuma hiç ulaşamayabilir.

Mini-yığın değerinin eğitim kümesindeki tüm elemanların sayısına eşit olduğu durumda yapılan işlem “batch gradient descent”le aynı olur ve bu durum “yığın eğitimi (batch training)” olarak adlandırılır. Aynı anda tüm eğitim verisinin işlenecek olması eğitimin çok uzun sürmesine, gürültülerin öğrenilememesine ve optimizasyon işlemindeki ilerlemelerin büyük adımlarla yapılmasına yol açacaktır. Bu tür eğitim genellikle eğitim kümesindeki eleman sayısının küçük (2000’den az) olduğu durumlarda tercih edilmektedir.

Mini-yığın boyutunun büyük olması, daha doğru gradyan değerlerinin hesaplanmasını sağlamaktadır, fakat ezberleme (overfitting) olasılığını da artırmaktadır. Ayrıca yığın boyutu arttıkça eğitim hesaplamalarındaki matris çarpım işlemlerinin büyüklüğü de artmakta dolayısıyla kullanılan bellek yetersiz kalabilmektedir. Mini-yığın boyutunun GPU belleğine sığacak şekilde 2, 4, 8, ..., 512 vb. ikinin katı değerler olarak belirlenmesi tavsiye edilmektedir [180; 181].

2.4.1.2.3. Epok (epoch) Sayısı

Epok (döngü) sayısı, eğitim algoritmasının tüm eğitim verisini toplamda kaç defa kullanacağını belirlemektedir. Doğru epok sayısını seçmek için dikkat edilmesi gereken metrik “doğrulama hatası”dır. Derin öğrenmede doğrulama hatasının ilk epoklarda yüksek, epoklar arttıkça ise düşük olması beklenmektedir, çünkü ağıdaki ağırlık değerleri adım adım güncellenmektedir. Sezgisel olarak doğrulama hatası azalmaya devam ettiği sürece modelin iterasyonlarına devam etmesi beklenmektedir. Modelin ne zaman durdurulacağını belirlemek için “Erken Durdurma (Early Stopping)” adlı bir teknik kullanılabilir. Bu teknikte, eğitim setindeki hata ile doğrulama setindeki hata arasındaki fark belirlenen eşik değerinin üzerine çıkmaya başladığında eğitim süreci durdurulmaktadır [113].

Epok ve mini-yığın parametrelerinin kullanımına bir örnek şu şekilde verilebilir: 200 örnekten oluşan bir veri kümesi, mini-yığın büyüklüğü 5 ve epok sayısı 1000 olacak şekilde eğitilmek istensin. Bu durumda veri kümesi 40 adet yığına bölünecektir (200/5). Her bir yığında 5 örnek yer alacaktır. Her bir 5 örnek için ağırlıklar hesaplandıktan sonra hata hesaplanacak ve ağırlık güncellemeleri yapılacaktır. Bir epok 40 yığın yani model için 40 güncelleme içerecektir. Dolayısıyla 1000 epok için model tüm veri kümesinden 1000 kere yararlanacak ve tüm eğitim sürecinde toplam 40000 güncelleme yapılmış olacaktır.

2.4.1.2.4. Momentum

Momentum, ağıda geri yayılım sürecinde ağırlıkların güncellenmesi sırasında kullanılan bir parametredir. Eğitim hızını ve doğruluğu artırmak amacıyla kullanılmaktadır. Özellikle stochastic gradient descent gibi çok fazla salınım oluşturan ve gürültü üreten yöntemlerde geri yayılım sırasında bir önceki ağırlık artış miktarını beta momentum katsayısı oranında işin içine katarak yeni ağırlık artış miktarı hesaplanmaktadır. Bu sayede salınımların ve gürültünün azaltılması ve hız artışı sağlanması hedeflenmektedir. Momentum parametresi büyükse, öğrenme hızı daha küçük tutulmalıdır. Büyük bir momentum değeri ayrıca yakınsamanın hızlı olacağı anlamına gelir. Ancak hem momentum hem de öğrenme hızı yüksek değerlerde tutulursa, yapılacak olan güncelleme çok büyük olacağından minimumu atlama gibi bir sorun oluşabilmektedir. Küçük bir momentum değeri ise yerel minimuma takılmayla sonuçlanabilir ve sistemin eğitimini de yavaşlatabilir. Tipik bir momentum seçimi 0.6 ile 0.8 arasındadır [98].

2.4.1.2.5. Gizli Katman Sayısı ve Gizli Katmanlardaki Nöron Sayısı

Gizli katman sayısının yüksek olması, sinir ağının eğitim verisine olan uyumunu artırabilmektedir. Tüm gizli katmanlar için aynı boyutu kullanmak genellikle daha iyi sonuçlar verebilmektedir. Ayrıca, ilk gizli katmanın girdi katmanından büyük bir boyuta sahip olması da ağıdın daha iyi çalışmasını sağlayabilmektedir. Katmanlardaki nöron sayısının artmasının daha fazla bellek ihtiyacı ve hesaplama yükü getireceği unutulmamalıdır. Nöron sayısının az olması yetersiz uyuma (underfitting) neden olabilmektedir [182].

Özellikle konvolüsyonel sinir ağlarında katman sayısı arttıkça modelin daha iyi öğrendiği görülmüştür. Örneğin, görüntü tanıma işlemlerinde ağıdaki ilk katmanlar görüntü ile ilgili genel hatları öğrenirken, daha derindeki katmanlar görüntüye özel detayları yakalayabilmektedirler [183].

2.4.1.2.6. Ağırlık Azalımı (Weight Decay)

Ağırlık azalımı, derin öğrenmede ağ modelinin eğitim sırasında aşırı öğrenmesini engellemek ve modelin yeni veriler üzerindeki performansını artırmak için kullanılan bir yaklaşımdır. Eğitim sırasında, geri yayılım gradyanı hesaplanırken ağıdın kayıp fonksiyonuna bir iyileştirme (regularization) terimi eklenir. Ağırlık azalım değeri, bu iyileştirme teriminin

gradyan hesaplamasında ne kadar baskın olacağını belirler. Tipik olarak optimal ağırlık azalım değeri sinir ağı modeli sayısız kez eğitilerek belirlenebilir. Yani, öncelikle aynı başlangıç ağırlık tahminleri setine ve aynı sinir ağı yapılandırmasına dayalı olarak sabit sayıda gizli katman birimiyle ağ eğitimi gerçekleştirilir. Ardından, en küçük genelleme hatasını ve optimal ağırlık azalımını belirleyebilmek için çeşitli ağırlık azalım değerlerine dayanarak sinir ağı modeli birkaç kez yeniden oluşturulur [184].

2.4.1.2.7. Başlangıç Ağırlık Değerlerinin Belirlenmesi

Ağın eğitimine başlanmadan önce yanlılık terimleri genelde 0 olarak başlatılmaktadır, ancak ağırlıkların başlangıç değerlerinin dikkatlice atanması eğitim algoritmasının minimuma ulaşmasında büyük bir etkiye sahip olacaktır. Ağırlık değerleri, 0.5 ile 0.9 arasında düzgün dağılıma veya 0 ortalamalı, 0.1 standart sapmalı normal dağılıma sahip olacak şekilde ya da daha önceki bir modelin ağırlık değerlerini kullanacak şekilde tanımlanabilir. Literatürde kullanılan farklı ağırlık belirleme yöntemleri bulunmaktadır [185]. Uygun başlangıç ağırlıklarının belirlenmesi modelin öğrenmesine ve öğrenme hızına etki etmektedir [186].

2.4.2. Semantik Segmentasyon

Semantik segmentasyon, görüntüdeki her bir pikseli, sınıflandırılacak kategorilerden biri ile ilişkilendiren derin öğrenme algoritmasıdır [187]. Bu bölümde, test görüntülerini deri, yanık ve arka plan olarak sınıflandırmak için, çoklu sınıflar için piksel bazlı segmentasyon yapabilen kodlayıcı–kod çözücü (encoder–decoder) CNN mimarilerinden U-Net ve SegNet mimarileri hakkında bilgiler verilmektedir.

2.4.2.1. U-Net Mimarisi

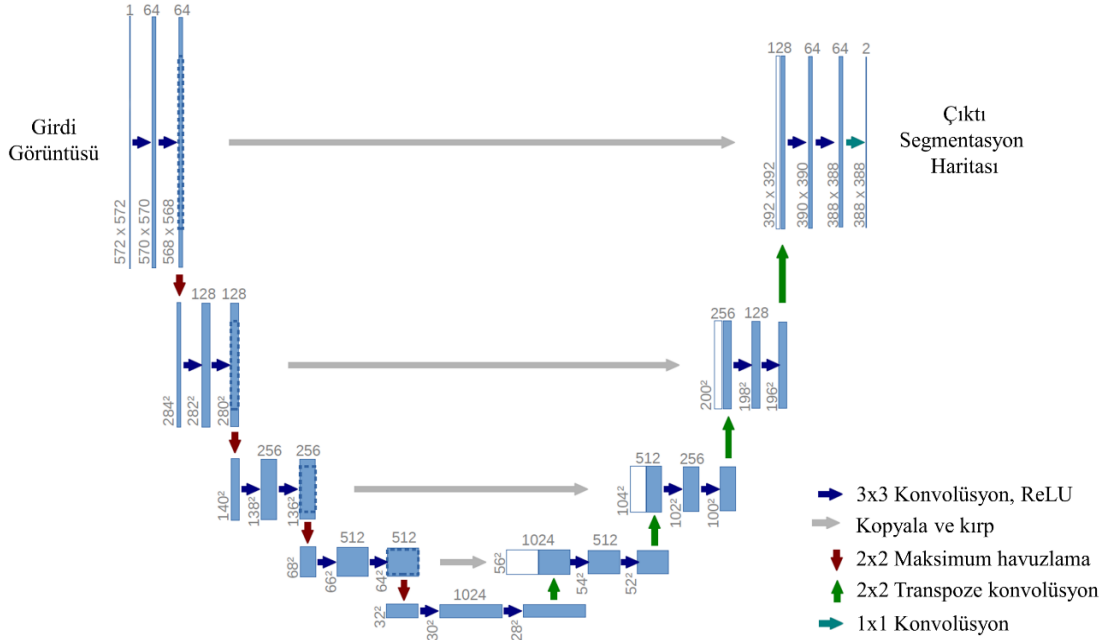
Biyomedikal görüntüler üzerinde daha iyi bir segmentasyon yapma önerisi ile ilk olarak 2015 yılında Olaf Ronneberger, Phillip Fischer, ve Thomas Brox tarafından tanıtılan U-Net mimarisi [188] bir çeşit CNN yaklaşımıdır.

CNN'in arkasındaki ana fikir, bir görüntünün öznetelik eşlemesini (feature mapping) öğrenmek ve daha ayrıntılı öznetelik eşlemesi yapabilmek için onu kullanmaktır. Bu işlem, görüntünün sınıflandırma aşamasında kullanılmak üzere bir vektöre dönüştürüldüğü

sınıflandırma problemlerinde iyi sonuçlar vermektedir. Fakat görüntü segmentasyonu işleminde, öznelik haritasının bir vektöre dönüştürülmesi adımından sonra, bu vektör kullanılarak görüntünün yeniden yapılandırılması gerekmektedir. Bir vektörün görüntüye dönüştürülmesi işlemi ise oldukça zor bir süreçtir. U-Net mimarisinin ortaya çıkış fikri de bu sorunun temeline dayanmaktadır. Bir görüntünün vektöre dönüştürülme aşamasında kullanılan öznelik eşlemesi, vektörün tekrardan görüntüye dönüştürülmesi için de kullanılacaktır. Bu da görüntüdeki bozulmaları büyük ölçüde azaltarak görüntünün yapısal bütünlüğünü koruyacaktır.

Şekil 23'te gösterilen U-Net ağı, büzülme yolu (contracting path) ve genişleme yolu (expansive path) olmak üzere iki yol içermektedir. Mimariye 'U' şeklini veren ve U-Net olarak adlandırılmasına neden olan da bu yollardır.

Kodlayıcı (encoder) da denilen büzülme yolu, görüntüdeki içeriği kapsayan ve her bir konvolüsyon işlemini (3x3 filtre) takiben, ReLU ve maksimum havuzlama (2x2 max pooling) işlemlerinin uygulandığı tekrarlı bir konvolüsyon işlemidir. En alttaki katman, büzülme katmanı ve genişleme katmanı arasında aracılık etmektedir ve iki adet 3x3'lük



Şekil 23. U-Net mimarisi (en düşük çözünürlüğün 32x32 piksel olduğu durum için bir örnek). Her mavi kutu çok kanallı bir öznelik haritasına karşılık gelmektedir. Kutuların üstünde kanal sayısı belirtilmektedir. x-y boyutu, kutunun sol alt kenarında verilmektedir. Beyaz kutular kopyalanan öznelik haritalarını temsil etmektedir. Her bir oka ilişkin işlev, şekle ait açıklama göstergesinde belirtilmektedir [188].

CNN katmanı ve ardından 2x2'lik üst konvolüsyon katmanından oluşmaktadır. Büzülme sırasında, konumsal bilgiler azaltılırken öznitelik bilgisi arttırılmakta ve bu sayede mimari karmaşık yapıları daha etkili bir şekilde öğrenebilmektedir.

Kod çözücü (decoder) olarak da adlandırılan genişleme yolu, transpoze konvolüsyonların kullanımıyla hassas lokalizasyonun sağlandığı simetrik bir yoldur. Dolayısıyla uçtan uca tamamen konvolüsyonel bir ağıdır, yani herhangi bir yoğun katman (dense layer) içermeyip sadece konvolüsyonel katmanlardan oluşmaktadır. Bu nedenle de herhangi bir boyuttaki görüntüye uygulanabilmektedir. Genişleme katmanında her bir blok, girdiyi 2 adet 3x3'lük CNN katmanına ve sonrasında onu takip eden 2x2'lik bir üst-örnekleme (upsampling) katmanına iletir. Ayrıca, her bir bloktan sonra, konvolüsyonel katman tarafından kullanılan öznitelik haritalarının sayısı simetriyi korumak amacıyla yarıya düşürülür. Bununla birlikte, her seferinde girdiye, mevcut genişleme katmanına karşılık gelen büzülme katmanının öznitelik haritası eklenir (depth concatenation). Bu eylem, görüntüyü küçültürken öğrenilen özniteliklerin onu yeniden yapılandırmak için kullanılmasını sağlayacaktır. Genişleme bloklarının sayısı, büzülme bloklarının sayısı ile aynıdır. Bu bloklar geçildikten sonra oluşan sonuç haritalama, ulaşılmak istenen küme sayısına eşit sayıda öznitelik haritasına sahip 3x3'lük bir CNN katmanından geçer. Elde edilen sonuç öznitelik haritasına önce piksel bazlı bir softmax aktivasyon fonksiyonu, sonrasında ise çapraz entropi kayıp fonksiyonu uygulanır. Bu sayede görüntüdeki her bir pikselin bir sınıfa ait olacak şekilde sınıflandırılması sağlanmakta, başka bir deyişle segmentasyon problemi çözülürken aynı zamanda çok sınıflı sınıflandırma işlemi yapılmaktadır [188].

2.4.2.2. SegNet Mimarisi

SegNet mimarisi [189], 2015 yılında Cambridge Üniversitesi tarafından öncelikli olarak trafik görüntülerinin piksel bazlı segmentasyonu için önerilen kodlayıcı-kod çözücü CNN alt yapısına sahip bir mimaridir.

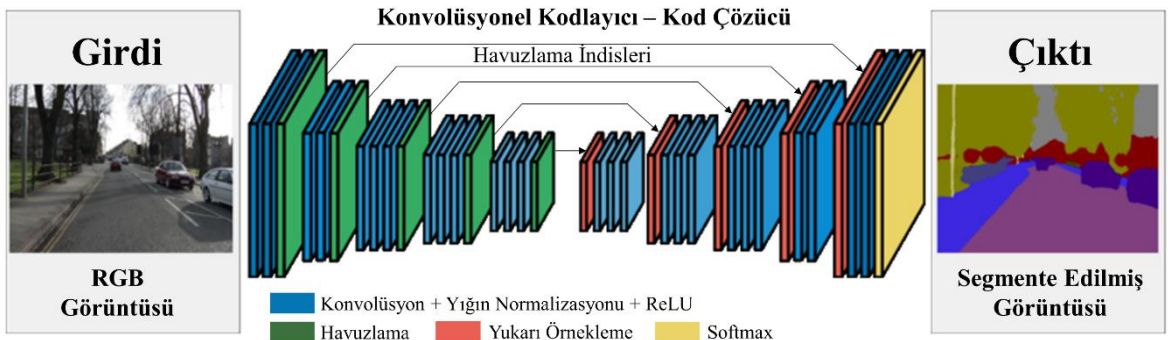
SegNet'i aynı düzeydeki diğer mimarilerden farklı kılan özelliği kod çözücünün düşük çözünürlükteki girdi öznitelik haritalarını yukarı örnekleme şeklindedir. Bu mimaride kod çözücü, maksimum havuzlama sırasında hesaplanan havuzlama indislerini lineer olmayan yukarı örnekleme işleminde kullanmaktadır. Dolayısıyla yukarı örnekleme için ihtiyaç duyulan öğrenme ihtiyacı da bu şekilde ortadan kalkmaktadır. Yukarı örneklenmiş haritalar

seyrektilir ve eğitilebilir filtreler ile konvolüsyon işlemine tabi tutularak yoğun öznitelik haritaları elde etmek mümkündür.

Şekil 24’te görüldüğü gibi SegNet mimarisi simetrik bir ağdan oluşur. Ağın bir tarafında kodlayıcı, diğer tarafında ise kod çözücü yer alır. Girdi, bir RGB görüntüsü; çıktı ise her bir rengin farklı bir nesneyi temsil ettiği, girdi görüntüsünün segmente edilmiş halidir.

Kodlayıcı bölümü üç çeşit katmandan oluşmaktadır: Konvolüsyon, havuzlama ve yığın normalizasyonu. Konvolüsyon katmanı, yerel öznitelikleri çıkarır; havuzlama katmanı, öznitelik haritasını aşağı örnekler ve konumsal değişmez öznitelikleri daha derin katmanlara iletir; yığın normalizasyonu katmanı ise eğitimi hızlandırmak amacıyla eğitim verisinin dağılımını normalleştirir. Temel olarak kodlayıcı, düz renkli piksellerden elde edilen düşük seviye görüntü bilgisinden yüksek seviye bilgi (trafik örneği için “araba”, “yol”, “yaya”) çıkartır. Kod çözücü daha sonra bu yüksek seviyeli bilgiyi alır ve başka bir orta seviyeye eşleyerek aynı büyüklükte bir görüntüye dönüştürür. Çıktı görüntüsünde, aynı etikete sahip nesnelere aynı renk ile temsil edilirler.

Kodlayıcı nesnelere “ne” oldukları ve yaklaşık olarak “nerede” oldukları bilgisini edinir. Fakat bu bilginin tam olarak hangi pikselden geldiği bilgisi kod çözücü tarafından elde edilir. Kod çözücü, kodlayıcı bölümünün ilgili kısmında maksimum havuzlama sırasında hesaplanan havuzlama indislerini kullanarak yukarı örnekleme yapar ve küçülmüş görüntüyü kendisine karşılık gelen görüntü ile aynı boyuta çeker. Sonrasında gelen konvolüsyon katmanı, kodlayıcının havuzlama katmanında meydana gelen kaybı telafi

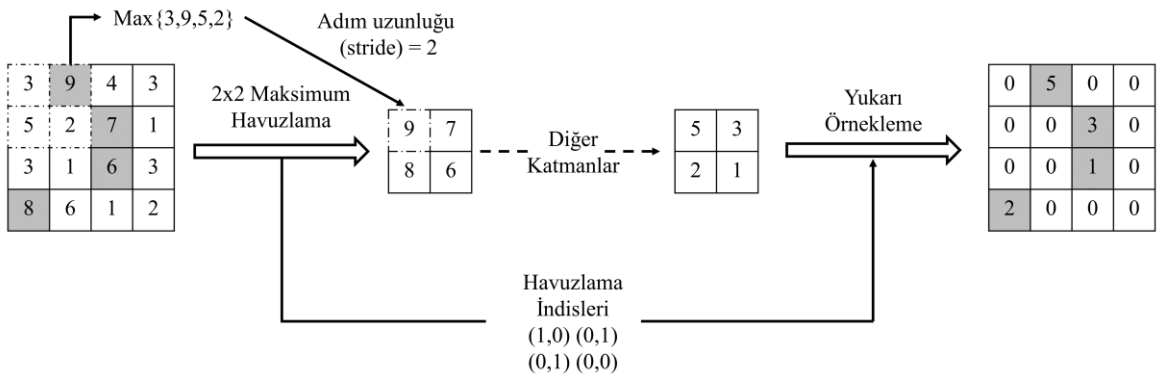


Şekil 24. SegNet mimarisinin bir örneği. Tam bağlantılı katman içermediği için tamamen konvolüsyonel bir mimaridir. Bir kod çözücü, kendisine karşılık gelen kodlayıcıdan gelen havuzlama indislerini kullanarak girdisini yukarı örnekler ve seyrek bir öznitelik haritası/haritaları üretir. Ardından, bu haritayı eğitilebilir bir filtre bankası ile konvolüsyon işlemine sokar ve haritanın yoğunlaşmasını sağlar. Son kod çözücü çıktı öznitelik haritaları piksel bazında sınıflandırma yapılabilmesi için bir softmax sınıflandırıcısına iletilir [189].

etmek için öznitelik haritasına geometrik detaylar ekler ve böylece oluşan sonuç görüntüsünün daha yumuşak geçişli olmasını sağlar. Kod çözücünün en sonunda elde edilen sonuç öznitelik haritalarını piksel bazında sınıflandıracak bir softmax katmanı bulunur.

SegNet'te, 2×2 'lik bir maksimum havuzlama kullanılır. Bu havuzlamada adım uzunluğu 2 olarak alındığından maksimumuna bakılacak olan 4 piksel her defasında farklıdır. Şekil 25'te önce $\max\{3,9,5,2\}$ 'ye bakılırken, bir sonraki adımda ise $\max\{4,3,7,1\}$ dikkate alınır. Adım uzunluğunun 2 olduğu 2×2 'lik bir maksimum havuzlama sonrası görüntünün uzunluk ve genişliği ikiye bölünmüştür. SegNet'in kod çözücüsünün yukarı örnekleme bölümünde ise girdinin boyutunun iki katına çıkarılması amaçlanır. Örneğin 2×2 'lik bir öznitelik haritasının 4×4 'lük hale gelmesi sırasında ilgili kodlayıcının maksimum havuzlama bölümünde seçilen maksimum değerlerin indislerinden yararlanır. Şekil 25'teki örnekte, ilk 2×2 'lik alan için maksimum değer olan 9, (1,0) indisindedir. Dolayısıyla ilgili katmanlardan geçtikten sonra kod çözücüye gelen 2×2 'lik girdinin ilk değeri 4×4 'lük çıktıda, ilk 2×2 'lik alanın (1,0) indisine yazılmalıdır. Bu işlem girdinin tüm pikselleri için uygulanmalıdır.

SegNet'te konvolüsyon katmanları hem kodlayıcı hem de kod çözücü bölümünde yer almaktadır. Mekanik olarak aynı çalıştıkları halde işlevleri farklıdır. Kodlayıcı aşamasında, konvolüsyon katmanı yerel öznitelikleri çıkarır ve bunları maksimum havuzlama katmanına iletir. Bu katmanda da boyut indirgemesi yapılır. Dolayısıyla kodlayıcı kısmındaki Konvolüsyon ve Havuzlama katmanı öznitelik çıkarmanın nasıl olacağını öğrenmektedir. Diğer taraftan, kod çözücü aşamasında, girdi öznitelik haritası konvolüsyon katmanlarına ulaşmadan önce yukarı örneklenir. 2×2 'lik bir yukarı örneklenmiş alanda sadece bir değer önceki katmandan gelirken diğer piksel hücreleri boştur. Bu boş hücreler daha sonra



Şekil 25. Maksimum havuzlama ve havuzlama indis değerlerine göre yukarı örnekleme yapılması

konvolüsyon katmanı tarafından doldurulacaktır. Dolayısıyla, kod çözücü kısmındaki Yukarı Örnekleme ve Konvolüsyon katmanı öznitelik haritasına nasıl öznitelik ekleneceğini öğrenmektedir. Hem kodlayıcı hem de kod çözücü aşamalarında kullanılan konvolüsyon hesaplama işlemleri aynı olmasına rağmen, belirtilen nedenlerden dolayı yapılan işlevler farklıdır. Bu nedenle literatürde kodlayıcı tarafındaki konvolüsyonların ismi değişmezken, kod çözücü tarafındaki konvolüsyon işlemleri “ters konvolüsyon” (deconvolution) veya “transpoze konvolüsyon” (transposed convolution) şeklinde adlandırılmaktadır [189].



3. BULGULAR VE İRDELEME

3.1. Kullanılan Değerlendirme Ölçekleri

Çıktının iki veya daha fazla sınıf olabileceği makine öğrenmesi sınıflandırma problemlerinde performans metriklerinin bulunabilmesi için Tablo 11’de gösterilen Hata Matrisinden (Confusion Matrix) yararlanır.

Tablo 11. Hata matrisi tablosu

		Gerçek Olan Sınıf	
		Pozitif	Negatif
Tahmin Edilen Sınıf	Pozitif	Doğru Pozitif (True Positive - TP)	Yanlış Pozitif (False Positive - FP) Tip 1 Hata
	Negatif	Yanlış Negatif (False Negative - FN) Tip 2 Hata	Doğru Negatif (True Negative - TN)

Hata matrisi tahmin edilen ve gerçek değerler için 4 farklı kombinasyonu içeren bir tablodur. Bu tabloda tahmin değerleri Pozitif ve Negatif şeklinde, gerçek değerler ise Doğru ve Yanlış olarak tanımlanmaktadır. Doğru Pozitif (TP) gerçekte pozitif olan örneklerden kaçının doğru bir şekilde yani pozitif olarak sınıflandırıldığını, Doğru Negatif (TN) ise gerçekte negatif olan örneklerden kaçının doğru bir şekilde yani negatif olarak sınıflandırıldığını gösterir. Diğer taraftan, Yanlış Pozitif (FP) gerçekte negatif olan örneklerden kaçının yanlış bir şekilde yani pozitif olarak sınıflandırıldığını, Yanlış Negatif (FN) ise gerçekte pozitif olan örneklerden kaçının yanlış bir şekilde yani negatif olarak sınıflandırıldığını gösterir.

Tablo 12, hata matrisinin değerlerine dayanarak ikili sınıflandırma için en sık kullanılan performans metriklerini göstermektedir. Çoklu sınıflandırma için sıkça kullanılan metriklerle ilişkin bilgiler ise Tablo 13’te gösterilmektedir. Tekil bir C_i sınıfı için ölçümler TP_i , FN_i , TN_i ve FP_i kullanılarak yapılır. $Doğruluk_i$, $Hassasiyet_i$, $Duyarlılık_i$; C_i ’ye ilişkin frekanslar kullanılarak hesaplanır.

Tablo 12. İkili sınıflandırma için kullanılan performans metrikleri (Tablo 11’de kullanılan notasyonlara göre) [95]

Metrik	Formül	Değerlendirme Odağı
Doğruluk (Accuracy)	$\frac{TP + TN}{TP + FN + FP + TN}$	Sınıflandırıcının genel verimliliği (doğru tahmin yapma seviyesi)
Hassasiyet veya Kesinlik (Precision)	$\frac{TP}{TP + FP}$	Pozitif olarak tahmin edilen etiketlerdeki doğruluk seviyesi
Duyarlılık (Recall veya Sensitivity)	$\frac{TP}{TP + FN}$	Pozitif durumun sınıflandırıcı tarafından doğru bir şekilde tahmin edilebilme seviyesi
F-skoru (F-score)	$\frac{(\beta^2 + 1)TP}{(\beta^2 + 1)TP + \beta^2 FN + FP}$	Hem hassasiyetin hem de duyarlılığın hesaba katıldığı performans seviyesi
Özgüllük (Specificity)	$\frac{TN}{FP + TN}$	Negatif durumun sınıflandırıcı tarafından doğru bir şekilde tahmin edilebilme seviyesi
Eğri Altındaki Alan (Area Under Curve-AUC)	$\frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right)$	Sınıflandırıcının yanlış sınıflandırmadan kaçınma yeteneği

β , hassasiyet ve duyarlılık arasındaki dengeyi belirlemek için kullanılan pozitif bir sayıdır.

Genel sınıflandırmanın kalitesi genellikle iki şekilde değerlendirilir. İlk ölçme şekli makro-ortalama olarak adlandırılır ve C_1, \dots, C_l için ölçülmüş aynı metriğe ilişkin değerlerin ortalamasının alınması ile bulunur. İkinci ölçme şekli olan mikro-ortalama ise birikimli TP , FN , TN ve FP 'yi elde etmek için frekans toplamlarına bakılır ve buna dayalı bir performans ölçüsü elde edilir [95]. Tablo 12 ve Tablo 13'te F skoru hesaplanırken kullanılan β katsayısı, hassasiyet ve duyarlılık arasındaki dengeyi belirlemek için kullanılan pozitif bir sayıdır. $0 < \beta < 1$ aralığında bir β seçildiğinde *hassasiyete*, $1 < \beta < \infty$ aralığında bir β seçildiğinde ise *duyarlılığa* daha yüksek ağırlık verilmiş olmaktadır. $\beta = 1$ alındığında hesaplanan F-skoru hesaplamak için kullanılan formül harmonik ortalama formülüne eşit olmaktadır ve bu şekilde hesaplanan F skoruna literatürde *F1-skoru* adı verilmektedir.

Literatürde makro-ortalama F-skorunu hesaplamak için iki yaygın formül kullanılmaktadır. Bunların ilkinde [190; 191] her bir sınıf için hassasiyet ve duyarlılık değerleri hesaplanmakta, daha sonra Tablo 12'deki F-skoru hesaplama formülü kullanılarak sınıfa ilişkin F-skoru elde edilmektedir. Her bir sınıfa ilişkin F-skorlarının ortalaması alınarak da makro-ortalama F-skoru elde edilmektedir. İkinci yöntemde [95] ise, Tablo 13'te belirtilen makro-ortalama hassasiyet ve makro-ortalama duyarlılık formülleri kullanılarak sınıflandırmaya ilişkin ilgili değerler hesaplanmakta ve bu değerler Tablo 13'te belirtilen makro-ortalama F-skoru formülünde yerine konularak istenilen F-skora ulaşılmaktadır.

Tablo 13. Tablo 12'deki ölçütlerin birçok sınıf için genelleştirilmesine dayanan çok sınıflı sınıflandırma için performans metrikleri [95]

Metrik	Formül	Değerlendirme Odağı
Ortalama Doğruluk	$\frac{\sum_{i=1}^l \frac{TP_i + TN_i}{TP_i + FN_i + FP_i + TN_i}}{l}$	Bir sınıflandırıcının ortalama sınıf başına verimliliği (doğru tahmin yapma seviyesi)
Hata Oranı	$\frac{\sum_{i=1}^l \frac{FP_i + FN_i}{TP_i + FN_i + FP_i + TN_i}}{l}$	Sınıf başına ortalama sınıflandırma hatası
Hassasiyet _μ (Precision _μ)	$\frac{\sum_{i=1}^l TP_i}{\sum_{i=1}^l (TP_i + FP_i)}$	Tahmin edilen veri sınıfı etiketlerinin, örnek başına alınan kararların toplamından hesaplanması durumunda, gerçek sınıf etiketleriyle olan uyumu
Duyarlılık _μ (Recall _μ)	$\frac{\sum_{i=1}^l TP_i}{\sum_{i=1}^l (TP_i + FN_i)}$	Tahmin edilen veri sınıfı etiketlerinin, örnek başına alınan kararların toplamından hesaplanması durumunda, sınıflandırıcının sınıf etiketlerini tanımlayabilme verimliliği
F-skoru _μ (F-score _μ)	$\frac{(\beta^2 + 1)Hassasiyet_{\mu} \cdot Duyarlılık_{\mu}}{\beta^2 Hassasiyet_{\mu} + Duyarlılık_{\mu}}$	Örnek başına kararların toplamına dayalı sınıflandırıcı tarafından tahmin edilen etiketler ile verinin pozitif etiketleri arasındaki ilişki
Hassasiyet _M (Precision _M)	$\frac{\sum_{i=1}^l \frac{TP_i}{TP_i + FP_i}}{l}$	Pozitif olarak tahmin edilen etiketlerdeki doğruluk seviyesinin sınıf başına ortalaması
Duyarlılık _M (Recall _M)	$\frac{\sum_{i=1}^l \frac{TP_i}{TP_i + FN_i}}{l}$	Pozitif durumun sınıflandırıcı tarafından doğru bir şekilde tahmin edilebilme seviyesinin sınıf başına ortalaması
F-skoru _M (F-score _M)	$\frac{(\beta^2 + 1)Hassasiyet_M \cdot Duyarlılık_M}{\beta^2 Hassasiyet_M + Duyarlılık_M}$	Her bir sınıf ortalamasına göre sınıflandırıcı tarafından tahmin edilen etiketler ile verinin pozitif etiketleri arasındaki ilişki

β , hassasiyet ve duyarlılık arasındaki dengiyi belirlemek için kullanılan pozitif bir sayıdır. C_i , ($i = 1, \dots, l$) sınıflar; TP_i , C_i için Doğru Pozitiflerin sayısı; FP_i , C_i için Yanlış Pozitiflerin sayısı; FN_i , C_i için Yanlış Negatiflerin sayısı; TN_i , C_i için Doğru Negatiflerin sayısı; μ ve M indisleri sırasıyla mikro ve makro ortalamalar

Tablo 12'de belirtilen performans metriklerine ek olarak literatürde ikili ve çoklu sınıflandırma için yaygın olarak kullanılan performans metriklerinden biri de Cohen'in kappa katsayısıdır [192]. Kappa katsayısı genellikle aynı nesneyi derecelendiren iki gözlemci arasındaki uyumu test etmek amacıyla kullanılmaktadır. Fakat ikiden çok gözlemci arasındaki uyumun ölçülmesi için de uygun bir istatistiktir. Sınıflandırma amaçlı kullanıldığında Kappa istatistiği, gerçek kategoriler ile tahmin edilen kategoriler arasındaki uyumun bir ölçütü olarak işlev görmektedir. C_{ij} ile hata matrisinde her bir hücreye düşen

eleman sayısı gösterildiğinde, hata matrisindeki tüm elemanların sayısı (S), Eşitlik 53 kullanılarak hesaplanmaktadır.

$$S = \sum_{i=1}^N \sum_{j=1}^N C_{ij} \quad (53)$$

Eşitlik 54'teki C_i ile hata matrisindeki i 'inci satır toplamı, Eşitlik 55'teki C_i ile ise i 'inci sütun toplamı gösterildiğinde, Cohen'in Kappa katsayısı Eşitlik 56 ile bulunmaktadır.

$$C_i = \sum_{j=1}^N C_{ij} \quad ; \quad i = 1, \dots, N \quad (54)$$

$$C_i = \sum_{j=1}^N C_{ji} \quad ; \quad i = 1, \dots, N \quad (55)$$

$$K = \frac{P_o - P_e}{1 - P_e} \quad (56)$$

Eşitlik 56'da P_o , gerçekte olan ile tahmin edilen kategoriler arasındaki uyum oranını, P_e ise şansa bağlı uyum oranını göstermektedir. Bu değerler Eşitlik 57 ve Eşitlik 58 kullanılarak hesaplanmaktadır.

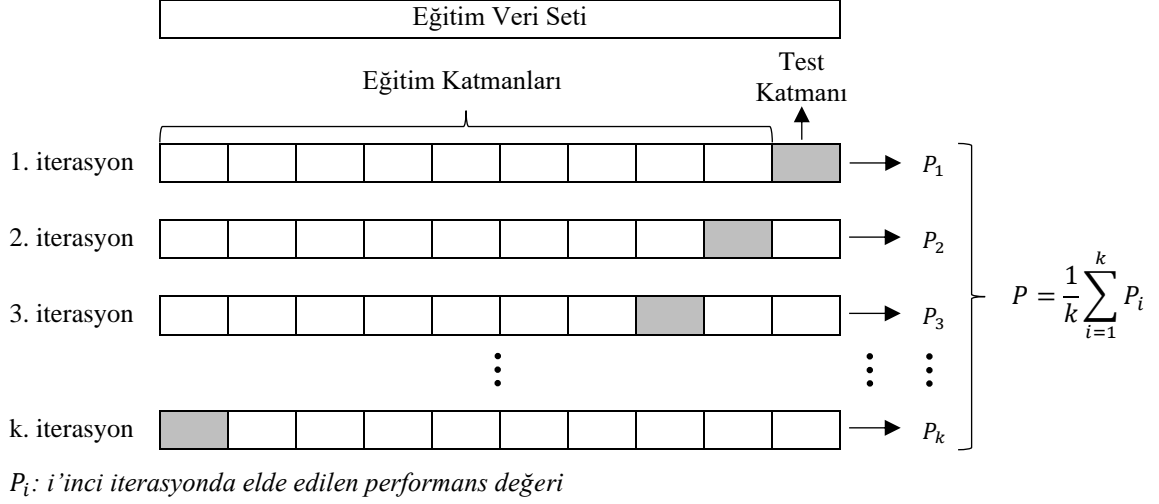
$$P_o = \frac{1}{S} \sum_{i=1}^N C_{ii} \quad (57)$$

$$P_e = \frac{1}{S^2} \sum_{i=1}^N C_i \cdot C_i \quad (58)$$

Kappa katsayısı -1 ile +1 arasında değişiklik göstermektedir. -1, negatif tam uyumu; 0, uyumsuzluğu; +1 ise pozitif tam uyumu belirtmektedir.

3.2. k-Katmanlı Çapraz Doğrulama

Bir sınıflandırıcı eğitilirken olası ezberleme sorunundan kaçınmak ve eğitilen modelin daha önceden görmediği veri seti üzerinde nasıl bir performans göstereceğini anlamak için



Şekil 26. k-katmanlı çapraz doğrulama yöntemi

veri seti eğitim ve test setleri olmak üzere ayrılmaktadırlar. Fakat modelin eğitim ve test aşamasında dağılımdan kaynaklı hatalar olabilmektedir. Bu hataları minimize edebilmek için k-Katmanlı Çapraz Doğrulama tekniği kullanılmaktadır. Bu teknikte, eğitim veri seti k parçaya bölünmekte, $k-1$ parça eğitim için kullanılırken 1 parça test için kullanılmaktadır. Bu işlem k defa tekrar edilmekte ve her iterasyonda elde edilen performans değerlerinin ortalaması alınarak modelin performansı değerlendirilmektedir. Şekil 26'da k-katmanlı çapraz doğrulama yönteminin nasıl çalıştığı gösterilmektedir. Şekilde veri seti k parçaya bölünmüştür. Her iterasyonda, boyalı alan test setine ayrılırken kalan parçalar eğitim için kullanılmaktadır. Her iterasyonun sonunda sınıflandırıcıdan gelen performans puanları (P_i) kaydedilmekte ve bütün iterasyonlar bittiğinde P_i değerlerinin aritmetik ortalaması olan P , modelin genel performans değeri olarak alınmaktadır [193].

Sunulan tezde, sınıflandırma yöntemlerinin performanslarının ölçülmesi aşamasında mikro ve makro-ortalama hassasiyet, mikro ve makro-ortalama duyarlılık, mikro ve makro-ortalama F-skoru, mikro ve makro-ortalama kappa metrikleri kullanılmıştır. Sınıflandırıcılar 10-katmanlı çapraz doğrulama yöntemi kullanılarak eğitilmiştir.

$HM(i)_j$ ($i = 1, \dots, 10; j = \{hassasiyet, duyarlılık, kappa\}$) ile 10-katmanlı çapraz doğrulamanın i 'inci katmandaki hata matrisinden elde edilen j performans değeri gösterildiğinde, makro-ortalama hassasiyet, makro-ortalama duyarlılık ve makro-ortalama F-skoru Eşitlik 59-61 kullanılarak, makro-ortalama kappa ise Eşitlik 62 kullanılarak hesaplanmıştır. $HM(i)_j$ hesaplanırken Tablo 13'teki ilgili formüller ve Eşitlik 56 kullanılmıştır.

$$\text{Makro-Ortalama Hassasiyet} = \frac{HM1_{hassasiyet} + \dots + HM10_{hassasiyet}}{10} \quad (59)$$

$$\text{Makro-Ortalama Duyarluluk} = \frac{HM1_{duyarlılık} + \dots + HM10_{duyarlılık}}{10} \quad (60)$$

$$\text{Makro-Ort. F-skoru} = \frac{2 \times \text{Makro-Ort. Has.} \times \text{Makro-Ort. Duy.}}{\text{Makro-Ort. Has.} + \text{Makro-Ort. Duy.}} \quad (61)$$

$$\text{Makro-Ort. Kappa} = \frac{HM1_{kappa} + \dots + HM10_{kappa}}{10} \quad (62)$$

10-katmanlı çapraz doğrulamada mikro-ortalama metrikleri hesaplanırken, çapraz doğrulamanın her bir katmanında elde edilen hata matrisleri ($HM(i)$ ($i = 1, \dots, 10$)) bir araya getirilerek toplanmış ve toplam hata matrisi elde edilmiştir (Eşitlik 63). Bu matris kullanılarak sınıflandırılacak her bir sınıf için (deri, yanık ve arka plan) hassasiyet ve duyarlılık değerleri hesaplanmış, sınıflara ilişkin hassasiyetlerin ortalaması alınarak mikro-ortalama hassasiyet (Eşitlik 64), sınıflara ilişkin duyarlılıkların ortalaması alınarak mikro-ortalama duyarlılık (Eşitlik 65) değerleri elde edilmiştir. Mikro-ortalama F-skoru ise Eşitlik 66 kullanılarak hesaplanmıştır [95]. Toplam hata matrisine (HM) ilişkin değerler kullanılarak Eşitlik 56 yardımıyla mikro-ortalama kappa değeri elde edilmiştir.

$$HM = \sum_{i=1}^{10} HM(i) \quad (63)$$

$$\text{Mikro-Ort. Has.} = \frac{\text{Deri}_{hassasiyet} + \text{Yanık}_{hassasiyet} + \text{Arka plan}_{hassasiyet}}{3} \quad (64)$$

$$\text{Mikro-Ort. Duy.} = \frac{\text{Deri}_{duyarlılık} + \text{Yanık}_{duyarlılık} + \text{Arka plan}_{duyarlılık}}{3} \quad (65)$$

$$\text{Mikro-Ort. F-skoru} = \frac{2 \times \text{Mikro-Ort. Has.} \times \text{Mikro-Ort. Duy.}}{\text{Mikro-Ort. Has.} + \text{Mikro-Ort. Duy.}} \quad (66)$$

3.3. İleriye Doğru Seçim

Sunulan tezde, öznelik seçimi aşamasında İleriye Doğru Seçim (Forward Selection) yaklaşımı kullanılmıştır. Bu yaklaşım, mevcut özneliklerin herhangi birinin seçimi ile

başlar ve her turda kullanılmayan özniteliklerden biri bu seçime eklenir. Eklenen her öznitelik için, belirlenen sınıflandırma yönteminin performansı 10-katmanlı çapraz doğrulama kullanılarak tahmin edilir ve daha sonra performansı en yüksek seviyeye çıkaran öznitelik bulunup seçime eklenir. Ardından, yeni bir tur başlatılır ve performans artışı olduğu sürece iterasyonlar sürdürülür. En yüksek başarıya ulaşan öznitelik grubu en iyi öznitelik alt kümesi olarak belirlenir [194].

3.4. Doku Örneklerinin Sınıflandırılmasına İlişkin Sonuçlar

Bölüm 1.10’da belirtildiği üzere, sunulan tezde kullanılan veri seti 5 alt veri setine bölünmüş ve her bir alt veri setinden 450’şer adet deri, yanık ve arka plan örneği rastgele olarak seçilmiştir. Dolayısıyla, eğitim öncesinde deri, yanık ve arka plan sınıflarının her birini temsil etmek için 2250 adet 64x64 piksel boyutunda doku örneği bulunmaktadır.

Bölüm 2.3’te teorik çalışma prensipleri verilen 10 adet sınıflandırıcı, Bölüm 2.1’de çıkartılan öznitelikler kullanılarak oluşturulan Öznitelik Matrisi 1 (ÖM-1), Öznitelik Matrisi 2 (ÖM-2) ve Öznitelik Matrisi 3 (ÖM-3) ile eğitilmiştir. Veri setinde yer alan toplamda 6750 adet doku örneği, eğitilen bu sınıflandırıcılar tarafından sınıflandırılmıştır. ÖM-1, Bölüm 2.1.1’de gri seviye histogram istatistikleri kullanılarak çıkartılan 49 adet öznitelik ile Bölüm 2.1.2’de GLCM kullanılarak elde edilen 364 Haralick doku özneliğinin birleşiminden oluşan 413 özneliği içermektedir. ÖM-2 ise ÖM-1’e ileriye doğru seçim uygulandıktan sonra elde edilen 9 özneliği içermektedir (ÖM-2’nin oluşturulmasına ilişkin detaylı bilgi Bölüm 3.4.2’de verilecektir). Son olarak ÖM-3, Bölüm 2.1.3’te renk kanallarının ortalama parlaklık değerlerinin kullanılması ile elde edilen 13 özneliği içermektedir.

3.4.1. ÖM-1 ile Eğitilmiş Sınıflandırıcıların Performans Sonuçları

Bölüm 2.3’te verilen 10 adet sınıflandırıcının 10-katmanlı çapraz doğrulama ortamında ÖM-1 ile eğitilmesi sonucu oluşan performans ölçütleri Tablo 14’te gösterilmektedir. Sınıflandırıcılar F-skoru performanslarına göre sıralandığında, sinir ağına dayalı sınıflandırıcıların diğer sınıflandırıcılardan daha başarılı olduğu görülmektedir. Ele alınan tüm performans metrikleri açısından en iyi sonucu, geri yayılım algoritması tarafından eğitilmiş çok katmanlı ileri beslemeli YSA’nın verdiği görülmüştür. Bu modele ilişkin mikro ve makro-ortalama kappa istatistiği 0.9191 olarak hesaplanmıştır. 1’e oldukça yakın olan bu değer sınıflandırma performansının oldukça yüksek olduğunu göstermektedir.

Tablo 14. ÖM-1 ile eğitilmiş sınıflandırıcıların performans sonuçlarının karşılaştırılması

Öğrenme Algoritması (Sınıflandırıcı)	Mikro/Makro Ort. Duyarlılık	Mikro-Ort. Hassasiyet	Makro-Ort. Hassasiyet	Mikro-Ort. F-Skoru	Makro-Ort. F-Skoru
YSA	94.61	94.67	94.70	94.64	94.65
DÖ	94.19	94.21	94.26	94.20	94.22
GBT	92.92	93.01	93.03	92.96	92.97
AutoMLP	90.33	90.64	90.72	90.48	90.52
KT	89.99	90.09	90.15	90.04	90.07
RO	89.35	90.11	90.14	89.73	89.74
GLM	89.21	89.60	89.64	89.40	89.42
kNN	84.80	84.93	84.96	84.86	84.88
NB	79.69	82.42	82.43	81.03	81.04
KA	79.38	79.26	73.90	79.32	76.54

Tablodaki değerler yüzdeler (%) olarak verilmiştir.

Tablo 15. ÖM-1 ile eğitilmiş YSA modeli kullanılarak 10-katmanlı çapraz doğrulama sonucunda elde edilen hata matrisi

		Gerçek			
		Deri	Yanık	Arka Plan	Toplam
Tahmin	Deri	2141	118	81	2340
	Yanık	64	2109	33	2206
	Arka Plan	45	23	2136	2204
	Toplam	2250	2250	2250	6750

YSA modelinin 10-katmanlı çapraz doğrulanması sonrasında elde edilen hata matrisini gösteren Tablo 15 incelendiğinde, sınıflandırıcının doku örneklerini sınıflandırırken tahmin seçeneği olarak en çok deri sınıfını tercih ettiği görülmektedir. Bu tercihler, deri sınıfı için hassasiyet ölçütünde artışa neden olmakla beraber, yanlış olarak tahmin edilen örnekler nedeniyle duyarlılık ölçütünde azalma meydana gelmektedir. Bununla birlikte, tabloda göze çarpan ikinci nokta, gerçekte yanık olmasına rağmen deri olarak tahmin edilen doku örneklerinin sayısıdır. Bu durumun özellikle 1. derece yanık bölgelerinden elde edilen doku örneklerinin deri örneklerine olan benzerliğinden kaynaklandığı düşünülmektedir.

3.4.2. ÖM-2 ile Eğitilmiş Sınıflandırıcıların Performans Sonuçları

En iyi sınıflandırıcı yöntemi bulunduktan sonraki ilk adım, daha az öznitelikle eğitilmiş basit bir modelin daha iyi veya eşdeğer bir performans sağlayıp sağlamayacağını belirlemek için bir ileriye doğru seçim algoritması gerçekleştirmektir. Bu amaçla, ÖM-1 ile eğitilmiş sınıflandırıcılardan en yüksek performansa sahip olan YSA modeli, 10-katmanlı çapraz doğrulama ortamında ileriye doğru seçme algoritmasına tabi tutulmuş ve veri setinde yer alan 6750 doku örneğini en yüksek performansla tahmin edecek en az sayıda öznitelik kümesi tespit edilmiştir. Tablo 16’da ileriye doğru seçim sonrasında elde edilen 9 öznitelik gösterilmektedir. Bu öznitelikler incelendiğinde, eğitim aşamasında Haralick özniteliklerinin, gri seviye histogramlardan çıkartılan istatistiksel özniteliklerden daha başarılı olduğu açıktır. Seçilen özniteliklerin tamamı Haralick özniteliği olmakla beraber $L^*a^*b^*$ renk uzayının ve özellikle a^* kanalının eğitim aşamasında ayırt edicilik etkisinin yüksek olduğu görülmüştür. Bununla birlikte, Tablo 4’te verilen Haralick özniteliklerinden f_4 ile gösterilen *varyans* ve f_6 ile gösterilen *toplam ortalama* da ileriye doğru seçimde öne çıkan öznitelikler olarak dikkat çekmektedir. Seçilen özniteliklerden büyük çoğunluğunun 0° komşuluk kullanılarak oluşturulan GLCM yardımıyla hesaplandığı da görülmektedir.

İleriye doğru seçimde 10-katmanlı çapraz doğrulama ile YSA modelinin geçerliliğinin onaylanması yapıldıktan sonra elde edilen hata matrisi Tablo 17’de verilmektedir. Öznitelik seçiminden sonra hesaplanan performans değerleri ve bu değerlerin Bölüm 3.4.1’de elde edilen sonuçlarla karşılaştırılması Tablo 18’de sunulmaktadır.

Tablo 16. İleriye doğru seçim algoritması sonrası elde edilen öznitelikler

	Renk Uzayı	Renk Kanalı	GLCM Derecesi	Seçilen Öznitelikler
1	RGB	R	45°	Haralick f_2
2	RGB	R	135°	Haralick f_{12}
3	RGB	G	90°	Haralick f_4
4	$L^*a^*b^*$	a^*	0°	Haralick f_5
5	$L^*a^*b^*$	a^*	0°	Haralick f_8
6	$L^*a^*b^*$	a^*	45°	Haralick f_6
7	$L^*a^*b^*$	a^*	90°	Haralick f_4
8	$L^*a^*b^*$	b^*	0°	Haralick f_6
9	$L^*a^*b^*$	b^*	0°	Haralick f_{10}

Tablo 17. ÖM-2 ile eğitilmiş YSA modeli kullanılarak 10-katmanlı çapraz doğrulama sonucunda elde edilen hata matrisi

		Gerçek			
		Deri	Yanık	Arka Plan	Toplam
Tahmin	Deri	2036	166	156	2358
	Yanık	99	2065	44	2208
	Arka Plan	115	19	2050	2184
	Toplam	2250	2250	2250	6750

Tablo 17 incelendiğinde, sadece seçili özneliklerin kullanımı ile elde edilen sınıflandırmada, sınıflandırıcının doku örneklerini sınıflandırırken tahmin seçeneği olarak Tablo 15’te olduğu gibi yine en çok deri sınıfını tercih ettiği görülmektedir. Bununla birlikte, gerçekte yanık olmasına rağmen deri olarak tahmin edilen ve gerçekte arka plan olmasına rağmen deri olarak tahmin edilen doku örneklerinin sayısı da artış göstermiştir. Bir diğer göze çarpan nokta ise gerçekte deri olan doku örneklerinden yanlış sınıflandırma ile arka plan olarak tahmin edilenlerin sayısındaki yükseliştir.

Tablo 18. ÖM-1 ve ÖM-2 ile eğitilmiş YSA sınıflandırıcılarının performans sonuçlarının karşılaştırılması

Öğrenme Algoritması (YSA)	Mikro/Makro Ort. Duyarlılık	Mikro Ort. Hassasiyet	Makro Ort. Hassasiyet	Mikro Ort. F-Skoru	Makro Ort. F-Skoru	Mikro/Makro Ort. Kappa
	ÖM-1 ile Eğitilmiş	94.61	94.67	94.70	94.64	94.65
ÖM-2 ile Eğitilmiş	91.13	91.24	91.26	91.18	91.19	86.69

Tablodaki değerler yüzdeler (%) olarak verilmiştir.

Tablo 18’den görülebileceği gibi hem ÖM-1 hem de ÖM-2 ile eğitilmiş YSA sınıflandırıcılarının performans değerleri oldukça yüksektir. Her ne kadar ileriye doğru seçim sonrasında elde edilen özneliklerden oluşan ÖM-2 ile eğitilen modelin performansı biraz daha kötü olsa da, öznelik seçim işleminden sonra, toplam öznelik sayısı 413’ten 9’a düşmüş, daha basit ve yüksek performanslı bir model elde edilmiştir. Ayrıca, hesaplama süreleri, özellikle test görüntüsünün fazla sayıda bölgeye bölündüğü ve her bir bölge için özneliklerin hesaplandığı SLIC için önemli ölçüde azaltılmıştır. Sınıflandırılmak istenen 1024x768 boyutundaki bir görüntüde, ÖM-1 ve ÖM-2’nin değişen küme sayılarına göre

hesaplama sürelerinde meydana gelen değişiklik Intel Core i7-6700HQ CPU, 20 GB RAM ve 512 GB SSD HDD ile donatılmış bir bilgisayarda gözlemlenmiştir. Bulunan sonuçlar Tablo 19’da gösterilmektedir. Tablodan görüldüğü üzere, ÖM-1’e ilişkin hesaplama süreleri ÖM-2’ye ilişkin hesaplama sürelerinin üç katından daha fazladır. Sunulan tezde, yeterince yüksek performansa sahip olması ve hesaplama süresindeki verimi nedeniyle, test görüntülerini sınıflandırmak için sadece ÖM-2 ile eğitilmiş YSA sınıflandırıcısı kullanılmıştır.

Tablo 19. ÖM-1 ve ÖM-2’nin hesaplama sürelerinin 1024x768 boyutunda görüntüler üzerinde değişen küme sayılarına göre karşılaştırılması (baz alınarak)

Küme Sayısı	ÖM-2’ye İlişkin Hesaplama Süresi (Saniye)	ÖM-1’e İlişkin Hesaplama Süresi (Saniye)
5	5.61	18.48
15	15.10	49.00
25	23.96	82.68
250	241.39	795.60
500	491.34	1579.76

3.4.3. ÖM-3 ile Eğitilmiş Sınıflandırıcıların Performans Sonuçları

Bölüm 2.1.3’te belirtildiği gibi, bazı sınıflandırma işlemlerinde tek başına renk özelliğinin ayırt edicilikte yeterli olabileceğine inanılmaktadır [195]. Bu noktadan yola çıkarak renk uzaylarına ait renk kanallarının ortalama piksel değerlerini içeren ÖM-3 oluşturulmuş, Bölüm 2.3’te verilen 10 adet sınıflandırıcının 10-katmanlı çapraz doğrulama ortamında bu öznitelik matrisi ile eğitilmesi sonucu oluşan performans ölçütleri Tablo 20’de gösterilmiştir. Sınıflandırıcılar F-skoru performanslarına göre sıralandığında, YSA yönteminin en iyi sonuçları elde ettiği gözlemlenmektedir. Bu modele ilişkin mikro ve makro-ortalama kappa istatistiği 0.8227 olarak hesaplanmıştır. Performans sonuçları her ne kadar Bölüm 3.4.1 ve Bölüm 3.4.2’de elde edilen sonuçlara oranla daha düşük olsa da, kullanılan özniteliklerin sayısının ve hesaplama yükünün düşük olması ÖM-3 ile eğitilen YSA sınıflandırıcısını test görüntüleri üzerinde denemeye değer kılmıştır. ÖM-3 ile eğitilmiş YSA modeli kullanılarak 10-katmanlı çapraz doğrulama sonucunda elde edilen hata matrisi Tablo 21’de gösterilmektedir.

Tablo 20. ÖM-3 ile eğitilmiş sınıflandırıcıların performans sonuçlarının karşılaştırılması

Öğrenme Algoritması (Sınıflandırıcı)	Mikro/Makro Ort. Duyarlılık	Mikro-Ort. Hassasiyet	Makro-Ort. Hassasiyet	Mikro-Ort. F-Skoru	Makro-Ort. F-Skoru
YSA	88.18	88.50	88.53	88.34	88.35
GBT	87.44	88.03	88.07	87.73	87.75
AutoMLP	86.18	86.71	86.79	86.44	86.48
DÖ	85.59	86.21	86.36	85.90	85.97
KT	84.87	84.84	84.89	84.85	84.88
RO	82.89	85.62	85.66	84.23	84.25
KA	82.04	85.59	85.68	83.78	83.82
GLM	82.68	83.93	83.98	83.30	83.32
kNN	82.43	82.66	82.69	82.54	82.56
NB	78.64	79.83	79.88	79.23	79.26

Tablodaki değerler yüzdeler (%) olarak verilmiştir

Tablo 21. ÖM-3 ile eğitilmiş YSA modeli kullanılarak 10-katmanlı çapraz doğrulama sonucunda elde edilen hata matrisi

		Gerçek			
		Deri	Yanık	Arka Plan	Toplam
Tahmin	Deri	1998	291	169	2458
	Yanık	165	1919	46	2130
	Arka Plan	87	40	2035	2162
	Toplam	2250	2250	2250	6750

Tablo 21'deki sınıflandırma hataları, Tablo 15 ve Tablo 17'de yapılan hatalar ile karşılaştırıldığında yanlış sınıflandırmaların genellikle aynı odaklı olduğu görülmektedir. Sınıflandırıcının performansını düşüren en temel neden gerçekte yanık olan doku örneklerinin kayda değer bir kısmının deri olarak sınıflandırılmış olmasıdır. Ayrıca, daha önceki sınıflandırıcılarda olduğu gibi ÖM-3 ile eğitilen YSA da doku örneklerini sınıflandırırken tahmin seçeneği olarak en çok deri sınıfını tercih etmektedir. Fakat deri sınıfına ilişkin hassasiyet değerinin diğer sınıflarına ilişkin hassasiyet değerlerinden oldukça düşük olduğu görülmektedir.

3.5. Test Görüntülerinin Segmentasyon ve Sınıflandırma Sonuçları

Bu bölümde, ÖM-2 ve ÖM-3 ile eğitilen YSA sınıflandırıcılarının veri setinde yer alan 105 görüntüyü sınıflandırma performanslarını belirleyebilmek için 5-katmanlı bir çapraz doğrulama ortamı tasarlanmıştır. Bölüm 1.10'da belirtildiği gibi, veri setindeki görüntüler rastgele olarak beş gruba ayrılmış ve VS-1, VS-2, VS-3, VS-4 ve VS-5 alt veri setleri oluşturulmuştur. Her bir alt veri setine ilişkin görüntü, kendisi dışındaki alt veri setlerinden toplanan doku örneklerinin eğitimi ile oluşturulan modellere göre sınıflandırılmıştır. Örneğin, VS-1'deki görüntülerin sınıflandırılmasında, VS-2, VS-3, VS-4 ve VS-5'ten toplanan doku örneklerinin eğitimi ile oluşturulan modeller kullanılmıştır. Dolayısıyla her bir alt veri setini test etmek için 2, toplamda ise 10 adet ANN modeli oluşturulmuştur. VS-1'de yer alan bir test görüntüsünün analizi sırasında uygulanan adımlar şu şekildedir:

- 1) *RGB* renk uzayındaki test görüntüsü *CIE L*a*b**, *HSV* ve *YCbCr* renk uzaylarına çevrilir.
- 2) Calinski-Harabasz kriterinin; k-Ortalamalar, FCM, BM ve SLIC kümeleme yöntemleri ile birlikte kullanımı sonrasında test görüntüsü için en uygun küme sayısı belirlenir. Bu işlem yapılırken kümeleme algoritmalarında girdi verisi olarak kullanılan renk kanalları şu şekildedir:
 - k-Ortalamalar: *CIE L*a*b** renk uzayının *a** ve *b**, *HSV* renk uzayının *H* ve *S*, *YCbCr* renk uzayının *Cb* ve *Cr* renk kanalları
 - FCM: *CIE L*a*b** renk uzayının *a** ve *b**, *HSV* renk uzayının *H* ve *S*, *YCbCr* renk uzayının *Cb* ve *Cr* renk kanalları
 - BM: *YCbCr* renk uzayının *Cr* ve *CIE L*a*b** renk uzayının *a** kanalı
 - SLIC: *CIE L*a*b** renk uzayının *a** ve *b** kanalları
- 3) Her bir kümeleme yöntemi için, belirlenen küme sayısına ve kullanılan renk kanalına göre test görüntüsü bölgelere ayrılır.
- 4) Test görüntüsünde yer alan her bir bölgeyi sınıflandıracak YSA modelleri elde edilir.
 - VS-2, VS-3, VS-4 ve VS-5'in her birinden alından 450'şer adet deri, yanık ve arka plan doku örnekleri üzerinden ÖM-2 ve ÖM-3 hesaplanır. Bu matrisler kullanılan özniteliklere ek olarak ilgili doku örneğine ilişkin etiket değerini de içermektedir.

- ÖM-2 ve ÖM-3, 10-katmanlı çapraz doğrulama ortamında YSA sınıflandırıcıları kullanılarak eğitilir. ÖM-2 ve ÖM-3'ün eğitimi sonrasında elde edilen sınıflandırıcılar sırasıyla YSA-1 ve YSA-2 şeklindedir.

- 5) Bölgelere ayrılmış test görüntüsü üzerindeki her bölge YSA-1 ve YSA-2 kullanılarak sınıflandırılır ve 2 farklı etiketli görüntü elde edilir.
- 6) Elde edilen etiketli görüntülerdeki her bir piksel, test görüntüsüne ilişkin gerçek sınıfların belirtildiği etiketli görüntünün ilgili pikselleri ile karşılaştırılarak modellerin sınıflandırma başarıları elde edilir.

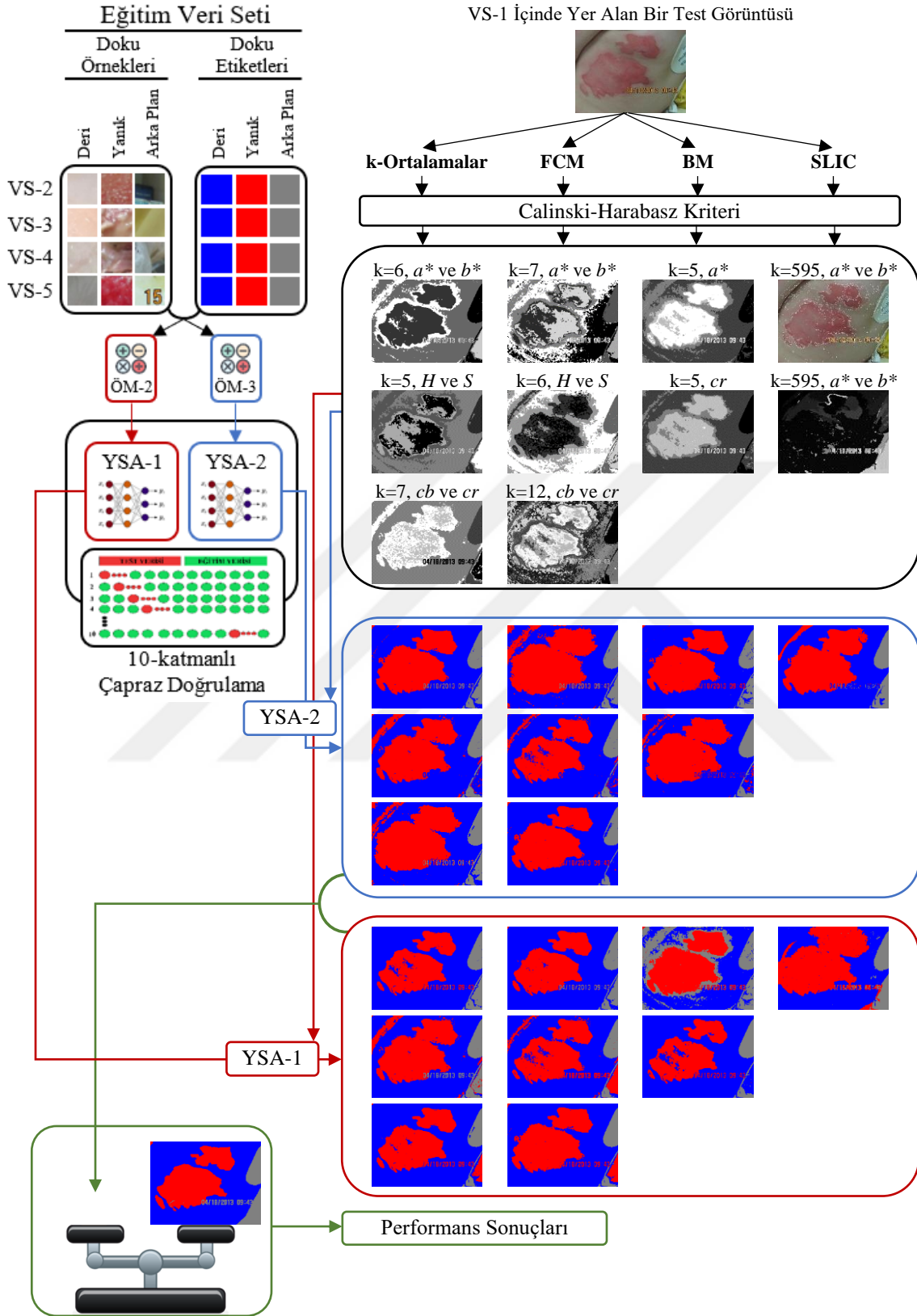
Şekil 27'de VS-1'de yer alan bir görüntünün sınıflandırma sürecinde izlenen adımlar şematik olarak gösterilmektedir. Sunulan tezde, bu adımlar alt veri setlerinde yer alan 21 görüntünün her biri için uygulanmıştır. Elde edilen performans değerlerinin ortalaması alınarak alt veri setine ilişkin ortalama başarımların değerleri bulunmuştur. Her bir alt veri setinin başarımlarının ortalaması alınarak sınıflandırıcıların genel başarımlarını sonuçları hesaplanmıştır. Tablo 22'de ÖM-2 ile eğitilmiş YSA modelinin (YSA-1) ve ÖM-3 ile eğitilmiş YSA modelinin (YSA-2), kullanılan segmentasyon yöntemine ve girdi verisinin türüne göre ortalama sınıflandırma performans sonuçları gösterilmektedir.

Tablo 22'deki sonuçlar incelendiğinde YSA-1 ve YSA-2'nin test görüntülerini sınıflandırmada çok yüksek başarımlar elde edemediği görülmektedir.

Tablo 22. YSA-1 ve YSA-2 sınıflandırıcılarının kullanılan segmentasyon yöntemine ve girdi verisinin türüne göre ortalama performans sonuçları

Metot	Renk Uzayı	k-Ortalamalar			BM		SLIC	FCM		
		CIE $L^*a^*b^*$	HSV	YCbCr	YCbCr	CIE $L^*a^*b^*$	CIE $L^*a^*b^*$	CIE $L^*a^*b^*$	HSV	YCbCr
	Renk Kanalı	a^* ve b^*	H ve S	cb ve cr	cr	a^*	a^* ve b^*	a^* ve b^*	H ve S	cb ve cr
YSA-1	O. H.	72.07	70.95	71.87	68.69	70.57	72.60	73.88	71.71	69.68
	O. D.	74.25	73.28	73.05	66.94	69.45	71.44	75.16	72.03	68.46
	O. F.	72.88	71.80	72.06	67.32	69.53	72.01	74.28	71.65	68.78
	O. K.	57.96	56.08	55.47	47.49	51.54	55.72	58.31	55.26	49.74
YSA-2	O. H.	68.36	70.34	66.98	65.78	65.09	74.86	65.99	68.33	62.41
	O. D.	70.54	70.13	69.87	67.39	68.69	73.26	69.88	69.34	67.63
	O. F.	68.89	69.66	67.82	66.12	66.41	73.79	67.29	68.02	64.36
	O. K.	59.37	56.74	58.12	55.65	56.61	60.99	58.73	55.19	56.62

YSA-1: ÖM-2 ile eğitilmiş model, YSA-2: ÖM-3 ile eğitilmiş model, O. H.: Ortalama Hassasiyet, O. D.: Ortalama Duyarlılık, O. F.: Ortalama F-skoru, O. K.: Ortalama Kappa. Tablodaki değerler yüzdelik (%) olarak verilmiştir.



Şekil 27. Segmentasyona dayalı sınıflandırma işleminin şematik gösterimi

YSA-1 modeli en yüksek sınıflandırma başarısına, girdi verisinin $CIE L^*a^*b^*$ renk uzayının a^* ve b^* kanalları, kümeleme yönteminin ise FCM olarak seçildiği durumda ulaşmıştır. Alt veri setlerinde yer alan toplam 105 görüntünün YSA-1 ile sınıflandırılmasına ilişkin ortalama F-skoru %74.28 olarak bulunmuştur. Öte yandan YSA-2 modelinin en yüksek sınıflandırma başarısını, girdi verisinin $CIE L^*a^*b^*$ renk uzayının a^* ve b^* kanalları olduğu ve kümeleme yöntemi olarak SLIC'in seçildiği durumda elde ettiği görülmektedir. YSA-2 modelinin test görüntülerini sınıflandırmadaki ortalama F-skoru %73.79'dur.

Performans sonuçlarının düşük olmasını etkileyen en önemli faktörün test görüntülerinin bölündüğü küme sayıları olduğu düşünülmektedir. Örneğin, test görüntüsünün bölüneceği optimum küme sayısının 3 olarak belirlendiği ve kümelerden birinin yanlış sınıflandırıldığı durumda, bu yanlış sınıflandırmanın performansa etkisi oldukça fazla olacaktır. Bu sorunun giderilebilmesi amacıyla Bölüm 3.6'da derin öğrenme yöntemlerinden biri olan semantik segmentasyon yaklaşımı kullanılmıştır.

3.6. Derin Öğrenme Yöntemlerinden Semantik Segmentasyon ile Test Görüntülerinin Sınıflandırılmasına İlişkin Performans Sonuçları

Bölüm 2.4.2'de belirtildiği gibi, semantik segmentasyon, görüntüdeki her bir pikseli, sınıflandırılacak kategorilerden biri ile ilişkilendiren derin öğrenme algoritmasıdır. Sınıflandırma işlemi piksel bazlı olduğundan eğitim için kullanılacak görüntülerin deri, yanık ve arka plan şeklinde ayrılma zorunluluğu yoktur. Eğitim görüntüsü üzerinde bu sınıfların üçü de bir arada bulunabilir. Her bir pikselin hangi sınıfa ait olduğunu gösteren etiket görüntüsü sayesinde semantik segmentasyon mimarileri eğitim işini piksel tabanlı olarak gerçekleştirebilmektedir.

Bu bölümde, test görüntülerini deri, yanık ve arka plan olarak sınıflandırmak amacıyla, çoklu sınıflar için piksel bazlı segmentasyon yapabilen kodlayıcı-kod çözücü (encoder-decoder) CNN mimarilerinden U-Net ve SegNet mimarileri kullanılmıştır. Veri setinde yer alan 105 görüntünün semantik segmentasyon yaklaşımı ile sınıflandırılması Bölüm 3.5'te tasarlanan 5-katmanlı çapraz doğrulama ortamında gerçekleştirilmiştir. Derin öğrenme modellerinin eğitim için çok fazla veriye ihtiyaç duyması nedeniyle veri setinde yer alan 105 görüntünün 32x32 ve 64x64 piksel büyüklüğünde bloklarından oluşan iki tür eğitim veri seti oluşturulmuştur. Bu sayede eğitim veri setinin sınıflandırma performansı üzerindeki etkisi araştırılmaya çalışılmıştır. VS-1'de yer alan bir test görüntüsünün semantik segmentasyon yöntemleri ile sınıflandırılmasına ilişkin adımlar şu şekildedir:

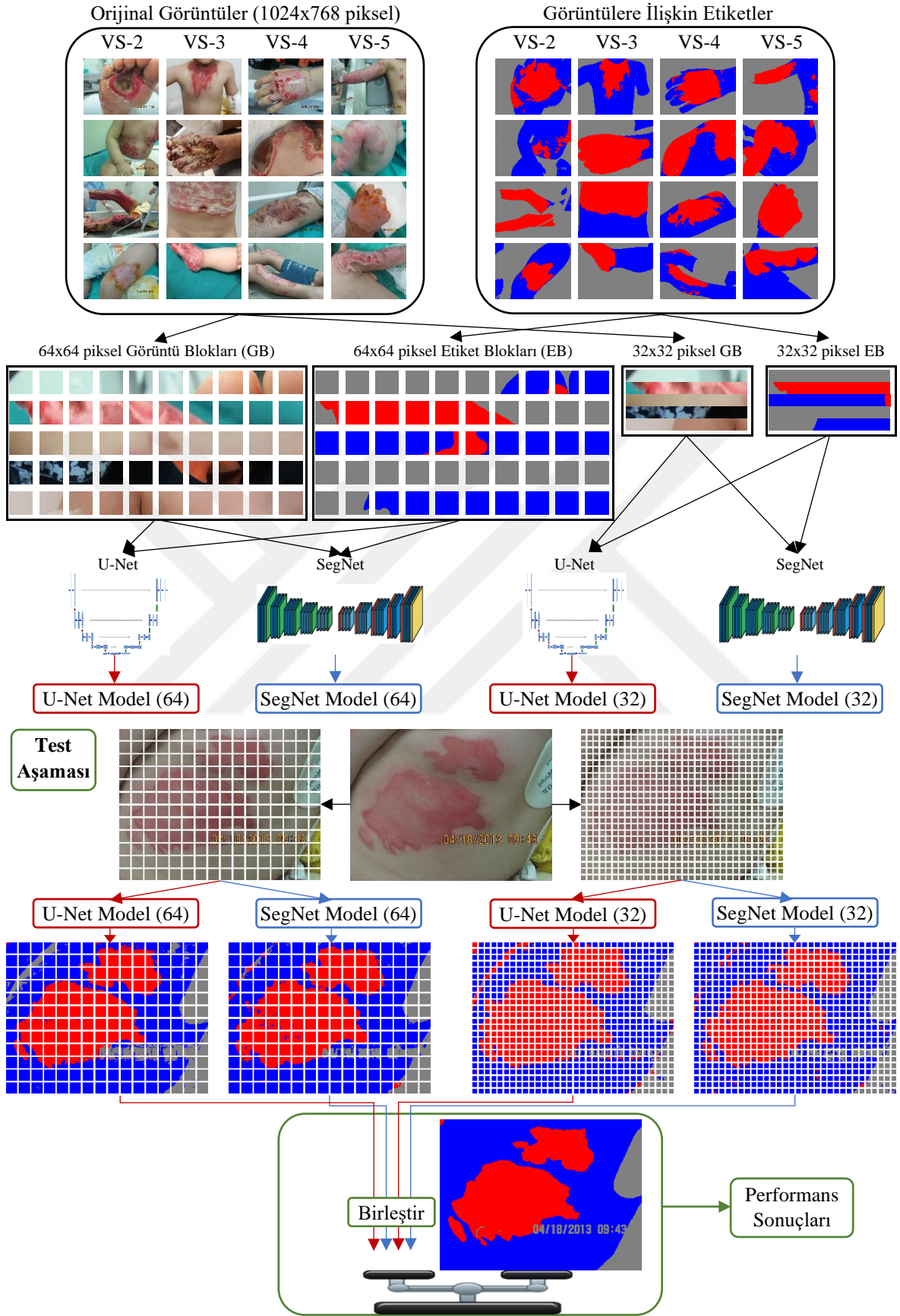
- 1) VS-2, VS-3, VS-4 ve VS-5'te yer alan tüm görüntüler ve bu görüntülere karşılık gelen etiket görüntüleri, 32x32 ve 64x64 piksel boyutunda bloklara bölünür.
- 2) İçerisinde sadece 32x32 piksel boyutundaki görüntülerin ve sadece 64x64 piksel boyutundaki görüntülerin yer aldığı iki ayrı eğitim veri seti oluşturulur.
- 3) Oluşturulan eğitim veri setleri SegNet ve U-Net mimarileri kullanılarak eğitilir ve iki ayrı model oluşturulur.
- 4) Sınıflandırılmak istenen 1024x768 piksel boyutundaki VS-1'de yer alan test görüntüsü, 32x32 ve 64x64 piksel boyutunda bloklara bölünür ve her bir blok görüntüsündeki pikseller, eğitim sonrasında elde edilen SegNet ve U-Net modelleri kullanılarak sınıflandırılır.
- 5) Sınıflandırılarak etiketli hale gelen 32x32 piksel boyutundaki ve 64x64 piksel boyutundaki blok görüntülerinin tekrardan birleştirilmesiyle, test görüntüsüne ilişkin 1024x768 boyutunda iki ayrı etiket görüntüsü ile edilir.
- 6) Elde edilen etiketli görüntülerdeki her bir piksel, test görüntüsüne ilişkin gerçek sınıfların belirtildiği etiketli görüntünün ilgili pikselleri ile karşılaştırılarak modellerin sınıflandırma başarıları elde edilir.

Şekil 28'de VS-1'de yer alan bir görüntünün semantik segmentasyon ile sınıflandırılması sürecinde izlenen adımlar şematik olarak gösterilmektedir. Sunulan tezde, bu adımlar alt veri setlerinde yer alan 21 görüntünün her biri için uygulanmıştır. Elde edilen performans değerlerinin ortalaması alınarak alt veri setine ilişkin ortalama başarımların değerleri bulunmuştur. Her bir alt veri setinin başarımlarının ortalaması alınarak sınıflandırıcıların genel başarımları hesaplanmıştır. Tablo 23'te 32x32 piksel ve 64x64 piksel boyutlarındaki görüntü blokları ile eğitilmiş SegNet ve U-Net mimarilerinden elde edilen ortalama sınıflandırma performans sonuçları gösterilmektedir.

Tablo 23. Semantik segmentasyon sonrası ortalama sınıflandırma performans sonuçları

	Eğitim Veri Seti			
	SegNet		U-Net	
	32x32 piksel boyutlu bloklar	64x64 piksel boyutlu bloklar	32x32 piksel boyutlu bloklar	64x64 piksel boyutlu bloklar
Ortalama Hassasiyet	78.90	79.98	78.11	77.4
Ortalama Duyarlılık	80.79	81.33	77.94	77.67
Ortalama F-skoru	79.67	80.50	77.83	77.32
Ortalama Kappa	71.71	72.64	67.65	66.17

Tablodaki değerler yüzdeler (%) olarak verilmiştir.



Şekil 28. Semantik segmentasyon işleminin şematik gösterimi

Tablo 23 incelendiğinde, SegNet mimarisi kullanılarak elde edilen sınıflandırma sonuçlarının, U-Net mimarisine oranla az da olsa daha iyi olduğu görülmektedir. SegNet mimarisi 64x64 piksel boyutundaki bloklar ile eğitildiğinde, U-Net mimarisi ise 32x32 piksel boyutlu bloklar ile eğitildiğinde daha başarılı sonuçlar vermiştir. Veri setindeki 105 görüntünün 5-katmanlı çapraz doğrulama ortamında sınıflandırılmasından sonra elde edilen en yüksek F-skoru Segnet mimarisi tarafından %80.5 olarak bulunmuştur. Modele ilişkin ortalama kappa istatistiği 0.7264 olarak hesaplanmıştır. 1'e kısmen yakın olan bu değer sınıflandırma performansının yüksek olduğunu göstermektedir.

Çok değişkenli istatistiksel yöntemlerin kullanıldığı yaklaşımda en yüksek başarıma sahip sınıflandırıcıların performans sonuçları ile SegNet ve U-Net mimarileri kullanılarak elde edilen performans sonuçları Tablo 24'te karşılaştırılmıştır.

Tablo 24. Semantik segmentasyon ve çok değişkenli istatistiksel yöntemler ile sınıflandırmanın performanslarının karşılaştırılması

Test Edilen Veri Seti		Yöntem			
		Semantik Seg. SegNet (64x64 piksel boyutundaki blok görüntüler ile eğitilmiş)	Semantik Seg. U-Net (32x32 piksel boyutundaki blok görüntüler ile eğitilmiş)	Seg.: FCM (Girdi: $CIE L^*a^*b^*$ - a^* ve b^* kanalları) Sınıflandırma: YSA Ö. M.: ÖM-2	Seg.: SLIC (Girdi: $CIE L^*a^*b^*$ - a^* ve b^* kanalları) Sınıflandırma: YSA Ö. M.: ÖM-3
VS-1	O. H.	82.67	82.52	71.81	77.19
	O. D.	83.81	82.95	74.52	77.52
	O. F.	83.10	82.48	73.05	77.43
	O. K.	78.56	77.68	53.39	70.69
VS-2	O. H.	79.29	75.14	77.38	73.90
	O. D.	82.00	74.19	78.19	69.81
	O. F.	80.48	74.43	77.76	71.48
	O. K.	74.67	63.42	60.07	55.95
VS-3	O. H.	72.86	72.14	70.57	62.14
	O. D.	71.57	69.33	74.62	57.52
	O. F.	72.14	70.48	72.29	59.14
	O. K.	59.89	56.04	64.94	40.83
VS-4	O. H.	85.10	82.67	76.57	83.05
	O. D.	85.57	83.14	74.76	81.19
	O. F.	85.19	82.76	75.52	81.95
	O. K.	75.92	71.65	57.91	70.83
VS-5	O. H.	80.00	78.10	73.05	78.00
	O. D.	83.71	80.10	73.71	80.24
	O. F.	81.62	79.00	72.76	78.95
	O. K.	74.17	69.47	55.27	66.66
	O. H.	79.98	78.11	73.88	74.86
	O. D.	81.33	77.94	75.16	73.26
	O. F.	80.50	77.83	74.28	73.79
	O. K.	72.64	67.65	58.31	60.99

O. H.: Ortalama Hassasiyet, O. D.: Ortalama Duyarlılık, O. F.: Ortalama F-skoru, O. K.: Ortalama Kappa, Seg.: Segmentasyon, Ö. M.: Öznitelik Matrisi, VS: Veri Seti. Tablodaki değerler yüzdelik (%) olarak verilmiştir.

Orijinal ve Etiketli Görüntüler			Kullanılan Yönteme Göre Sınıflandırma Sonuçları							
Orijinal Görüntü	Gerçek Görüntü Etiketi	Gerçek Görüntü Etiketi	Semantik Seg. SegNet (Eğitim: 64x64 piksellik görüntüler)		Seg.: FCM (Girdi: - a^* ve b^*) Sınıflandırma: YSA Ö. M.: ÖM-2		Seg.: SLIC (Girdi: a^* ve b^*) Sınıflandırma: YSA Ö. M.: ÖM-3			
			Gerçek Görüntü Etiketi	Gerçek Görüntü Etiketi	Gerçek Görüntü Etiketi	Gerçek Görüntü Etiketi	Gerçek Görüntü Etiketi	Gerçek Görüntü Etiketi		
1	18									
2	19									
3	20									
4	21									
5	22									
6	23									
7	24									
8	25									
9	26									
10	27									
11	28									
12	29									
13	30									
14	31									
15	32									
16	33									
17	34									

Mavi: Deri, Kırmızı: Yanık, Gri: Arka Plan, Ö. M.: Öznitelik Matrisi, Seg.: Segmentasyon

Şekil 29. Sınıflandırılmış görüntülerin yöntem bazında karşılaştırılması

Tablo 24 incelendiğinde, semantik segmentasyon ile elde edilen performans sonuçlarının çok değişkenli istatistiksel yöntemlerin kullanımı ile edilen sonuçlardan daha iyi olduğu açıkça görülmektedir. SegNet mimarisi kullanılarak eğitilen model tüm alt veri setlerinde diğer modellerden daha iyi başarımlar elde etmiştir. Dikkat çeken bir başka nokta ise yöntemlerin alt veri setlerindeki görüntüleri sınıflandırmadaki başarımlarının benzerlik göstermesidir. VS-3'te yer alan görüntüler tüm modeller tarafından en düşük başarıyla sınıflandırılırken, VS-4'te yer alan görüntüler ise tüm modeller tarafından en yüksek başarıyla sınıflandırılmıştır. Rastgele olarak oluşturulan bu alt veri setleri incelendiğinde, VS-3'te daha çok 3. ve 4. derece yanık görüntülerinin olduğu, görüntülerin alındığı ortamdaki arka planın da karmaşık bir yapıda bulunduğu tespit edilmiştir. Diğer taraftan VS-4'te ise düşük dereceli yanık görüntülerinin yer aldığı ve görüntü arka planlarının sade olduğu görülmüştür. Şekil 29'da, Tablo 24'te performans karşılaştırılması yapılan yöntemlerin 34 adet test görüntüsü üzerindeki sınıflandırma başarımları gösterilmektedir.

3.7. Elde Edilen Sonuçların Literatürdeki Benzer Çalışmalarla Karşılaştırılması

Bilgisayar destekli sistemler tıbbi görüntü analizlerinde yaygın olarak kullanılmaktadır. Bununla birlikte, birçok durumda, bu sistemlerin performansı takip edilen protokollerin kalitesine göre değişmektedir. Sunulan tezde kullanılan görüntüler dijital kameralar ve cep telefonları kullanılarak çekilmiş ve çekim sırasında herhangi bir protokol izlenmemiştir. Bu nedenle, görüntüler; parlak ve karanlık bölgeler, gölgeler, karmaşık bir arka plan veya yanık bölge üzerinde bandaj kalıntıları içerebilmektedir. Bu gibi durumlarda renkli görüntülerin analizi her zaman zordur. Örneğin, hastanın ten rengiyle aynı renge sahip bir duvarın önünde çekilen bir görüntüde, kullanılan segmentasyon yönteminin duvarı ve hastanın cildini aynı küme içine alması olası bir durumdur. Dolayısıyla yanlış segmentasyon sonrasında sınıflandırma da hatalı olacak ve sistemin başarısı düşecektir. Şekil 30'da, yanlış sınıflandırmaya neden olabilecek birkaç doku örneği gösterilmektedir. Segmentasyon sürecinde meydana gelecek bu tür sorunların üstesinden gelmek için en iyi çözüm, görüntülerin ayırt edilebilir arka plana sahip bir ortamda, uygun aydınlatma altında belirli bir mesafeden çekildiği bir görüntü alma protokolü oluşturmaktır.

Yanık görüntü sınıflandırması popüler bir araştırma alanı olmakla birlikte, yanık cildin sağlıklı ciltten ayırt edilmesine yönelik çalışmalar sınırlıdır [54; 58; 60; 67]. Bölüm 1.9'da ele alınan çalışmaların çoğu, esas olarak yanık derinliğinin sınıflandırılmasına

odaklanmıştır. Sunulan tezde ana amaç, renkli görüntülerde yanık değerlendirilmesinin ön basamağı olan yanık deri bölgelerini sağlıklı deri bölgelerinden ayırmak için yeni bir sistem önermektir. Elde edilen sonuçların daha sonra yanık yüzey alanının tespiti için kullanılması planlanmaktadır. Ayrıca tezde önerilen yöntemler, yanık derinliğini sınıflandırmak için de kullanılabilme potansiyeline sahiptir.



Şekil 30. Yanık sınıflandırmaya neden olabilecek doku örnekleri. Sütunlarda sırasıyla deri, yanık ve arka plan dokuları yer almaktadır.

Yanık deri bölgelerini sağlıklı deri bölgelerinden ayırt etme amacıyla literatürde kullanılan çalışmalar temel olarak dört dezavantaja sahiptir:

- 1) Acha ve diğerleri [43], aydınlatma ve kamera kalibrasyonu ile ilgili problemlerin üstesinden gelmek için özel bir görüntü alma protokolü geliştirmiştir. Aynı protokol sonraki yıllarda yapılan birçok çalışmada kullanılmıştır [43–49]. Bu çalışmalarda önerilen yöntemin performansı, görüntü alma protokolünün doğru takip edilmesine bağlı olmakla beraber kurallara uymayan bir teknisyen tarafından çekilen görüntü için hatalı sonuçlar elde edilme riski bulunmaktadır.
- 2) Literatürde yer alan bazı çalışmalarda [50; 53] kullanılan veri seti düşük kalitede görüntüler içermekle beraber bu görüntülerin bir kısmı web sayfalarından toplanmış, bir kısmı kitaplardan taranmış, bazıları ise hastanelerden edinilmiştir.
- 3) Literatürde yer alan bazı çalışmalarda [43–46] $CIE L^*u^*v^*$ renk uzayı temel alınarak manuel segmentasyon uygulanmıştır. Bu çalışmalarda segmentasyon, yanık bölgesindeki küçük bir bölgenin bilgisayar faresi kullanılarak seçilmesi ile başlamış ve görüntünün ön işlemleriyle devam etmiştir. Görüntü tek bir kanala dönüştürüldükten sonra eşikleme ve son işleme ile segmentasyon işlemi tamamlanmıştır. Veri setinde yer alan görüntülerdeki tüm yanık bölgelerin

bilgisayar faresi yardımıyla segmente edilmesi çok fazla çaba ve deneyim gerektiren bir işlemdir.

- 4) Literatürde yer alan hemen hemen tüm çalışmalarda sınıflandırma ve segmentasyon yöntemleri, çıkartılan öznitelikler ve kullanılacak renk uzayları önceden belirlenmiş, bu kriterler arasındaki etkileşim araştırılmamıştır.

Sunulan tezde önerilen sistem bu dezavantajları ortadan kaldırmayı amaçlamıştır. Literatürde yer alan çalışmalarda elde edilen performans değerlerinin tezde önerilen sistemden elde edilen sonuçlar ile karşılaştırılması Tablo 25'te yapılmıştır. Tablo

Tablo 25. Önerilen yöntemlere ilişkin sınıflandırma performansının literatürdeki yöntemlerde elde edilen performanslarla karşılaştırılması

Çalışma	Sınıflandırılan Bölgeler	Yöntem	Performans
Betanzos ve diğerleri [60]	Yanık, Deri, Arka Plan	Bulanık kNN	Yaklaşık doğruluk: %83
		FCM	Yaklaşık doğruluk: %80
Wantanajittikul ve diğerleri [67]	Yanık ve Deri	FCM	Pozitif kestirim değeri: %92 Duyarlılık: %85
Badea ve diğerleri [54]	Yanık ve Deri	Derin Öğrenmeli RGB-LeNet	Hassasiyet: %75.91
		Derin Öğrenmeli Ölçeklenmiş RGB	Hassasiyet: %72.17
		Renk Özü-Doygunluk	Hassasiyet: %53.87
Despo ve diğerleri [58]	Yanık ve Deri	CRF kullanılmayan FCN	Piksel doğruluğu: %82 Birleşim üzerinde ortalama kesişim: %54
		FCN-CRF	Piksel doğruluğu: %85 Birleşim üzerinde ortalama kesişim: %56
		FCN-CRF ve Veri Artırma	Piksel doğruluğu: %85 Birleşim üzerinde ortalama kesişim: %67
Sunulan tezde önerilen yaklaşımlar	Yanık, Deri, Arka Plan	Bulanık c-Ortalamalar, ÖM-2 ile eğitilmiş YSA	Hassasiyet: %73.88, Duyarlılık: %75.16, F-skoru: %74.28, Kappa: 0.5831
		SLIC, ÖM-3 ile eğitilmiş YSA	Hassasiyet: %74.86, Duyarlılık: %73.26, F-skoru: %73.79, Kappa: 0.6099
		U-Net mimarisi ile semantik segmentasyon	Hassasiyet: %78.11, Duyarlılık: %77.94, F-skoru: %77.83, Kappa: 0.6765
		SegNet mimarisi ile semantik segmentasyon	Hassasiyet: %79.98, Duyarlılık: %81.33, F-skoru: %80.50, Kappa: 0.7264

incelendiğinde, tezde önerilen yöntemlerin daha yüksek bir başarı oranına sahip olduğu görülse de en adil performans karşılaştırmasının ele alınan tüm çalışmaların aynı veri tabanını ve aynı performans metriği kullanması durumunda elde edileceği düşünülmektedir.

Sunulan tezde Bölüm 2.3'te önerilen yöntemlerin dezavantajlarından biri, performansın büyük ölçüde segmentasyon sonuçlarına bağlı olmasıdır. Segmente edilmiş bölge hem sağlıklı hem de yanık deriyi bir arada içeriyorsa, bu bölgeden çıkartılan öznelilikler sınıflandırıcıların hatalı şekilde eğitilmesine neden olabilmektedir. Çoğu segmentasyon yöntemi, küme sayısı doğru bir şekilde belirtildiğinde görüntü üzerindeki pikselleri doğru bir şekilde kümeleyebilmektedir, fakat optimum küme sayısının seçilmesi zor bir süreçtir. Sunulan tezde optimum küme sayısını tahmin edebilmek için, Calinski-Harabasz kriteri farklı girdi verileri ve farklı segmentasyon yöntemleri kullanılarak uygulanmıştır. Bununla birlikte, aynı amaç için başka bir kriterin kullanılması durumunda farklı sonuçlar elde etmek mümkündür. Stafilokokal haşlanmış deri sendromu, herpes zoster (zona), büllöz impetigo ve erizipel gibi enfeksiyonların ve hastalıkların, cildin yanıkmiş gibi görünmesine neden olabileceği bilinmektedir. Tezde kullanılan sınıflandırma modelleri sadece sağlıklı ve yanık deri bölgelerine ilişkin görüntüler kullanılarak eğitildiği için, bu modellerin hastada bulunan yanık benzeri bir enfeksiyonu yanık olarak sınıflandırması olası bir durumdur. Tezde önerilen sistem kullanılmadan önce hastaya bu tür bir özel durumunun olup olmadığı sorulmadır. Bununla birlikte, bu nadir durum, sınıflandırma sistemini eğitecek enfeksiyon görüntülerinin bulunmamasından dolayı sunulan tezin kapsamı dışındadır. Deride yanık benzeri izler ile sonuçlanan herpes zoster enfeksiyonuna (zona hastalığına) ilişkin bir görüntü örneği Şekil 31'de sunulmuştur.

Sunulan tezde kullanılan görüntülerin herhangi bir protokol izlenmeden çekilmiş olduğu dikkate alındığında, sınıflandırma sonuçlarının tatmin edici olduğuna inanılmaktadır.



Şekil 31. Herpes zoster enfeksiyonu için örnek bir görüntü [196]. Copyright 2019, StatPearls Publishing

4. SONUÇLAR

Günümüzde yanık ünitelerinde ve yanık merkezlerinde karşılaşılan en yaygın sorunlardan biri yanık hastasına uygulanacak sıvı resüsitasyonuna hatalı olarak karar verilmesidir. Bu kararda rol oynayan en önemli iki etken hastanın yanık yüzdesi ve yanıkların derecesidir. Yanık yüzdesi hesaplanırken kullanılan en yaygın yöntem olan dokuzlar kuralında verilen kararlar yanık uzmanları arasında farklılık gösterebilmekte, dolayısıyla hastaya uygulanması düşünülen sıvı resüsitasyonunun dozajı da değişebilmektedir. Bu soruna bir çözüm olarak, üç boyutlu kameralardan oluşan bir görüntü alma ortamında, hastanın taranması ile hastaya özel 3D modelin oluşturulması ve bu model üzerinde yanık tespiti yapılarak hastanın yanık yüzdesinin gerçeğe en yakın şekilde hesaplanması planlanmaktadır. Sunulan tezde, yanık hastalarına ilişkin dijital (2D) görüntülerdeki sağlıklı ve yanık deriye ilişkin bölgelerin birbirinden ayırt edilebilmesi için, çok değişkenli istatistiksel yöntemler ve semantik segmentasyon derin öğrenme algoritmaları kullanılmış ve bu sayede oluşturulması planlanan 3D yanık analiz sisteminde uygulanacak işlemler için ön hazırlık yapılmıştır.

Çalışmanın sonucu, üç ana süreç için ön hazırlık olmuştur: Yanıkların sınıflandırılması, yanık derecesinin belirlenmesi ve periyodik olarak hastalardan alınan dijital fotoğraflara dayanarak yanık iyileşme sürecinin seyri hakkında bilgi edinilmesi.

Sunulan tez çalışmasında, veri setinde yer alan 105 adet görüntüdeki sağlıklı ve yanık deri bölgelerin ayırt edilebilmesi için iki tip yaklaşım önerilmiştir:

İlk yaklaşım, segmentasyon ve sınıflandırma olmak üzere iki aşamadan oluşmaktadır. Segmentasyon aşamasında test edilen her bir görüntü dört farklı segmentasyon yöntemi kullanılarak kümelere ayrılmıştır. Görüntülerin kaç farklı kümeye ayrılması gerektiğine Calinski-Harabasz kriteri kullanılarak karar verilmiştir. Bu kriter uygulanırken *RGB* görüntüsü, *CIE L*a*b**, *HSV* ve *YCbCr* renk uzaylarına dönüştürülmüş ve bu renk uzaylarının *a**, *b**, *H*, *S*, *Cb* ve *Cr* kanallarına ait piksel değerleri kriter tarafından optimum küme sayısının bulunabilmesi için kullanılmıştır. Kriterden elde edilen sonuçlar doğrultusunda her bir görüntü bölgelere ayrılmış ve segmentasyon aşaması tamamlanmıştır. Sınıflandırma aşamasında, segmente edilen bölgeleri sınıflandırabilecek en iyi model bulunmaya çalışılmıştır. Bu amaçla, literatürde yaygın olarak kullanılan 10 sınıflandırıcının eğitilebilmesi için veri setinde yer alan 105 görüntü, 5 alt veri setine bölünmüş ve her bir alt

veri setinden 450'şer adet 64x64 piksel boyutlu deri, yanık ve arka plan örneği rastgele olarak seçilmiştir. Bu örnekler üzerinden ÖM-1, ÖM-2 ve ÖM-3 adında üç adet öznitelik matrisi hesaplanmıştır (Bu matrisler dokuya ilişkin sınıf bilgisinin tutulduğu bir etiket değeri de içermektedir). ÖM-1, gri seviye histogram istatistikleri kullanılarak çıkartılan 49 adet öznitelik ile GLCM kullanılarak elde edilen 364 Haralick doku özneliğinin bir araya getirilmesi ile oluşan toplamda 413 özneliği; ÖM-2, ÖM-1'e ileriye doğru seçim uygulandıktan sonra elde edilen 9 adet özneliği; ÖM-3 ise renk kanallarının ortalama parlaklık değerlerinin kullanılması ile elde edilen 13 özneliği içermektedir. Doku örneklerini sınıflandırmada en iyi performansların, öznitelik matrislerinin her birinin geri yayılım algoritmasının uygulandığı çok katmanlı ileri beslemeli YSA modelinin eğitilmesi için kullanıldığında elde edildiği görülmüştür. 10-katmanlı çapraz doğrulama ortamında eğitilen üç YSA modeli de sınıflandırma sonucunda %88'in üzerinde makro ve mikro-ortalama F-skoru elde etmiştir. Yeterince yüksek sınıflandırma performansına ve sınıflandırma işleminin hızlanmasını sağlayacak az sayıda özneliğe sahip olmalarından dolayı test görüntülerini sınıflandırmada ÖM-2 ve ÖM-3 kullanılmıştır. Bu matrisler ile eğitilen YSA sınıflandırıcılarının veri setinde yer alan 105 görüntüyü sınıflandırma performanslarını belirleyebilmek için 5-katmanlı bir çapraz doğrulama ortamı tasarlanmıştır. Herhangi bir alt veri setindeki 21 görüntü sınıflandırılırken, diğer alt veri setlerindeki 84 görüntüden çıkartılmış 64x64 piksellik doku örnekleri eğitim veri seti olarak kullanılmıştır. Bu işlem 5 alt veri setinin hepsi için uygulanmıştır. Dolayısıyla 5'i ÖM-2, 5'i ÖM-3 ile eğitilen 10 adet YSA modeli, 4 farklı segmentasyon yönteminin, farklı renk kanallarını kullanarak segmente ettiği test görüntülerindeki bölgeleri sınıflandırmıştır. Elde edilen performans değerlerinin ortalaması alınarak en iyi sınıflandırıcı tespit edilmiştir. Görüntüleri bölgelere ayırırken $CIE L^*a^*b^*$ renk uzayının a^* ve b^* renk kanallarının FCM'de girdi verisi olarak kullanıldığı ve segmente edilmiş görüntüdeki her bir bölgenin ÖM-2 ile eğitilmiş YSA modeli ile sınıflandırıldığı durumda %74.28'lik ortalama F-skoru ile en iyi sınıflandırma performansının elde edildiği görülmüştür. İkinci sıradaki en yüksek sınıflandırma performansına, görüntüleri bölgelere ayırırken a^* ve b^* renk kanallarının SLIC algoritmasında kullanıldığı ve segmente edilmiş görüntüdeki her bir bölgenin ÖM-3 ile eğitilmiş YSA modeli ile sınıflandırıldığı durumda ulaşıldığı tespit edilmiştir. Bu durumda sınıflandırmaya ilişkin hata tablosunda ortalama F-skoru %73.79 olarak bulunmuştur.

Sunulan tez çalışmasında, görüntülerdeki sağlıklı ve yanık deri bölgelerinin ayırt edilebilmesi için kullanılan ikinci yaklaşımda derin öğrenme yöntemlerinden biri olan semantik segmentasyon kullanılmıştır. Segmentasyona dayalı sınıflandırma işleminde oluşturulan 5-katmanlı çapraz doğrulama ortamı bu yaklaşım için de kullanılmıştır. Veri setinde yer alan 105 görüntünün 32x32 ve 64x64 piksel büyüklüğünde bloklarından oluşan iki tür eğitim veri seti oluşturulmuştur. Herhangi bir alt veri setindeki 21 görüntü sınıflandırılırken, diğer alt veri setlerindeki 84 görüntüden çıkartılmış 32x32 ve 64x64 piksellik görüntü blokları ve bu bloklara ilişkin etiketli bloklar eğitim veri seti olarak kullanılmıştır. Bu işlem 5 alt veri setinin hepsi için uygulanmıştır. Sınıflandırma aşamasında çoklu sınıflar için piksel bazlı segmentasyon yapabilen kodlayıcı-kod çözücü CNN mimarilerinden U-Net ve SegNet mimarileri kullanılmıştır. Dolayısıyla her bir alt veri setindeki görüntüler test edilirken iki çeşit eğitim seti (32x32 ve 64x64 piksel boyutundaki görüntü blokları) ve iki çeşit semantik segmentasyon yöntemi (SegNet ve U-Net) kullanıldığı için toplamda 4 çeşit model oluşturulmuştur. Yani, 5 alt veri setinin tamamı sınıflandırılırken toplamda 20 model kullanılmıştır. Her bir alt veri setinden elde edilen performans değerlerinin ortalaması alınarak yönteme ilişkin ortalama başarımlar elde edilmiştir. SegNet mimarisi 64x64 piksel boyutundaki bloklar ile eğitildiğinde, U-Net mimarisi ise 32x32 piksel boyutlu bloklar ile eğitildiğinde daha başarılı sonuçlar vermiştir. SegNet mimarisinde veri setinde yer alan 105 görüntünün 5-katmanlı çapraz doğrulama ortamında sınıflandırılmasından sonra elde edilen en yüksek ortalama F-skoru %80.5 iken, U-Net mimarisinde bu değer %77.83 olarak bulunmuştur.

Sunulan tezde, herhangi bir protokol kullanılmadan çekilen 105 görüntünün otomatik analizi sonrasında elde edilen ortalama F-skoru, dijital görüntülerindeki sağlıklı ve yanık deri bölgelerinin sınıflandırılması alanındaki mevcut yöntemlerin çoğunda elde edilen performans değerlerinden daha yüksektir.

5. ÖNERİLER

Sunulan tezde, dijital yanık görüntülerinde bulunan sağlıklı deriye, yanık deriye ve arka plana ilişkin bölgeler iki tip yaklaşım ile sınıflandırılmıştır. Bu yaklaşımlardan ilki olan çok değişkenli istatistiksel yöntemler ile sınıflandırmada başarımların oranı; görüntünün bölüneceği küme sayısı, kullanılan segmentasyon yönteminin, seçilen özniteliklerin ve bu özniteliklere göre eğitilen sınıflandırıcıların performansı ile doğru orantılıdır. İkinci yaklaşım olan semantik segmentasyondaki başarımlar ise eğitim için kullanılacak olan görüntü sayısı ile doğru orantılı olmakla beraber diğer yaklaşıma göre çok daha az kriterden etkilenmektedir. Dolayısıyla yanık görüntülerinde yanık derecesinin aynı olduğu bölgelerin sınıflandırılmasına yönelik bir çalışma yapılması durumunda tercih edilmesi gereken ilk yöntem semantik segmentasyon olmalıdır.

Yanık görüntülerinin literatürde en yaygın olarak kullanılan segmentasyon yöntemleri ile bölgelere bölünmesinde *CIE La*b** renk uzayının *a** ve *b** kanallarına ilişkin verilerin kullanılması durumunda daha iyi kümeleme sonuçları alınmıştır. Bu nedenle yanık üzerine yapılan çalışmalarda bu renk uzayının kullanılması tercih edilebilir.

Sunulan tezde kullanılan görüntülerin birçoğunda, fotoğraf çekimi sırasında sadece yanığa veya deriye ilişkin bölgeler dikkate alınmıştır. Dolayısıyla görüntüde yer alan insanların veya uzuvların tespit edilmesi imkânsızdır. Görüntü çekimi sırasında bu noktaya dikkat edilmesi durumunda sınıflandırıcıların eğitimi sırasında uzuv dışında kalan bölgelerin deri veya yanık olarak algılanmaması sağlanabilir. Hatta, görüntülerin ayırt edilebilir arka plana sahip bir ortamda, uygun aydınlatma altında belirli bir mesafeden çekildiği bir görüntü alma protokolü de oluşturulabilir.

Elde edilen sonuçlar doğrultusunda, yanık bölgeye ilişkin rengin genelde kırmızı olmakla beraber değişiklik gösterebileceği görülmüştür. Ayrıca yanığın belirli bir şeklinin ve yüksek ayırt ediciliğe sahip belirgin bir dokusunun da olmadığı saptanmıştır. Diğer taraftan, deriye ilişkin bölgeleri segmente etmedeki başarımın, özellikle arka plan etkisinin ortadan kaldırıldığı durumlarda oldukça yüksek olduğu tespit edilmiştir. Sunulan tezin bir sonraki aşamasında yanık hastalarının stereo kameralarla taranarak üç boyutlu modellerinin oluşturulması ve bu model üzerinden vücut yanık alan yüzdesinin hesaplanması planlanmaktadır. Tezden elde edilen bilgiler ışığında, üç boyutlu modelde yanık alanları bulmak yerine, deriye ilişkin bölgelerin bulunup, tüm vücut yüzey alanından deriye ait

alanların ıkartılmasıyla yanık yzey alanının hesaplanmasının yksek bařarımla sonulanacağı dřnlmektedir.

Dijital yanık grntleri zerinde yapılması planlanan alıřmalarda, kullanılacak olan yntemlere iliřkin performans sonularının adil bir řekilde karřılařtırılabilmesi iin ortak bir veri tabanı oluřturulmalıdır. Bu amala ierisinde yanık grntlerinin ve grntlere iliřkin etiketlemelerin yer alacağı geniř bir yanık veri tabanının oluřturulması planlanmaktadır.



6. KAYNAKLAR

1. World Health Organization, WHO: Burns-Key Facts. <https://www.who.int/news-room/fact-sheets/detail/burns> 6 Mart 2018.
2. T.C. Sağlık Bakanlığı, 2018 Yılı Faaliyet Raporu. <https://sgb.saglik.gov.tr/TR,59134/faaliyet-raporu-2018.html> Şubat 2019.
3. Erdemir, N., Hypericum Perforatum Yağı Yüklenmiş Polimerik Membranların Yanık Yara İyileşmesinde Skar Doku Oluşumuna Etkisinin İncelenmesi, Yüksek Lisans Tezi, Tokat Gaziosmanpaşa Üniversitesi, Fen Bilimleri Enstitüsü, Tokat, 2019.
4. Turkington, C. ve Dover, J., S., The Encyclopedia of Skin and Skin Disorders, Third Edition, Infobase Publishing, New York, 2006.
5. Burr, S. ve Penzer, R., Promoting Skin Health, Nursing Standard, 19, 36 (2005) 57-65.
6. Penzer, R. ve Ersser, S., Principles of Skin Care: A Guide for Nurses and Health Care Practitioners, First Edition, John Wiley & Sons, New Jersey, 2010.
7. Champion, R., H., Burrington, J., L., Ebling, F., J., G., Textbook of Dermatology, Fifth Edition, Blackwell Scientific Publications, New York, 1992.
8. Vural, M., K., Yanıklı Hastalarda Yüzeysel Sürüntü ve Kantitatif Doku Biyopsi Kültürü Örneklerinin Karşılaştırılması, Uzmanlık Tezi, Atatürk Üniversitesi, Erzurum, 2009.
9. Özkal, Ö., Alt Ekstremitte Yanık Yaralanması Olan Bireylerde Akut Dönemde Uygulanan Farklı Yürüme Eğitimlerinin Fiziksel Fonksiyonlara Etkisinin İncelenmesi, Doktora Tezi, Hacettepe Üniversitesi, Sağlık Bilimleri Enstitüsü, Ankara, 2018.
10. Richards, A. ve Dafydd, H., Key Notes on Plastic Surgery, Second Edition, Wiley-Blackwell, New Jersey, 2014.
11. Hettiaratchy, S. ve Dziewulski, P., Pathophysiology and Types of Burns, British Medical Journal, 328, 7453 (2004) 1427-1429.
12. Cydulka, R., K., Cline, D., M. ve Ma, O., J., Tintinalli's Emergency Medicine Manual, Eighth Edition, McGraw-Hill Education, New York, 2017.
13. Marx, J. A., Hockberger, R., S. ve Walls, R., M., Rosen's Emergency Medicine: Concepts and Clinical Practice, Seventh Edition, Mosby Elsevier, Philadelphia, 2010.
14. Park, J., W. ve Jung, D., I., Integumentary Physical Therapy, First Edition, Springer-Verlag Berlin Heidelberg, Berlin, 2016.

15. Woodson, L., C., Diagnosis and Grading of Inhalation Injury, Journal of Burn Care and Research, 30, 1 (2009) 143-145.
16. Araz, C. ve Pirat, A., İnhalasyon Hasarı, Türk Yoğun Bakım Derneği Dergisi, 9, 1 (2011) 37-45.
17. Koltka, K., Yanık Yaralanmaları: Yanık Derinliği, Fizyopatolojisi ve Yanık Çeşitleri, Türk Yoğun Bakım Derneği Dergisi, 9, 1 (2011) 1-6.
18. Prahlow, J., A., Forensic Pathology for Police, Death Investigators, Attorneys, and Forensic Scientists, First Edition, Humana Press, New Jersey, 2010.
19. Monstrey, S., Hoeksema, H., Verbelen, J., Pirayesh, A. ve Blondeel, P., Assessment of Burn Depth and Burn Wound Healing Potential, Burns, 34, 6 (2008) 761-769.
20. Kahramanlar, D., Çocuk Yanık Yoğun Bakım Ünitesinde İzlenen Hastalarda Akut Böbrek Hasarı, Uzmanlık Tezi, Sağlık Bilimleri Üniversitesi, Ankara, 2019.
21. Jaskille, A., D., Shupp, J., W., Jordan, M., H. ve Jeng, J., C., Critical Review of Burn Depth Assessment Techniques: Part I. Historical Review, Journal of Burn Care & Research, 30, 6 (2009) 937-947.
22. Jeschke, M., G., Kamolz, L., P., Sjöberg, F. ve Wolf, S., E., Handbook of Burns Volume 1: Acute Burn Care, First Edition, Springer, New York, 2012.
23. Kartal Erdost, Ş. ve Çetinkale, O., İstanbul Üniversitesi Cerrahpaşa Tıp Fakültesi Sürekli Tıp Eğitimi Etkinlikleri, Yara Bakımı ve Tedavisi, Sempozyum Dizisi 67, (2008) 107-117.
24. Buz, A., Parsiyel Kalınlıkta Termal Yanıkların Tedavisinde Mezoterapinin Etkinliği-Deneysel Çalışma, Uzmanlık Tezi, Bülent Ecevit Üniversitesi, Zonguldak, 2012.
25. Kagan, R., J., Peck, M., D., Ahrenholz, D., H., Hickerson, W., L., Holmes IV, J., H., Korentager, R., Kraatz, J., Pollock, K. ve Kotoski, G., Surgical Management of the Burn Wound and Use of Skin Substitutes: An Expert Panel White Paper, Journal of Burn Care & Research, 34, 2 (2013) 60-79.
26. Jeschke, M., G., Kamolz, L., P. ve Shahrokhi, S., Burn Care and Treatment: A Practical Guide, First Edition, Springer, New York, 2013.
27. Mertens, D., M., Jenkins, M., E. ve Warden, G., D., Outpatient Burn Management, The Nursing Clinics of North America, 32, 2 (1997) 343-364.
28. Peate, W., F., Outpatient Management of Burns, American Family Physician, 45, 3 (1992) 1321-1330.
29. Clayton, M., C. ve Solem, L., D., No Ice, No Butter Advice on Management of Burns for Primary Care Physicians, Postgraduate Medicine, 97, 5 (1995) 151-165.
30. Senarath-Yapa, K. ve Enoch, S., Management of Burns in the Community, Wounds UK, 5, 2 (2009) 38-41.

31. Wallace, A., B., The Exposure Treatment of Burns, The Lancet, 257, 6653 (1951) 501-504.
32. Wachtel, T., L., Berry, C., C., Wachtel, E., E. ve Frank, H., A., The Inter-Rater Reliability of Estimating the Size of Burns From Various Burn Area Chart Drawings, Burns, 26, 2 (2000) 156-170.
33. Lund, C., C. ve Browder, N., C., The Estimation of Areas of Burns, Surgery Gynecology And Obstetrics, 79, 1 (1944) 352-358.
34. Yastı, A., Ç., Şenel, E., Saydam, M., Özok, G., Çoruh, A. ve Yorgancı, K., Guideline and Treatment Algorithm for Burn Injuries, Ulusal Travma ve Acil Cerrahi Dergisi, 21, 2 (2015) 79-89.
35. Dirnberger, J., Giretzlehner, M., Ruhmer, M., Haller, H. ve Rodemund, C., Modelling Human Burn Injuries in a Three-Dimensional Virtual Environment, Studies in Health Technology and Informatics, 94, 1 (2003) 52-58.
36. Dirnberger, J., Giretzlehner, M., Luckeneder, T., Siegl, D., Haller, H., L. ve Rodemund, C., BurnCase 3D-Realistic Adaptation of 3-Dimensional Human Body Models, 7th International Conference on Medical Image Computing and Computer-Assisted Intervention, Eylül 2004, Saint-Malo, Proceedings Part II: 363-370.
37. Haller, H., L., Dirnberger, J., Giretzlehner, M., Rodemund, C. ve Kamolz, L., "Understanding Burns": Research Project BurnCase 3D-Overcome the Limits of Existing Methods in Burns Documentation, Burns, 35, 3 (2009) 311-317.
38. Neuwalder, J., M., Sampson, C., Breuing, K., H. ve Orgill, D., P., A Review of Computer-Aided Body Surface Area Determination: Sage II and EPRI's 3D Burn Vision, Journal of Burn Care and Rehabilitation, 23, 1 (2002) 55-59.
39. Prieto, M., F., Acha, B., Gómez-Cía, T., Fondón, I. ve Serrano, C., A System for 3D Representation of Burns and Calculation of Burnt Skin Area, Burns, 37, 7 (2011) 1233-1240.
40. Sheng, W., Zeng, D., Wan, Y., Yao, L., Tang, H. ve Xia, Z., BurnCalc Assessment Study of Computer-Aided Individual Three-Dimensional Burn Area Calculation, Journal of Translational Medicine, 12, 1 (2014) 242.
41. Herndon, D., N., Total Burn Care, Fourth Edition, Elsevier Saunders, China, 2012.
42. American Burn Association, Appendix B to Hospital Resources Document. Guidelines for Service Standards and Severity Classifications in the Treatment of Burn Injury, Bulletin of the American College of Surgeons, 69, 10 (1984) 24-28.
43. Acha, B., Serrano, C., Acha, J., I. ve Roa, L., M., CAD Tool for Burn Diagnosis, 18th International Conference on Information Processing in Medical Imaging, Temmuz 2003, Ambleside, Proceedings: 294-305.

44. Acha, B., Serrano, C., Palencia, S. ve Murillo, J., J., Classification of Burn Wounds Using Support Vector Machines, *Medical Imaging 2004: Image Processing*, Mayıs 2004, San Diego, Proceedings Volume 5370: 1018-1025.
45. Acha, B., Serrano, C., Acha, J., I. ve Roa, L., M., Segmentation and Classification of Burn Images by Color and Texture Information, *Journal of Biomedical Optics*, 10, 3 (2005) 034014.
46. Serrano, C., Acha, B., Gómez-Cía, T., Acha, J., I. ve Roa, L., M., A Computer Assisted Diagnosis Tool for the Classification of Burns by Depth of Injury, *Burns*, 31, 3 (2005) 275-281.
47. Fondón, I., Acha, B., Serrano, C. ve Sosa, M., New Characteristics for the Classification of Burns: Experimental Study, 3rd International Conference on Image Analysis and Recognition, Eylül 2006, Póvoa de Varzim, Proceedings Part II: 502-512.
48. Acha, B., Serrano, C., Fondon, I. ve Gomez-Cia, T., Burn Depth Analysis Using Multidimensional Scaling Applied to Psychophysical Experiment Data, *IEEE Transactions on Medical Imaging*, 32, 6 (2013) 1111-1120.
49. Serrano, C., Boloix-Tortosa, R., Gómez-Cía, T. ve Acha, B., Features Identification for Automatic Burn Classification, *Burns*, 41, 8 (2015) 1883-1890.
50. Deepak, L., Antony, J. ve Niranjana U., C., Hardware Co-Simulation of Skin Burn Image Analysis, 19th IEEE International Conference in High Performance Computing: Student Research Symposium, Aralık 2012, Pune.
51. Suvarna, M., Sivakumar, B. ve Niranjana, U., C., Classification Methods of Skin Burn Images, *International Journal of Computer Science and Information Technology*, 5, 1 (2013) 109-118.
52. Suvarna, M., Sivakumar, B., Kumar, K. ve Niranjana, U., C., Diagnosis of Burn Images Using Template Matching, k-Nearest Neighbor and Artificial Neural Network, *International Journal of Image Processing*, 7, 2 (2013) 191-202.
53. Suvarna, M. ve Venkatesh, N., Performance Measure and Efficiency of Chemical Skin Burn Classification Using KNN Method, *Procedia Computer Science*, 70, (2015) 48-54.
54. Badea, M., S., Vertan, C., Florea, C., Florea, L. ve Badoiu, S., Automatic Burn Area Identification in Color Images, IEEE International Conference on Communications, Mayıs 2016, Kuala Lumpur, Proceedings: 65-68.
55. Badea, M., S., Felea, I., I., Florea, L., M. ve Vertan, C., The Use of Deep Learning in Image Segmentation, Classification and Detection, arXiv preprint arXiv:1605.09612 (2016).
56. Tran, H., S., Le, T., H. ve Nguyen, T., T., The Degree of Skin Burns Images Recognition Using Convolutional Neural Network, *Indian Journal of Science and Technology*, 9, 45 (2016) 1-6.

57. Tran, H., Le, T., Le, T. ve Nguyen, T., Burn Image Classification Using One-Class Support Vector Machine, 4th International Conference on Context-Aware Systems and Applications, Kasım 2015, Vung Tau, Proceedings: 233-242.
58. Despo, O., Yeung, S., Jopling, J., Pridgen, B., Shekter, C., Silberstein, S., Fei-Fei, L. ve Milstein, A., BURNED : Towards Efficient and Accurate Burn Prognosis Using Deep Learning. <http://cs231n.stanford.edu/reports/2017/pdfs/507.pdf> 20 Ocak 2019.
59. Kuan, P., N., Chua, S., Safawi, E., B., Wang, H., H. ve Tiong, W., A Comparative Study of the Classification of Skin Burn Depth in Human, Journal of Telecommunication, Electronic and Computer Engineering, 9, 2-10 (2017) 15-23.
60. Betanzos, A., A., Varela, B., A. ve Martínez, A., C., Analysis and Evaluation of Hard and Fuzzy Clustering Segmentation Techniques in Burned Patient Images, Image and Vision Computing, 18, 13 (2000) 1045-1054.
61. Castro, A., Bóveda, C. ve Arcay, B., Analysis of Fuzzy Clustering Algorithms for the Segmentation of Burn Wounds Photographs, 3rd International Conference on Image Analysis and Recognition, Eylül 2006, Póvoa de Varzim, Proceedings Part II: 491-501.
62. Sonka, M., Hlavac, V. ve Boyle, R., Image Processing, Analysis and Machine Vision, First Edition, Springer, Boston, 1993.
63. Keller, J., M., Gray, M., R. ve Givens, J., A., A Fuzzy k-Nearest Neighbor Algorithm, IEEE Transactions on Systems, Man, and Cybernetics, 4 (1985) 580-585.
64. Lim, Y., W. ve Lee, S., U., On the Color Image Segmentation Algorithm Based on the Thresholding and the Fuzzy c-Means Techniques, Pattern Recognition, 23, 9 (1990) 935-952.
65. Pham, D.L., Spatial Models for Fuzzy Clustering, Computer Vision and Image Understanding, 84, 2 (2001) 285-297.
66. Wu, Z., D., Xie, W., X. ve Yu, J., P., Fuzzy c-Means Clustering Algorithm Based on Kernel Method, 5th International Conference on Computational Intelligence and Multimedia Applications, Eylül 2003, Xi'an, Proceedings: 49-54.
67. Wantanajittikul, K., Auephanwiriyaikul, S., Theera-Umpon, N. ve Koanantakool, T., Automatic Segmentation and Degree Identification in Burn Color Images, 4th Biomedical Engineering International Conference, Ocak 2012, Chiang Mai, Proceedings: 169-173.
68. Saranya, G., Thasny, S., S. ve Sobhia, K., Parallel Implementation of Wound Image Analysis System for Diabetic Patient, International Journal of Innovative Research in Science, Engineering and Technology, 5, 11 (2016) 112-118.
69. Otsu, N., A Threshold Selection Method from Gray-Level Histograms, IEEE Transactions on Systems, Man, and Cybernetics, 9, 1 (1979) 62-66.

70. Arel, I., Rose, D., C. ve Karnowski, T., P., Deep Machine Learning-A New Frontier in Artificial Intelligence Research, IEEE Computational Intelligence Magazine, 5, 4 (2010) 13-18.
71. Mukherjee, R., Manohar, D., D., Das, D., K., Achar, A., Mitra, A. ve Chakraborty, C., Automated Tissue Classification Framework for Reproducible Chronic Wound Assessment, BioMed Research International, 2014, (2014) 1-9.
72. Lecun, Y., Bottou, L., Bengio, Y. ve Haffner, P., Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, 86, 11 (1998) 2278-2324.
73. Lin, M., Chen, Q. ve Yan, S., Network in Network, arXiv preprint arXiv:1312.4400 (2013).
74. Long, J., Shelhamer, E. ve Darrell, T., Fully Convolutional Networks for Semantic Segmentation, 28th IEEE Conference on Computer Vision and Pattern Recognition, Haziran 2015, Boston, Proceedings: 3431-3440.
75. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C. ve Torr, P., H., S., Conditional Random Fields as Recurrent Neural Networks, 15th IEEE International Conference on Computer Vision, Aralık 2015, Santiago, Proceedings: 1529-1537.
76. Frank, E., Hall, M., A. ve Witten, I., H., The WEKA Workbench, Fourth Edition, Morgan Kaufmann, Burlington, 2016.
77. Sabeena, B. ve Rajkumar, P., Diagnosis and Detection of Automatic Skin Burn Area Color Images Identification of Burn Area Depth in Color Images, International Journal of Engineering Trends and Technology, 48, 1 (2017) 48-54.
78. Shapiro, L., G. ve Stockman, G., C., Computer Vision, First Edition, Pearson, New Jersey, 2001.
79. Subashini, T., S., Ramalingam, V. ve Palanivel, S., Automated Assessment of Breast Tissue Density in Digital Mammograms, Computer Vision and Image Understanding, 114, 1 (2010) 33-43.
80. Haralick, R., M., Shanmugam, K. ve Dinstein, I., Textural Features for Image Classification, IEEE Transactions on Systems, Man, and Cybernetics, 6, (1973) 610-621.
81. Schanda, J., Colorimetry: Understanding the CIE system, John Wiley & Sons, New Jersey, 2007.
82. Woods, B., J., Computer-Aided Detection of Malignant Lesions in Dynamic Contrast Enhanced MRI Breast and Prostate Cancer Datasets, Doktora Tezi, The Ohio State University, Department of Electrical and Computer Engineering, Ohio, 2008.
83. Dempster, A., P., Laird, N., M. ve Rubin, D., B., Maximum Likelihood from Incomplete Data via the EM Algorithm, Journal of the Royal Statistical Society: Series B (Methodological), 39, 1 (1977) 1-38.

84. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P. ve Süstrunk, S., SLIC Superpixels Compared to State-of-the-Art Superpixel Methods, IEEE Transactions on Pattern Analysis and Machine Intelligence, 34, 11 (2012) 2274-2282.
85. Phung, S., L., Bouzerdoun, A. ve Chai, D., A Novel Skin Color Model in Ycbr Color Space and Its Application to Human Face Detection, International Conference on Image Processing, Eylül 2002, Rochester, Proceedings Volume I: 289-292.
86. Lloyd, S., P., Least Squares Quantization in PCM, IEEE Transactions on Information Theory, 28, 2 (1982) 129-137.
87. Arthur, D. ve Vassilvitskii, S., k-Means++: The Advantages of Careful Seeding, 8th Annual ACM-Siam Symposium on Discrete Algorithms, Ocak 2007, New Orleans, Proceedings: 1027-1025.
88. Dunn, J., C., A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, Journal of Cybernetics, 3, 3 (1973) 32-57.
89. Bezdek, J., C., Ehrlich, R., Full, W., FCM: The Fuzzy c-Means Clustering Algorithm, Computers & Geosciences, 10, 2-3 (1984) 191-203.
90. Tsekouras, G., E. ve Sarimveis, H., A New Approach for Measuring the Validity of the Fuzzy c-Means Algorithm, Advances in Engineering Software, 35, 8-9 (2004) 567-575.
91. Avcı, D., Bulanık C Ortalamalar Kümeleme Tabanlı Hedef Eko Sinyal Ortamlarındaki Gerçek Hedef Sayısının Tespiti, Fırat Üniversitesi Mühendislik Bilimleri Dergisi, 29, 1 (2017) 73-82.
92. Pal, N., R. ve Bezdek, J., C., On Cluster Validity for the Fuzzy c-Means Model, IEEE Transactions on Fuzzy Systems, 3, 3 (1995) 370-379.
93. Özer, A., Cep Telefonu Elektromanyetik Işımasının İnsan Kulak Bölgesinde Sebep Olduğu Sıcaklık Artışının Termal Görüntüleme Tekniği, Bulanık C-Ortalama ve Beklenti Maksimizasyonu Yöntemleri ile Analizi, Yüksek Lisans Tezi, Kırıkkale Üniversitesi, Fen Bilimleri Enstitüsü, Kırıkkale, 2014.
94. Caliński, T. ve Harabasz, J., A Dendrite Method for Cluster Analysis, Communications in Statistics-Theory and Methods, 3, 1 (1974) 1-27.
95. Sokolova, M. ve Lapalme, G., A Systematic Analysis of Performance Measures for Classification Tasks, Information Processing & Management, 45, 4 (2009) 427-437.
96. Sutter, J., M. ve Kalivas, J., H., Comparison of Forward Selection, Backward Elimination, and Generalized Simulated Annealing for Variable Selection, Microchemical Journal, 47, 1-2 (1993) 60-66.
97. McCulloch, W., S. ve Pitts, W., A Logical Calculus of the Ideas Immanent in Nervous Activity, The Bulletin of Mathematical Biology, 5, 4 (1943) 115-133.
98. Öztemel, E., Yapay Sinir Ağları, 2. Baskı, Papatya Yayıncılık, İstanbul, 2006.

99. Yamamichi, S., Horibe, A., Aoki, T., Hosokawa, K., Hisada, T. ve Mori, H., Implementation Challenges for Scalable Neuromorphic Computing, 2017 Symposium on VLSI Technology, Haziran 2017, Kyoto, Proceedings: T182-T183.
100. Elmas, Ç., Yapay Sinir Ağları: Kuram, Mimari, Eğitim, Uygulama, 1. Baskı, Seçkin Yayıncılık, Ankara, 2003.
101. Yetkin, M., Tanker Şamandıra Bağlama Sistemlerinin Yapay Sinir Ağları Tekniğiyle Optimizasyonu, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2014.
102. Öğücü, M., O., Yapay Sinir Ağları ile Sistem Tanıma, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2006.
103. Akkaya, G., Yapay Sinir Ağları ve Tarım Alanındaki Uygulamaları, Atatürk Üniversitesi Ziraat Fakültesi Dergisi, 38, 2 (2007) 195-202.
104. Can, A., Yapay Sinir Ağları ile Heyelan Duyarlılık Haritalarının Üretilmesinde Farklı Algoritmaların Kullanımının Araştırılması, Yüksek Lisans Tezi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2014.
105. Dede, G., Yapay Sinir Ağları ile Konuşma Tanıma, Yüksek Lisans Tezi, Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2008.
106. Kim, S., Park, S., Lim, S. ve Kim, D., Classification Performance Analysis of Weight Update Method Applied to Various ConvNet Models, International Conference on Control and Robots, Eylül 2018, Hong Kong, Proceedings: 78-83.
107. Yu, X., Ye, C. ve Xiang, L., Application of Artificial Neural Network in the Diagnostic System of Osteoporosis, Neurocomputing, 214 (2016) 376-381.
108. Sağıroğlu, S., Beşdok, E. ve Erler, M., Mühendislikte Yapay Zeka Uygulamaları-I: Yapay Sinir Ağları, 1. Baskı, Ufuk Kitap Kırtasiye-Yayıncılık, Kayseri, 2003.
109. Andina, D. ve Pham, D., T, Computational Intelligence: For Engineering and Manufacturing, First Edition, Springer, Dordrecht, 2007.
110. Kaftan, İ., Batı Türkiye Gravite ve Deprem Katalog Verilerinin Yapay Sinir Ağları ile Değerlendirilmesi, Doktora Tezi, Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, İzmir, 2010.
111. Haykin, S., S., Neural Networks : A Comprehensive Foundation, Second Edition, Prentice Hall, New Jersey, 1998.
112. Emre, İ., Yarı-Kesikli Fermantasyonda Besleme ve Havalandırma Profillerinin Optimizasyonu, Doktora Tezi, Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2007.
113. Montavon, G., Orr, G., B. ve Müller, K., R., Neural Networks: Tricks of the Trade, Second Edition, Springer, Berlin, 2012.

114. Kartalopoulos, S., V., Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications, First Edition, Wiley-IEEE Press, New York, 1996.
115. Hebb, D., O., The Organization of Behavior: A Neuropsychological Theory, First Edition, John Wiley & Sons, New York, 1949.
116. Widrow, B. ve Hoff, M., E., Adaptive Switching Circuits, IRE WESCON Convention Record, 4, (1960) 96-104.
117. Hopfield, J., J., Neural Networks and Physical Systems with Emergent Collective Computational Abilities, Proceedings of the National Academy of Sciences, 79, 8 (1982) 2554-2558.
118. Kohonen, T., Self-Organized Formation of Topologically Correct Feature Maps, Biological Cybernetics, 43, 1 (1982) 59-69.
119. Rumelhart, D., E., Hinton, G., E. ve Williams, R., J., Learning Representations by Back-Propagating Errors, Nature, 323, 6088 (1986) 533-536.
120. Kumar, S., Neural Networks: A Classroom Approach, First Edition, Tata McGraw-Hill, New Delhi, 2004.
121. Hagen, M., T., Demuth, H., B. ve Beale, M., Neural Network Design, First Edition, PWS Publishing, Boston, 1996.
122. Öztürk, C., Yapay Sinir Ağlarının Yapay Arı Kolonisi Algoritması ile Eğitilmesi, Doktora Tezi, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Kayseri, 2011.
123. Fausett, L., V., Fundamentals of Neural Networks: Architectures, Algorithms, and Applications, First Edition, Prentice-Hall, New Jersey, 1994.
124. Auclair, A., Feed-Forward Neural Networks Applied to the Estimation of Magnetic Field Distributions, Yüksek Lisans Tezi, McGill University, Department of Electrical and Computer Engineering, Montreal, 2004.
125. Arbib, M., A., The Handbook of Brain Theory and Neural Networks, First Edition, The MIT Press, Cambridge, 1998.
126. Nisbet, R., Elder, J. ve Miner, G., Handbook of Statistical Analysis and Data Mining Applications, First Edition, Academic Press, Burlington, 2009.
127. Quinlan, J., R., C4.5: Programs for Machine Learning, First Edition, Morgan Kaufmann, San Mateo, 1993.
128. Pal, M. ve Mather, P., M., An Assessment of the Effectiveness of Decision Tree Methods for Land Cover Classification, Remote Sensing of Environment, 86, 4 (2003) 554-565.
129. Patel, N. ve Upadhyay, S., Study of Various Decision Tree Pruning Methods with Their Empirical Comparison in WEKA, International Journal of Computer Applications, 60, 12 (2012) 20-25.

130. Breiman, L., Friedman, J., H., Olshen, R., A. ve Stone, C., J., Classification and Regression Trees, First Edition, Chapman and Hall, New York, 1984.
131. Akar, Ö. ve Güngör, O., Rastgele Orman Algoritması Kullanılarak Çok Bantlı Görüntülerin Sınıflandırılması, Jeodezi ve Jeoinformasyon Dergisi, 1, 2 (2012) 139-146.
132. Liaw, A. ve Wiener, M., Classification and Regression by randomForest, R News, 2,3 (2002) 18-22.
133. Gislason, P., O., Benediktsson, J., A. ve Sveinsson, J., R., Random Forests for Land Cover Classification, Pattern Recognition Letters, 27, 4 (2006) 294-300.
134. Breiman, L., Random Forests, Machine Learning, 45, 1 (2001) 5-32.
135. Archer, K., J. ve Kimes, R., V., Empirical Characterization of Random Forest Variable Importance Measures, Computational Statistics and Data Analysis, 52, 4 (2008) 2249-2260.
136. Pal, M., Random Forest Classifier for Remote Sensing Classification, International Journal of Remote Sensing, 26, 1 (2005) 217-222.
137. Akman, M., Veri Madenciliğine Genel Bakış ve Random Forests Yönteminin İncelenmesi: Sağlık Alanında Bir Uygulama, Yüksek Lisans Tezi, Ankara Üniversitesi, Sağlık Bilimleri Enstitüsü, Ankara, 2010.
138. Friedman, J., H., Gradient Function Approximation: A Gradient Boosting Machine, The Annals of Statistics, 29, 5 (2001) 1189-1232.
139. Kolak, Ş., N., Çavuş, G., Gültekin, M. ve Kalıpsız, O., Çevik Model Kullanan Projelere Ait Veri Seti ile İş Gücü Kestirimi, 12. Ulusal Yazılım Mühendisliği Sempozyumu, Eylül 2018, İstanbul.
140. Duda, R., O., Hart, P., E. ve Stork, D., G., Pattern Classification, Second Edition, Wiley-Interscience, Hoboken, 2000.
141. Özkan, Y., Veri Madenciliği Yöntemleri, 1. Baskı, Papatya Bilim, İstanbul, 2008.
142. Han, J. ve Kamber, M., Data Mining: Concepts and Techniques, First Edition, Morgan Kaufmann, Waltham, 2001.
143. Mitchell, T., M., Machine Learning, First Edition, McGraw-Hill, New York, 1997.
144. Küçük, H., Tepe, C. ve Eminoğlu, İ., K-En Yakın Komşu Algoritması ve Destek Vektör Makinesi Yöntemleri ile EMG İşaretlerinin Sınıflandırılması, 21. Sinyal İşleme ve İletişim Uygulamaları Konferansı, Nisan 2013, Haspolat, Bildiri Kitabı: 1-4.
145. Theodoridis, S. ve Koutroumbas, K., Pattern Recognition, Fourth Edition, Academic Press, San Diego, 2008.

146. Montgomery, D., C., Peck, E., A. ve Vining, G., G., Introduction to Linear Regression Analysis, Fifth Edition, John Wiley & Sons, New Jersey, 2012.
147. Myers, R., H., Generalized Linear Models: With Applications in Engineering and the Sciences, Second Edition, Wiley, New York, 2010.
148. Nelder, J., A. ve Wedderburn, R., W., M., Generalized Linear Models, Journal of the Royal Statistical Society: Series A (General), 135, 3 (1972) 370-384.
149. McCullagh, P. ve Nelder, J., A., Generalized Linear Models, Second Edition, Chapman & Hall, London, 1989.
150. Goldberg, D., E., Genetic Algorithms in Search, Optimization and Machine Learning, First Edition, Addison-Wesley Professional, Boston, 1989.
151. Holland, J., H., Genetic Algorithms, Scientific American, 267, 1 (1992) 66-73.
152. Kingma, D., P. ve Ba, J., Adam: A Method for Stochastic Optimization, arXiv preprint arXiv:1412.6980 (2014).
153. Duchi, J., Hazan, E. ve Singer, Y., Adaptive Subgradient Methods for Online Learning and Stochastic Optimization, Journal of Machine Learning Research, 12 (2011) 2121-2159.
154. Fürnkranz, J., Separate-and-Conquer Rule Learning, Artificial Intelligence Review, 13, 1 (1999) 3-54.
155. Quinlan, J., R., Induction of Decision Trees, Machine Learning, 1, 1 (1986) 81-106.
156. Cendrowska, J., PRISM: An Algorithm for Inducing Modular Rules, International Journal of Man-Machine Studies, 27, 4 (1987) 349-370.
157. Webb, G., I. ve Brkic, N., Learning Decision Lists by Prepending Inferred Rules, AI'93 Workshop on Machine Learning and Hybrid Systems, Kasım 1993, Melbourne, Proceedings: 6-10.
158. Michalski, R., S., Mozetic, I., Hong, J. ve Lavrac, N., The Multi-Purpose Incremental Learning System AQ15 and Its Testing Application to Three Medical Domains, 5th National Conference on Artificial Intelligence, Ağustos 1986, Philadelphia, Proceedings: 1041-1045.
159. Clark, P. ve Niblett, T., The CN2 Induction Algorithm, Machine Learning, 3, 4 (1989) 261-283.
160. Fürnkranz, J. ve Widmer, G., Incremental Reduced Error Pruning, 11th International Conference on Machine Learning, Temmuz 1994, New Brunswick, Proceedings: 70-77.
161. Cohen, W., W., Fast Effective Rule Induction, 12th International Conference on Machine Learning, Temmuz 1995, Tahoe City, Proceedings: 115-123.

162. Çöllüoğlu Gülen, Ö., Veri Madenciliği Teknikleri ile Üstün Yetenekli Öğrencilerin İlgi Alanlarının Analizi, Doktora Tezi, Gazi Üniversitesi, Bilişim Enstitüsü, Ankara, 2014.
163. Quinlan, J., R., Learning Logical Definitions From Relations, Machine Learning, 5, 3 (1990) 239-266.
164. Fürnkranz, J., Pruning Algorithms for Rule Learning, Machine Learning, 27, 2 (1997) 139-172.
165. Ata, S. ve Yıldız, O., T., Searching for the Optimal Ordering of Classes in Rule Induction, 21st International Conference on Pattern Recognition, Kasım 2012, Tsukuba, Proceedings: 1277-1280.
166. Özdemir, O., Bulanık Mantık ile Belirlenmiş Öğrenme Stillere Dayalı Öğrenme Ortamlarının Öğrencilerin Başarı ve Tutumlarına Etkisi, Doktora Tezi, Fırat Üniversitesi, Sosyal Bilimler Enstitüsü, Elazığ, 2009.
167. Novini, A., R., Lighting and Optics Expert System for Machine Vision, SPIE Conference on Optics, Illuminations, and Image Sensing, Kasım 1988, Cambridge, Proceedings: 1005-1019.
168. Alpaydin, E., Introduction to Machine Learning, Second Edition, MIT Press, Cambridge, 2010.
169. Yonel, B., Mason, E. ve Yazıcı, B., Deep Learning for Passive Synthetic Aperture Radar, IEEE Journal of Selected Topics in Signal Processing, 12, 1 (2018) 90-103.
170. LeCun, Y., Jackel, L., D., Boser, B., Denker, J., S., Graf, H., P., Guyon, I., Henderson, D., Howard, R., E. ve Hubbard, W., Handwritten Digit Recognition: Applications of Neural Network Chips and Automatic Learning, IEEE Communications Magazine, 27, 11 (1989) 41-46.
171. Lecun, Y., Bengio, Y. ve Hinton, G., Deep Learning, Nature, 521, 7553 (2015) 436-444.
172. Krizhevsky, A., Learning Multiple Layers of Features from Tiny Images, Technical Report TR-2009, University of Toronto, Toronto, 2009.
173. Ghosh, S., Das, N., Das, I. ve Maulik, U., Understanding Deep Learning Techniques for Image Segmentation, ACM Computing Surveys, 52, 4 (2019) 73:1-73:35.
174. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. ve Salakhutdinov, R., Dropout: A Simple Way to Prevent Neural Networks from Overfitting, Journal of Machine Learning Research, 15, 1 (2014) 1929-1958.
175. Bjorck, J., Gomes, C., Selman, B. ve Weinberger, K., Q., Understanding Batch Normalization, 32nd Conference on Neural Information Processing Systems, Aralık 2018, Montréal, Proceedings: 7705-7716.

176. Song, J., Gao, S., Zhu, Y. ve Ma, C., A Survey of Remote Sensing Image Classification Based on CNNs, Big Earth Data, 3, 3 (2019) 232-254.
177. Vasant, P., Zelinka, I. ve Weber, G., W., Intelligent Computing and Optimization, First Edition, Springer, Switzerland, 2019.
178. Gülcü, A. ve Kuş, Z., Konvolüsyonel Sinir Ağlarında Hiper-Parametre Optimizasyonu Yöntemlerinin İncelenmesi, Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji, 7, 2 (2019) 503-522.
179. Josh, P. ve Adam, G., Deep Learning: A Practitioner's Approach, First Edition, O'Reilly Media, USA, 2017.
180. Radiuk, P., M., Impact of Training Set Batch Size on the Performance of Convolutional Neural Networks for Diverse Datasets, Information Technology and Management Science, 20, 1 (2017) 20-24.
181. Peng, C., Xiao, T., Li, Z., Jiang, Y., Zhang, X., Jia, K., Yu, G. ve Sun, J., MegDet: A Large Mini-Batch Object Detector, 30th IEEE/CVF Conference on Computer Vision and Pattern Recognition, Haziran 2018, Salt Lake City, Proceedings: 6181-6189.
182. Kirk, M., Thoughtful Machine Learning with Python: A Test-Driven Approach, First Edition, O'Reilly Media, USA, 2017.
183. Cheng, G., Li, Z., Yao, X., Guo, L. ve Wei, Z., Remote Sensing Image Scene Classification Using Bag of Convolutional Features, IEEE Geoscience and Remote Sensing Letters, 14, 10 (2017) 1735-1739.
184. Matignon, R., Neural Network Modeling Using SAS Enterprise Miner, First Edition, AuthorHouse, 2005.
185. De Sousa, C., A., R., An Overview on Weight Initialization Methods for Feedforward Neural Networks, International Joint Conference on Neural Networks, Temmuz 2016, Vancouver, Proceedings: 52-59.
186. Murayama, Y., Progress in Geospatial Analysis, First Edition, Springer, 2012.
187. Li, B., Shi, Y., Qi, Z. ve Chen, Z., A Survey on Semantic Segmentation, 18th IEEE International Conference on Data Mining, Kasım 2018, Singapore, Proceedings: 1233-1240.
188. Ronneberger, O., Fischer, P. ve Brox, T., U-Net: Convolutional Networks for Biomedical Image Segmentation, 18th International Conference on Medical Image Computing and Computer-Assisted Intervention, Ekim 2015, Munich, Proceedings Part III: 234-241.
189. Badrinarayanan, V., Kendall, A. ve Cipolla, R., SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 12 (2017) 2481-2495.

190. Lewis, D., D., Schapire, R., E., Callan, J., P. ve Papka, R., Training Algorithms for Linear Text Classifiers, 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Ağustos 1996, Zurich, Proceedings: 298-306.
191. Yang, Y. ve Liu, X., A Re-Examination of Text Categorization Methods, 22nd Annual International ACM Sigir Conference on Research and Development in Information Retrieval, Ağustos 1999, Berkeley, Proceedings: 42-49.
192. Cohen, J., A Coefficient of Agreement for Nominal Scales, Educational and Psychological Measurement, 20 1 (1960) 37-46.
193. Hastie, T., Tibshirani, R. ve Friedman, J., The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition, Springer, New York, 2009.
194. Liu, H. ve Motoda, H., Feature Extraction, Construction and Selection: A Data Mining Perspective, First Edition, Springer, Berlin, 1998.
195. Phetphoung, W., Kittimeteeworakul, N. ve Waranusast, R., Automatic Sushi Classification From Images Using Color Histograms and Shape Properties, 3rd ICT International Senior Project Conference, Mart 2014, Nakhon Pathom, Proceedings: 83-86.
196. Nair, P., A. ve Patel, B., C., Herpes Zoster (Shingles). <https://www.ncbi.nlm.nih.gov/books/NBK441824/>. 29 Ağustos 2019

ÖZGEÇMİŞ

1985 yılında Trabzon'da doğan Erdiñ KARAKULLUKÇU; ilkokulu Kurtuluş İlkokulu'nda, ortaokulu Cudibey Ortaokulu'nda okudu. Ortaokulu okul ikincisi olarak bitirdikten sonra lise eğitimini Tefvik Serdar Anadolu Lisesi'nde tamamladı. 2004 yılında girdiđi KTÜ, Fen Edebiyat Fakültesi, İstatistik ve Bilgisayar Bilimleri Bölümü'nden 2008'de sınıf ve fakülte birinciliđi dereceleriyle mezun oldu. 2005-2008 yılları arasında Türk Eğitim Vakfı'ndan öğrenci bursu aldı. 2009 yılında MEB tarafından sağlanan Yurt Dışına Lisansüstü Öğrenim Görmek Üzere Gönderilecek Öğrencileri Seçme ve Yerleştirme (YLSY) programına ilişkin bursu kazandı. Aynı yıl Marmara Üniversitesi'nde gördüğü İngilizce eğitimi sonrasında "İngilizce Eğitim Sertifikası" aldı. YLSY burs programı kapsamında dil eğitimi için gitmiş olduđu Philadelphia, Pennsylvania, ABD'de yer alan ELS Language Centers'tan "Bursta Üstün Başarı Sertifikası" ve "Yoğun İngilizce Programının 112. Seviyesini Tamamlama Sertifikası" elde etti. 2010 yılında ABD'de bulunan University of North Carolina at Chapel Hill'in İstatistik ve Yöneylem Araştırması Bölümü'nde yüksek lisans yapmaya hak kazandı. İki yıl süren eğitim sonrasında mezun oldu ve Türkiye'ye döndü. KTÜ'ye atama işlemlerinin devam ettiđi 2012-2013 yılları arasında Trabzon İl Millî Eğitim Müdürlüğü İnsan Kaynakları Yönetimi Bölümü ve Strateji Geliştirme Bölümü'nde memur olarak çalıştı. Bu süreçte kısa dönem askerlik görevini yerine getirdi. 2013 yılında KTÜ, Fen Fakültesi, İstatistik ve Bilgisayar Bilimleri Bölümü'ne araştırma görevlisi olarak atandı. 2014-2018 yılları arasında TÜBİTAK 2211-A Yurt İçi Doktora Bursu aldı. Eğitim hayatı boyunca birçok sportif faaliyette başarılar elde etti. 2006-2008 yılları arasında yer aldığı Fen Fakültesi öğrenci futbol takımı ile fakülteler arası futbol turnuvasında iki kez 3.'lük, 2011-2012 yılları arasında University of North Carolina at Chapel Hill'de öğrenciler arası salon futbolu turnuvasında 1.lık, 2017-2019 yılları arasında antrenör olarak görev aldığı Fen Fakültesi futbol takımı ile bir kez alt lig 2.liđi ve bir kez süper lig 3.'lüğü elde etti. Halen KTÜ, Fen Fakültesi, İstatistik ve Bilgisayar Bilimleri Bölümü'nde araştırma görevlisi olarak çalışmaktadır. Evli ve ikiz çocuk babasıdır. İyi derece İngilizce bilmektedir.

Şevik U., Karakullukçu E., Berber T., Akbaş Y. ve Türkyılmaz S., Automatic Classification of Skin Burn Colour Images Using Texture-Based Feature Extraction, IET Image Processing, 13, 11 (2019) 2018-2028.