

**T.C.
YOZGAT BOZOK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MEKATRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

Doktora Tezi

**DERİN ÖĞRENME VE MORFOLOJİK GÖRÜNTÜ
İŞLEME YÖNTEMLERİ İLE MR GÖRÜNTÜLERİNDE
OTOMATİK MENÜSKÜS SEGMENTASYONU**

Emre ÖLMEZ

**Tez Danışmanı
Doç. Dr. Orhan ER**

Yozgat 2020

**T.C.
YOZGAT BOZOK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MEKATRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

Doktora Tezi

**DERİN ÖĞRENME VE MORFOLOJİK GÖRÜNTÜ
İŞLEME YÖNTEMLERİ İLE MR GÖRÜNTÜLERİNDE
OTOMATİK MENÜSKÜS SEGMENTASYONU**

Emre ÖLMEZ

**Tez Danışmanı
Doç. Dr. Orhan ER**

**Bu çalışma, Yozgat Bozok Üniversitesi Bilimsel Araştırma Projeleri Birimi
Tarafından 6602c-MMF/18-196 nolu proje ile desteklenmiştir.**

Yozgat 2020

İÇİNDEKİLER

Sayfa

ÖZET.....	i
ABSTRACT	iii
TEŞEKKÜR	iv
TABLolar LİSTESİ.....	v
ŞEKİLLER LİSTESİ.....	vi
1. GİRİŞ	1
1.1. Menüsküs Segmentasyonu Üzerine Literatür Çalışması	7
1.2. Motivasyon	9
2. YAPAY SİNİR AĞLARI VE DERİN ÖĞRENME	10
2.1. Makine Öğrenmesi	12
2.1.1. Denetimli Öğrenme (Supervised Learning).....	12
2.1.2. Denetimsiz Öğrenme (Unsupervised Learning)	12
2.1.3. Yarı Denetimli Öğrenme (Semi-Supervised Learning)	13
2.1.4. Pekiştirmeli Öğrenme (Reinforcement Learning)	13
2.2. Yapay Sinir Ağı Temelleri	13
2.2.1. Perceptron	14
2.2.2. Sigmoid Nöron.....	15
2.3. Yapay Sinir Ağı Mimarisi	17
2.3.1. İleri Yayılım (Forward Propagation)	18
2.3.2. Maliyet Fonksiyonu (Cost Function).....	20
2.3.3. Geri Yayılım Algoritması (Back Propagation Algorithm)	21
2.4. Konvolüsyonel Sinir Ağı (Convolutional Neural Network, CNN).....	23
2.4.1. CNN Mimarisi	23
2.4.2. Konvolüsyon Katmanı (Convolutional Layer)	25
2.4.3. ReLU Katmanı (Rectified Linear Unit Layer).....	26
2.4.4. Pooling Katmanı (Pooling Layer).....	27

2.4.5. Tam Bağlantılı Katman (Fully Connected Layer)	28
2.4.6. Softmax Fonksiyonu	29
2.4.7. CNN Eğitimi	30
2.5. Bölgesel Tabanlı CNN (Regions with CNN, R-CNN).....	32
3. MORFOLOJİK GÖRÜNTÜ İŞLEME	36
3.1. Görüntü İşleme Altyapı	36
3.2. Görüntü Çeşitleri	37
3.2.1. Gri Ölçekli Görüntü (Grey Scale Image).....	38
3.2.2. Binary Görüntü (Binary Image).....	38
3.2.3. Renkli Görüntü (Color Image).....	39
3.2.4. Üç Boyutlu görüntü (3D Image).....	40
3.3. Pikseller Arası İlişkiler	42
3.4. Aşınma (Erosion)	43
3.5. Genişleme (Dilation)	44
3.6. Morfolojik Geriçatma (Morphological Reconstruction)	45
3.7. Sınır Temizleme (Border Clearing).....	46
3.8. Bağlı Bileşenler (Connected Components)	47
4. MENÜSKÜS SEGMENTASYONU	49
4.1. R-CNN ile Menüsküs Bölge Tespiti	50
4.1.1. Öğrenme Transferi (Transfer Learning)	50
4.1.2. Öğrenme Transferinde Kullanılan CNN (ConvNetBrain).....	51
4.1.3. Bölge Tespiti için R-CNN Modeli Tasarımı ve Eğitimi.....	53
4.1.4. MR Görüntülerinin R-CNN ile Taranması	55
4.1.5. Tarama Sonuçlarından Menüsküs Bölgesinin Çıkartılması	57
4.2. Çıkarılan Menüsküs Bölgesinden Menüsküs Segmentasyonu	59
5. SONUÇLAR VE DEĞERLENDİRME	69
5.1. Değerlendirme Yöntemi	69
5.1.1. Küp Benzerlik Katsayısı (Dice Similarity Coefficient, DSC)	70
5.1.2. Duyarlılık (Sensitivity)	71
5.1.3. Sonuçlar ve Değerlendirme	71

5.2. Özet ve Gelecek Çalışmalar	73
KAYNAKLAR	75
ÖZGEÇMİŞ.....	79



DERİN ÖĞRENME VE MORFOLOJİK GÖRÜNTÜ İŞLEME YÖNTEMLERİ İLE MR GÖRÜNTÜLERİNDE OTOMATİK MENÜSKÜS SEGMENTASYONU

Emre ÖLMEZ

**Yozgat Bozok Üniversitesi
Fen Bilimleri Enstitüsü
Mekatronik Mühendisliği Anabilim Dalı
Doktora Tezi**

2020; Sayfa: 79

Tez Danışmanı: Doç. Dr. Orhan ER

ÖZET

İnsan anatomisinde önemli bir işlevi olan menüsküsün muayenesinde Manyetik Rezonans (MR) Görüntüleme önemli bir rol oynamaktadır. MR görüntülerinin çeşitliliği dolayısıyla geleneksel görüntü işleme metotlarıyla menüsküsün otomatik olarak segmentasyonu oldukça zordur. Bir MR görüntü dizisi birden fazla görüntü karesinden oluşur ve ilgilenilen bölgenin özellikleri görüntü dizisi içerisindeki her bir görüntü karesinde değişiklik arz edebilir. Bu nedenle özellik çıkarma ve buna bağlı olarak segmentasyon için kod yazma işlemi oldukça zorlaşır. Derin öğrenme yöntemlerinden, Konvolüsyonel Sinir Ağları (CNN), özellik çıkarma işlemini doğrudan veri seti üzerinden gerçekleştirdiği için uzman tarafından özellik çıkarma gereksinimini ortadan kaldırır. Bölgesel Tabanlı Konvolüsyonel Sinir Ağları (Regions with CNN, R-CNN) ise CNN özelliklerini bölge önerileri ile kombine ederek CNN özelliklerinin nesne tespitinde kullanımını sağlar.

Bu tez çalışması iki ana aşamadan oluşmaktadır. Birinci aşamada, MR görüntülerinden menüsküs bölgesinin tespiti için bir R-CNN modeli tasarlanmıştır. Tasarlanan R-CNN'in eğitiminin az miktarda menüsküs verisi ve daha az hesaplama yükü ile gerçekleştirebilmesi için öğrenme transferi kullanılmıştır. Çalışmanın ikinci aşamasında, R-CNN ile tespit edilen menüsküs bölgesinden menüsküs dokusunun segmentasyonu morfolojik görüntü işleme yöntemleri ile birlikte iki farklı MR görüntü dizisi kullanılarak gerçekleştirilmiştir. Menüsküs bölgesinin R-CNN ile otomatik olarak algılanması, menüsküs segmentasyon

sürecini kolaylaştırırken iki farklı görüntü dizisinin farklı kontrast özelliklerinin morfolojik işlemlerle birlikte kullanılması menüsküs dokusunun çevresindeki dokulardan ayırt edilmesine olanak sağlamıştır. Bu çalışmanın sonucunda derin öğrenme yöntemleri ile morfolojik görüntü işleme yöntemlerinin bir arada kullanılmasıyla menüsküs segmentasyonunun başarılı bir şekilde uygulandığı görülmüş olup uzman hekime menüsküs tanılarında karar verebilmesi için bir karar destek sistemi olarak uygulanabileceği gösterilmiştir.



Anahtar Kelimeler: Otomatik Menüsküs Segmentasyonu; Bölgesel Tabanlı Konvolüsyonel Sinir Ağı; Bölge Önerileri; Transfer Öğrenmesi; Derin Öğrenme

AUTOMATIC SEGMENTATION OF MENISCUS IN MRI USING DEEP LEARNING AND MORPHOLOGICAL IMAGE PROCESSING

Emre ÖLMEZ

**Yozgat Bozok University
The Institute of Science and Technology
Department of Mechatronics Engineering
Doctor of Philosophy Thesis**

2020; Pages: 79

Supervisor: Assoc. Prof. Dr. Orhan ER

ABSTRACT

The meniscus has a significant function in human anatomy, and Magnetic Resonance Imaging (MRI) has an essential role in meniscus examination. Due to a variety of MRI data, it is excessively difficult to segment the meniscus with image processing methods. An MRI data sequence contains multiple images, and the region features we are looking for may vary from each image in the sequence. Therefore, feature extraction becomes more difficult, and hence, explicitly programming for segmentation becomes more difficult. Convolutional Neural Network (CNN) extracts features directly from images and thus eliminates the need for manual feature extraction. Regions with Convolutional Neural Network (R-CNN) allow us to use CNN features in object detection problems by combining CNN features with Region Proposals.

In this thesis study, we designed and trained an R-CNN for detecting meniscus region in MRI data sequence. We used transfer learning for training R-CNN with a small amount of meniscus data. After detection of the meniscus region by R-CNN, we segmented meniscus by morphological image analysis using two different MRI sequences. Automatic detection of the meniscus region with R-CNN made the meniscus segmentation process easier, and the use of different contrast features of two different image sequences allowed us to differentiate the meniscus from its surroundings.

Keywords: Automatic Segmentation of Meniscus; Regions with Convolutional Neural Network; Region Proposals; Transfer Learning; Deep Learning

TEŞEKKÜR

Bu tezin hazırlanma aşamasında yardımlarını ve desteklerini esirgemeyen tez danışmanım Doç. Dr. Orhan ER'e ve medikal anlamdaki bilgileriyle bize destek olan Doç. Dr. Murat KORKMAZ'a teşekkür ederim. Ayrıca bu zamana kadar desteklerini hiçbir zaman esirgemeyen sevgili hocalarım Prof. Dr. Feyzullah TEMURTAŞ ve Doç. Dr. Halit ÖZTEKİN'e de teşekkürü bir borç bilirim.

Maddi, manevi destekleriyle her zaman yanımda olan sevgili aileme, eşime ve zamanlarından fedakârlık eden oğullarıma sonsuz teşekkür ederim.



TABLÖLAR LİSTESİ

	<u>Sayfa</u>
Tablo 1.1. İlgili çalışmalar	9
Tablo 4.1. ConvNetBrain Hiper parametreleri ve eğitim sonuçları	52
Tablo 4.2. R-CNN eğitimi sonuçları ve Hiper parametreleri	55
Tablo 5.1. Karışıklık matrisi.....	69
Tablo 5.2. On farklı deneğe ait segmentasyon sonuçları.....	72
Tablo 5.3. İlgili çalışmalara ait sonuçlar. Sonuçlar ortalama artı/eksi standart sapma (eğer varsa) şeklinde verilmiştir.....	73

ŞEKİLLER LİSTESİ

Sayfa

Şekil 1.1. Menüsküs dokusu ön ve üst görüntüsü	1
Şekil 1.2. Sırasıyla aksiyal, koronal ve sagital düzlemlerde çekilmiş diz MR görüntüleri	2
Şekil 1.3. Sagital düzlemde çekilmiş 21 kareden oluşan bir diz MR görüntü dizisi ...	2
Şekil 1.4. Aynı kesitin farklı MR görüntü dizilerindeki görünüşü	2
Şekil 1.5. Sırasıyla PD ve T1 ağırlıklı görüntü dizilerinde menüsküs dokusu	6
Şekil 2.1. Derin öğrenme performans veri miktarı ilişkisi	10
Şekil 2.2. Üç girişli örnek perceptron modeli	14
Şekil 2.3. Üç girişli sigmoid nöron modeli	15
Şekil 2.4. Sigmoid fonksiyon	16
Şekil 2.5. İki gizli katmanlı YSA	18
Şekil 2.6. İleri beslemeli YSA	19
Şekil 2.7. Örnek CNN mimarisi	24
Şekil 2.8. Konvolüsyon işlemi	26
Şekil 2.9. ReLU işlemi	27
Şekil 2.10. 2x2 Filtre ile max pooling işlemi	28
Şekil 2.11. MR görüntülerinde max-pooling işlemi	28
Şekil 2.12. Tam bağlantılı katman	29
Şekil 2.13. Gradyan iniş	32

Şekil 2.14. R-CNN ile elde edilmiş çıkış görüntüsü.....	33
Şekil 2.15. R-CNN mimarisi.....	34
Şekil 3.1. NxM sayısal görüntü	36
Şekil 3.2. 8-Bit gri ölçekli diz ve beyin MR görüntüsü	38
Şekil 3.3. 8-Bit gri ölçekli görüntü ve T=55 ile eşiklendirilmiş binary görüntüsü	39
Şekil 3.4. RGB renkli görüntü	40
Şekil 3.5. Medikal görüntülemelerde anatomik düzlemler	41
Şekil 3.6. Aksiyal düzlemden elde edilmiş üç boyutlu beyin görüntüsü	41
Şekil 3.7. İki boyutlu görüntüde sırasıyla 4 ve 8-bitişiklik.....	42
Şekil 3.8. Üç boyutlu görüntülerde 6, 18 ve 26-Bitişiklik.....	43
Şekil 3.9. B yapılandırma elemanı ile X'in aşınma işlemi	44
Şekil 3.10. B yapılandırma elemanı ile X'in genişleme işlemi.....	45
Şekil 3.11. 3x3 yapılandırma elemanı ile morfolojik geriçatma işlemi	46
Şekil 3.12. Sınır temizleme işlemi	47
Şekil 3.13. Sınır temizleme işlemi için işaretleyici üretimi	47
Şekil 3.14. Bağlı bileşenlerin çıkarılması	48
Şekil 4.1. Çalışmanın ana aşamaları	49
Şekil 4.2. R-CNN ile menüsküs bölgesinin tespit aşamaları	50
Şekil 4.3. R-CNN eğitiminde kullanılan CNN (ConvNetBrain)	52
Şekil 4.4. R-CNN eğitim setinin tamamı	54
Şekil 4.5. Bir görüntü dizisi boyunca R-CNN tarama sonuçları.....	56

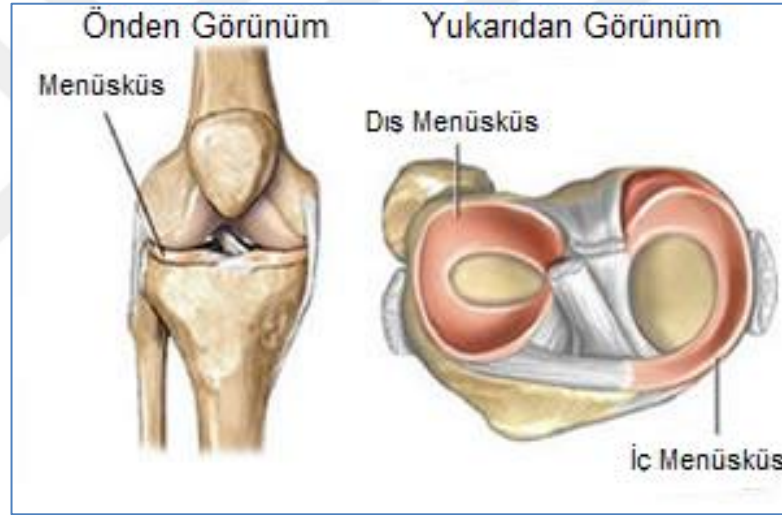
Şekil 4.6. Maksimum olmayan sonuçların bastırılması (Non-max suppression)	57
Şekil 4.7. Beş farklı deneğe ait tarama sonuçları.....	58
Şekil 4.8. Binary T1 ve PD ağırlıklı görüntülerin çarpımı	59
Şekil 4.9. Menüsküs bölgesinden menüsküs doku segmentasyonu.....	60
Şekil 4.10. T1 ağırlıklı görüntü dizisi	61
Şekil 4.11. PD ağırlıklı görüntü dizisi	61
Şekil 4.12. T1 ağırlıklı binary diz MR görüntü sekansı.....	61
Şekil 4.13. PD ağırlıklı binary diz MR görüntü sekansı	61
Şekil 4.14. T1 ve PD ağırlıklı binary görüntülerin çarpımı	61
Şekil 4.15. Menüsküs dokusu içermeyen karelerin temizlenmesi	63
Şekil 4.16. Sınırları temizlenmiş görüntü sekansı	63
Şekil 4.17. Sınırları temizlenmiş görüntülerden küçük nesnelerin kaldırılması.....	63
Şekil 4.18. Yatay aks filtresi ile ilk filtreleme	65
Şekil 4.19. Yatay aks filtresinden sonra küçük nesnelerin kaldırılması	65
Şekil 4.20. Yatay aks filtresi ile ikinci filtreleme	66
Şekil 4.21. İkinci yatay aks filtresinden sonra küçük nesnelerin kaldırılması.....	66
Şekil 4.22. Üç boyutlu yapılandırma elemanı ile genişleme	67
Şekil 4.23. Morfolojik rekonstrüksiyon ile bölge büyütme	67
Şekil 4.24. Segmente edilmiş menüsküs dokusu	68
Şekil 4.25. Segmente edilen menüsküs dokusunun üç boyutlu sunumu	68
Şekil 5.1. MR görüntüsüne ait segmentasyon sonuçları	70

KISALTMALAR LİSTESİ

MR	: Manyetik Rezonans
CNN	: Convolutional Neural Network
ConvNet	: Convolutional Neural Network
R-CNN	: Regions With Convolutional Neural Network
ROI	: Region of Interest
YSA	: Yapay Sinir Ağı
PD	: Proton Density
EM	: Expectation-Maximization
GMM	: Gaussian Mixture Model
ELM	: Extreme Learning Machine
DRF	: Discriminative Random Fields
HOG	: Histogram Oriented Gradients
LBP	: Local Binary Patterns
CPU	: Central Processing Unit
GPU	: Graphics Processing Unit
SSMs	: Statistical Shape Models
SVM	: Support Vector Machine
BT	: Bilgisayarlı Tomografi
RGB	: Red, Green, Blue
HSI	: Hue, Saturation, Intensity
CMY	: Cyan, Magenta, Yellow
CMYK	: Cyan, Magenta, Yellow, Black
TP	: True Positive
FN	: False Negative
FP	: False Positive
DSC	: Dice Similarity Coefficient

1. GİRİŞ

Menüsküs insan anatomisinin önemli bir parçasıdır. Şekil 1.1.'de [1] görüldüğü üzere, her dizde iç (medial) ve dış (lateral) olmak üzere iki adet fibro kıkırdak menüsküs dokusu vardır. Yarım ay şeklinde olan menüsküsün periferik kısımları dışbükey ve üçgen şeklindedir, dışarıdan içe doğru incelir ve tibia eklemi yüzünün üçte ikilik bölümünü kaplayacak şekilde yerleşir. Menüsküs, elastik yapıdadır ve sıkıştırma basınçlarına direnmek için yoğun ve sıkı örgü şeklindeki kolajen liflerinden oluşur. Menüsküsün şok emici işlevi diz stabilitesini sağlar ve bu durum ağırlığın eşit yayılmasına yardımcı olur ve eklem kıkırdaklarını yüksek basınçtan korur [2-4].

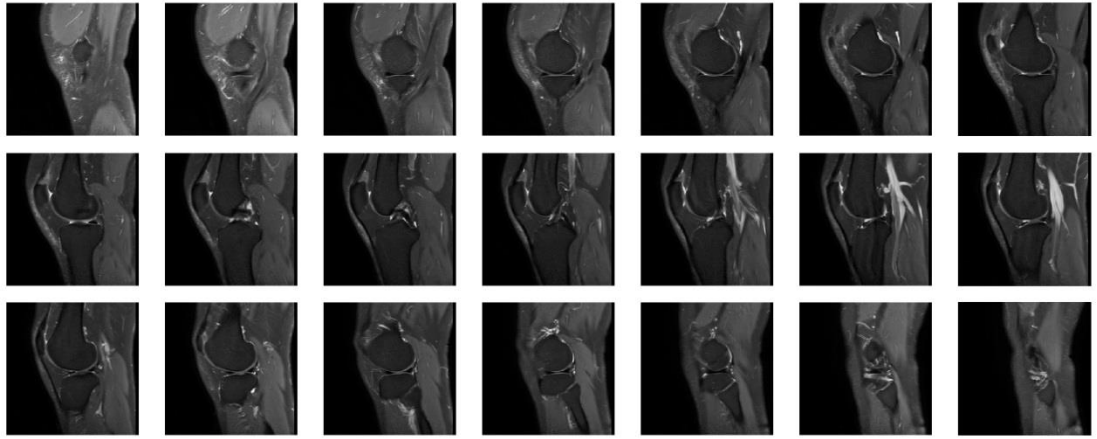


Şekil 1.1. Menüsküs dokusu ön ve üst görüntüsü

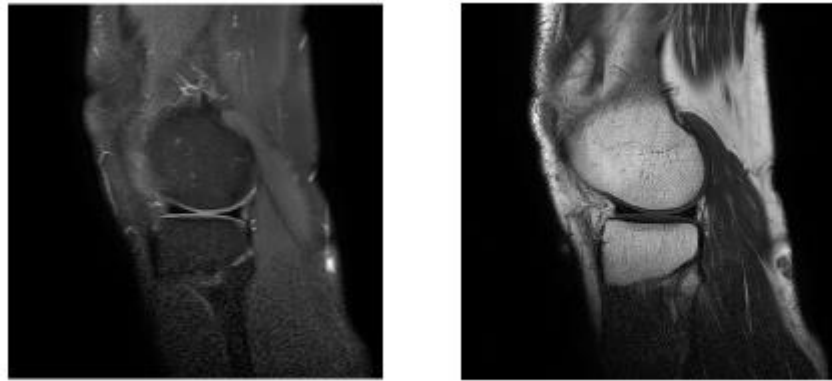
Menüsküs problemlerinin tespitinde, Manyetik Rezonans (MR) Görüntüleme sıklıkla kullanılan yöntemlerin başında gelir. Her bir MR görüntü dizisi belirli bir çekim tekniği ile bir düzlem boyunca çekilmiş (aksiyal, koronal veya sagittal) birden fazla görüntü karesinden oluşur (bkz. şekil 1.2., şekil 1.3.). MR görüntü dizileri farklı düzlemlerde elde edilebileceği gibi çekim tekniğine bağlı olarak aynı düzlemde farklı kontrast özelliklerinde de elde edilebilirler. MR görüntülerinin farklı düzlemlerde ve farklı çekim teknikleri ile elde dileyebilmesi MR görüntüleme ile bir dokuya ait sağlanan veri miktarını oldukça artırır. Farklı görüntü dizilerinde aynı dokulara ait görüntü piksellerinin parlaklık değerleri farklılık arz edebilir (bkz. şekil 1.4.).



Şekil 1.2. Sırasıyla aksiyal, koronal ve sagittal düzlemlerde çekilmiş diz MR görüntüleri



Şekil 1.3. Sagittal düzlemde çekilmiş 21 kareden oluşan bir diz MR görüntü dizisi



Şekil 1.4. Aynı kesitin farklı MR görüntü dizilerindeki görünüşü

Bir radyoloğun veya uzman bir hekiminin gerçekleştireceği MR görüntülerinde doku inceleme işlemlerinde farklı tekniklerle ve farklı düzlemlerde elde edilmiş birden fazla MR görüntü dizisi kullanılabilir. Her bir dizisinde birden fazla görüntü içeren farklı MR görüntü dizilerinin kullanımı uzmana inceleyeceği bölge için fazla

miktarda veri sađlarken inceleme iřlemi de bu artan veri nedeniyle zorlu ve daha fazla zaman alan bir hale gelebilmektedir.

MR grntleri zerinde inceleme yapılırken ođu zaman tek bir grnt karesi zerinden deđil grnt dizisi ierisinde yer alan birden fazla grnt karesi zerinden bir karara varılır. Tek bir grntde ortaya ıkabilecek olan bir kusur grnt dizisinin btn ele alınmadıđında yanlış karar vermeye neden olabilir. Ayrıca grnt ierisinde bazı detayların zlebilmesi iin uzmanın bir grnt karesinden diđerine geiřte grnty zihninde  boyutlu olarak birleřtirmesi gerekebilir. Bu nedenle ilgilenilen blgenin (Region of Interest, ROI) grntden ayrıştırılarak  boyutlu (3-D) bir modelinin oluřturulması uzmanın daha sađlıklı kararlar verebilmesi aısından faydalı olacaktır.

Genellikle medikal grnt iřlemede ilgilenilen blgenin segmentasyonu en temel problem olarak karřımıza ıkar [5-7]. ROI'nin segmentasyonu iin kullanılan yntemler manuel, otomatik ve yarı otomatik olarak  ana bařlıkta karřımıza ıkabilmektedir [8]. Bir uzman tarafından gerekleřtirilebilecek olan manuel segmentasyon gvenilir bir yntem olmasına karřın olduka zaman alıcıdır. zellikle MR grntleme gibi ok fazla veri ieren grntleme yntemlerinde manuel segmentasyon olduka zorlayıcı bir hal almaktadır. MR grntlerinde herhangi bir doku, organ veya lezyonun el ile segmentasyonu iin ilgili uzmanın grnt dizisinin tamamında tek tek iřlem yapması gerekir. Ayrıca manuel olarak yapılan segmentasyonlarda zaman kaybının yanı sıra kenar hassasiyeti gibi durumlar uzmandan uzmana deđiřecektir. Yarı otomatik segmentasyon uzmanın iřini azaltmasına rađmen yođun veri ieren sistemlerde manuel segmentasyonla aynı zorluklarla karřı karřıya bırakır. Otomatik segmentasyon en ideal yntem olmasına karřın MR grntleme gibi eřitliliđi ok fazla olan veri setlerinde otomatik bir algoritma yazmak olduka zorlayıcıdır ve bařarı oranları dřebilir. Segmente edilmek istenilen blgenin zelliđi farklı dzlemlerde farklı ekim teknikleri ile elde edilen grntlerde deđiřeceđi gibi aynı grnt dizisi ierisindeki farklı grnt karelerinde de farklılık arz edebilir. Bir MR grnt dizisi ierisinde segmente edilmek istenen dokunun farklı kısımları (bařlangı, bitiř vs.) farklı řekillerde

karşımıza çıkacaktır. Bu durum geleneksel metotlarla kod yazarak çözümü oldukça zorlayıcı hale getirdiğinden bu noktada makine öğrenmesi bize yardımcı olabilir.

Arthur Samuel 1959 yılında Makine öğrenmesini, bilgisayarlara açıkça kod yazmadan öğrenme yeteneği kazandıran çalışma alanı olarak tanımlamıştır [9]. Birçok farklı görüntü dizisi içeren MR görüntülemenin her bir görüntü dizisinde onlarca görüntü vardır. Bu görüntü kompleksliği ve çeşitliliği MR görüntülerinde ROI'yi otomatik olarak segmente etmek için özellik çıkararak açıkça kod yazmayı oldukça zorlaştırmaktadır. Kodlamanın zorlayıcı olduğu ve açıkça programlayarak istenilen başarı oranlarına ulaşılamadığı durumlarda makine öğrenmesi bize alternatif bir çözüm önerisi sunabilir.

Özel bir makine öğrenmesi tekniği olan derin öğrenme, özellikleri ve görevleri doğrudan bir veri seti üzerinden öğrenir. Derin öğrenme terimi genellikle çok katmanlı Yapay Sinir Ağlarının (YSA) eğitimini kastetmektedir [10]. Güvenilir sonuçları üretebilecek bir derin öğrenme uygulaması için gerekli olanlar bol miktarda etiketlenmiş veri (en azından yaklaşık bin adet görüntü) ve hesaplama gücüdür. Çoğu zaman bu çok fazla veri ve hesaplama gücü gereksiniminin üstesinden gelmek için öğrenme transferi (Transfer Learning) bize yardımcı olabilir [11, 12]. Bir problemin çözümü için kısıtlı miktarda veri ve hesaplama gücüne sahip olduğumuz durumlarda sıfırdan bir ağ tasarımı yapıp eğitmek yerine benzer problemlerin çözümü için tasarlanmış bir YSA'nın parametreleri ve mimarisini kullanarak ağ eğitimine başlamak problemin çözümünde bize yardımcı olabilir.

Bilgisayar görünümünün en temel problemlerinden biri olan görüntü sınıflandırma bir giriş görüntüsüne belirli bir etiket setinden herhangi birini atamak veya verilen bir görüntünün etiket kümesindeki her bir etiket değerine ait olma olasılığını hesaplamaktır. Verilen bir medikal görüntünün normal veya normal değil şeklinde bir değere atanması veya bir tümör görüntüsünün iyi huylu (benign) veya kötü huylu (malignant) şeklinde etiketlenmesi görüntü sınıflandırmaya birer örnektir [35]. Görüntü sınıflandırmada kullanılan en popüler derin Yapay Sinir Ağı (YSA) Konvolüsyonel Sinir Ağıdır (Convolutional Neural Network, CNN veya ConvNet) [13, 14]. Görüntü sınıflandırmada CNN kullanmanın en büyük avantajı CNN mimarisinin doğası gereği girişini iki veya üç boyutlu görüntü olarak varsaymasıdır.

Ayrıca Konvolüsyonel Sinir Ağları, minimum ön işleme ile görsel desenleri doğrudan görüntü piksellerinden tanıyacak şekilde tasarlanmıştır [15]. Krizhevsky ve arkadaşlarının ImageNet sınıflandırma probleminde [16], Girshick ve arkadaşlarının nesne tespitinde [17] elde ettiği başarılar ile CNN görüntü işlemede oldukça popüler hale gelmiştir.

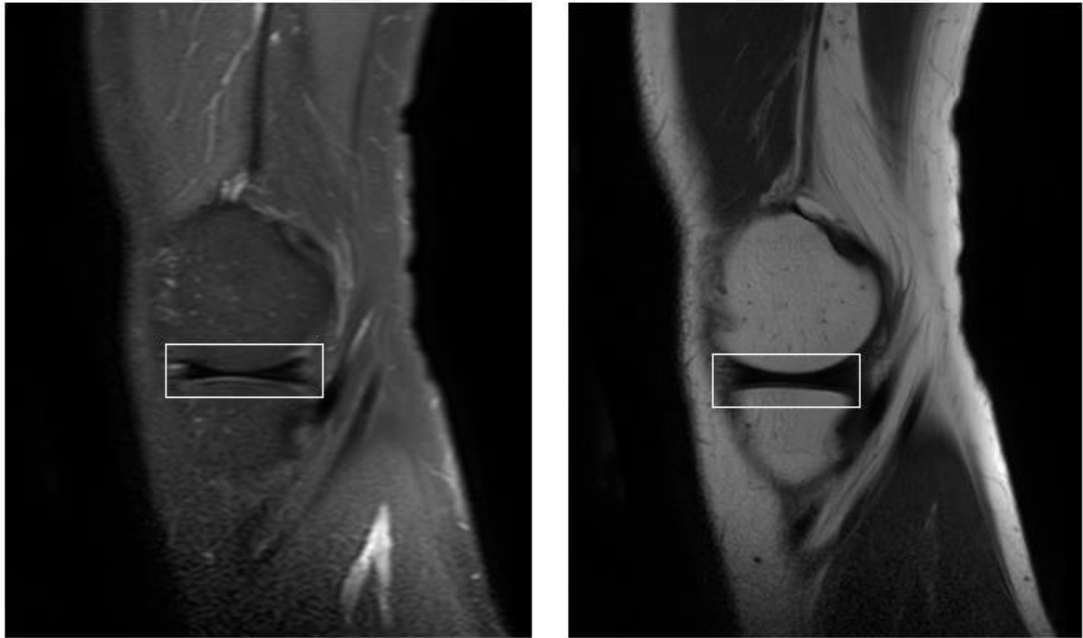
Görüntü sınıflandırmanın daha basit bir problem olması dolayısıyla nesne tanıma ve segmentasyon gibi farklı diğer bilgisayar görü problemleri de görüntü sınıflandırma problemine indirgenebilir. Girshick ve arkadaşları gerçekleştirdikleri çalışmada CNN özelliklerini nesne tespitinde kullanmışlardır. Bölge önerilerini CNN ile kombine ettikleri metotlarını Bölgesel Tabanlı CNN (Regions with CNN, R-CNN) olarak adlandırmışlardır [17]. Bu çalışmada verilen bir giriş görüntüsü içerisinde farklı boyutlarda yaklaşık 2000 adet bölge önerisi çıkartılır ve her bir bölge önerisi CNN ile sınıflandırılarak istenilen bölge olup olmadığına karar verilir. Bu şekilde bir nesne tespit problemi görüntü sınıflandırma problemine indirgenerek çözüm sağlanmış olur.

İki aşamadan oluşan bu tez çalışmasında sagittal düzlemde elde edilmiş diz MR görüntülerinde menüsküs dokusunun segmentasyonu için otomatik bir metot önerilmiştir. Çalışmanın ilk aşamasında menüsküs bölgesinin tespiti için R-CNN kullanılırken çalışmanın ikinci aşamasında tespit edilen menüsküs bölgesinden menüsküs dokusunun ayrıştırılması için sagittal düzlemde elde edilmiş iki farklı MR görüntü dizisi ile birlikte morfolojik görüntü işleme yöntemleri bir arada kullanılmıştır.

Çalışmanın ilk aşamasında PD Ağırlıklı (Proton Density Weighted) MR görüntü dizisinde R-CNN ile menüsküs bölgesinin tespiti gerçekleştirilmiştir. Menüsküs bölgesini tespit etmek için tasarlanan R-CNN'in eğitiminde 150 adet el ile çıkarılmış menüsküs bölgesi görüntüsü kullanılmıştır. R-CNN tasarımı ve eğitimi sıfırdan gerçekleştirmek yerine öğrenme transferi ile daha önce beyin MR görüntülerinin sınıflandırılmasında kullanılan bir CNN'in mimarisi ve parametreleri ile başlamak bu denli az miktarda veri ile R-CNN eğitimi mümkün kılmıştır. R-CNN eğitimi gerçekleştirildikten sonra menüsküs bölgesinin tespiti için PD ağırlıklı MR görüntü dizilerinde bulunan tüm görüntüler R-CNN ile taranarak her bir kare için menüsküs

bölgesi tespiti yapılmıştır. Her bir kare için elde edilen bölge tespit sonuçlarından belirli bir doğruluk değeri altında kalanlar elenip geriye kalan sonuçlar üst üste çakıştırılarak menüsküs bölgesinin tespiti sağlanmıştır.

Çalışmanın ikinci aşamasında tespit ettiğimiz menüsküs bölgesinden menüsküs dokusunun segmentasyonu için iki farklı MR görüntü dizisi (T1 ve PD ağırlıklı) ile birlikte morfolojik görüntü işleme yöntemleri kullanılmıştır. Şekil 1.5.'te görüldüğü gibi kullanılan T1 ve PD ağırlıklı görüntü dizilerinde menüsküs dokusu benzer yoğunluk değerine sahipken menüsküsü çevreleyen dokuların yoğunluk değerleri bu iki görüntü dizisinde de farklı özellik göstermektedir. Bu özellikten faydalanarak belirli bir eşik değeri ile binary hale getirilen T1 ve PD ağırlıklı görüntüler çarpılarak menüsküs dokusunun çevresindeki dokulardan ayrışması sağlanmış olur. Çarpma işleminden sonra elde ettiğimiz görüntüde çevresindeki dokulardan ayrılmış menüsküs dokusu ile birlikte bazı doku parçaları da kalacaktır. Çalışmanın devamında bu istenmeyen bileşenlerin görüntüden filtre edilmesi ve menüsküsün tekrar inşası için morfolojik görüntü işleme yöntemleri kullanılmıştır.



Şekil 1.5. Sırasıyla PD ve T1 ağırlıklı görüntü dizilerinde menüsküs dokusu

Çalışmanın birinci aşamasında özellik çıkarmanın oldukça zorlayıcı olduğu menüsküs bölgesi tespiti için derin öğrenme kullanılırken ikinci aşamada iki farklı MR görüntü dizisi kullanılarak elde edilen bir ayırt edici özellik sayesinde menüsküs

dokusunun tespit edilen bölgeden segmentasyonu morfolojik görüntü işleme yöntemleri kullanılarak gerçekleştirilebilmiştir.

Bu çalışmada diz MR görüntüleri içerisinde menüsküs dokusunun üç boyutlu olarak otomatik segmentasyonu amaçlanmıştır. Menüsküs dokusunun üç boyutlu olarak MR görüntülerinden çıkarılarak hekime sunulmasıyla menüsküs incelemelerinde hekime kolaylık sağlanarak yanlış teşhis koyma durumlarının azaltılması ve incelemede geçen zaman kayıplarının önüne geçilmesi hedeflenmiştir. Ayrıca bu çalışma ileride gerçekleştirilmesi planlanan otomatik menüsküs teşhisi modellerinin de ilk basamağını oluşturmaktadır. Bu çalışma ile otomatik olarak segmente edilen menüsküs dokularına gerek hekim gerek otomatik bir teşhis sistemi ile daha hızlı ve kolay tanı konulabilecektir.

1.1. Menüsküs Segmentasyonu Üzerine Literatür Çalışması

Literatürde MR görüntülerinde otomatik menüsküs segmentasyonu için sınırlı sayıda çalışma bulunmaktadır. Köse ve arkadaşları [18] menüsküs bölgesinin tespiti için histogram tabanlı bir metodu kenar tespit filtrelemesi ve istatistiksel segmentasyon tabanlı yöntemle birlikte kullandılar. Daha da fazlası tespit edilen bölgeden menüsküsü çıkarmak için bir şablon eşleştirme tekniği kullandılar. Fripp ve arkadaşları şekil tabanlı bir segmentasyon yöntemi geliştirdiler [19]. Daha önceki çalışmalarında kullandıkları kıkırdak segmentasyon yöntemlerini menüsküs bölgesinin tespiti için kullandılar. Sonrasında menüsküsün segmentasyonu için Beklenti-Maksimizasyon (Expectation-Maximization, EM) tabanlı Gaussian Karışım Modeli (Gaussian Mixture Model, GMM) kullandılar. Bir diğer şekil tabanlı segmentasyon yöntemi Paproki ve arkadaşları tarafından önerildi [20]. Paproki ve arkadaşları istatistiksel şekil modelleri içeren üç boyutlu (3D) bir aktif şekil modeli önerdiler.

Zhang ve arkadaşları çoklu kontrast MR görüntülerini kendi çalışmalarında kullandılar [21]. Multi kontrast MR görüntülerini kemik segmentasyonu ve segmente edilen kemiklere bağlı olarak menüsküs bölgesinin tespiti için kullandılar. Zhang ve arkadaşları Uç Öğrenme Makinelerini (Extreme Learning Machine, ELM) Ayrımcı Rastgele Alanlar (Discriminative Random Fields, DRF) ile birlikte kullandılar. Bu

şekilde DRF, özel bağımlılıkların sınıflandırmaya dâhil edilmesine olanak sağladı. Saygılı ve arkadaşları [22] menüsküs bölgesinin tespiti için global bir çerçeve ve kayan pencereler kullandılar. Özellik çıkarma için görüntü işleme yöntemlerinden “Histogram Oriented Gradients (HOG)” ve “Local Binary Patterns (LBP)” kullanırken regresyon için ELM ve rasgele alanlar kullandılar.

Felicia Aldrin [23] otomatik menüsküs segmentasyonu için iki boyutlu Konvolüsyonel U-Net kullanırken Tack ve arkadaşları [24] iki boyutlu ve üç boyutlu Konvolüsyonel U-Net’i İstatistiksel Şekil Modelleri ile kombine etti (Statistical Shape Models, SSMs). İstatistiksel Sekil Modellerini sonuç görüntüsünde kalan küçük bölgeleri filtrelemek ve menüsküsü tekrar inşa etmek için kullandılar.

Literatürde menüsküs segmentasyonu için farklı yöntemler kullanılmıştır. Menüsküs bölgesinin tespiti için Köse ve arkadaşları [18] ve Zhang ve arkadaşları [21] daha önce gerçekleştirdikleri kemik segmentasyonu sonuçlarını kullanırken Fripp ve arkadaşları [19] daha önce gerçekleştirdikleri kırık segmentasyonu sonuçlarını kullandılar. Bu tez çalışmasında çalışmada menüsküs bölgesinin tespiti için R-CNN kullanımı ile yüksek doğruluk oranlarına ulaşılırken çeşitliliği çok fazla olan MR görüntülerinde bölge tespiti için oldukça zorlayıcı olan özellik çıkarma işleminden kaçınılmış oldu.

Multi kontrast MR görüntülerini Zhang ve arkadaşları [21] kemik dokusunu ayırmak için kullanırken bu çalışmanın ikinci aşamasında Multi kontrast MR görüntüleri direkt olarak menüsküs dokusunu çevresinden ayırt etmek için kullanılmıştır. Çalışmanın ikinci aşamasında şekil tabanlı [18, 19, 20] veya makine öğrenmesi tabanlı [23, 24] herhangi bir yaklaşım kullanılmamış olup menüsküs dokusu çevresindeki dokulardan Multi kontrast MR görüntüleri kullanılarak net bir özelliklerle ayırt edilmiştir. Son olarak menüsküs dokusunu tekrar inşa etmek ve sonuç görüntüsündeki istenmeyen bileşenleri filtrelemek için şekil tabanlı bir yaklaşım yerine [24] morfolojik görüntü işleme yöntemleri kullanılmıştır.

Tablo 1.1. İlgili çalışmalar

Yazarlar	Yöntemler	Yıl
Köse ve ark.[18]	Thresholding ve Triangular Template Matching	2009
Fripp ve ark.[19]	Shape Model Fitting	2009
Paproki ve ark.[20]	3D Active Shape Model	2014
Zhang ve ark.[21]	ELM ve DRF	2013
Saygılı ve ark.[22]	ELM ve RF ile Kayan Pencereler	2017
Aldrin F.[23]	2D U-Net	2017
Tack ve ark.[24]	SSM ile 2D ve 3D U-Net	2018

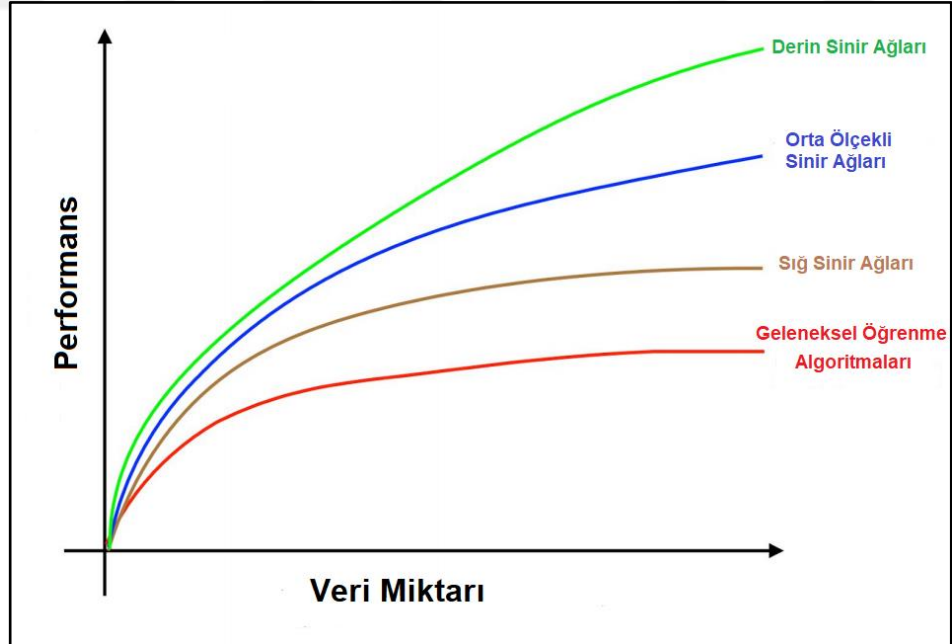
1.2. Motivasyon

Bu bölümde yapılan literatür çalışmasından da anlaşıldığı üzere MR görüntülerinde menüsküs segmentasyonu ile ilgili çalışmalar oldukça kısıtlıdır. Menüsküs hastalıkları toplumda oldukça yaygın olmasına rağmen MR görüntülerinde menüsküs segmentasyonu çalışmalarının azlığı bu görevin zorluğunu kanıtlar niteliktedir. Bu durumun temel nedeni MR görüntülerinin çeşitliliği ve ilgilenilen bölgenin bu görüntü çeşitliliği içerisinde farklı özellikler sergilemesidir. Bu çalışmanın temel motivasyonu derin öğrenme yöntemleri ve geleneksel metotları bir arada kullanarak MR görüntülerinde otomatik menüsküs segmentasyonu için alternatif arayışlara önemli bir katkı sağlayabilmektir.

Gerçekleştirilen çalışmanın hedefleri doğrultusunda tezin ikinci bölümünde yapay sinir ağları ve derin öğrenme üzerine genel bilgiler verilmiştir. Bu tez çalışmasının üçüncü bölümünde görüntü çeşitleri ve morfolojik görüntü işleme yöntemlerden bahsedilmiştir. Çalışmanın dördüncü bölümünde menüsküs bölgesinin tespiti ve tespit edilen bölgeden menüsküs dokusunun çıkarılması şeklinde iki ana aşamadan oluşan önerilen metot anlatılmıştır. Beşinci ve son bölümde ise önerilen metot ile elde edilen sonuçlar paylaşarak değerlendirilmiş ve gelecek çalışmalara ait öneriler sunulmuştur.

2. YAPAY SINIR AĞLARI VE DERİN ÖĞRENME

İnternet kullanımının artması, dijitalleşmeden kaynaklı gelişmeler ve dünya üzerinde özellikle son 20 yılda dijital ortamlarda kullanılan verinin artması sayesinde birçok problemin çözümü için ihtiyaç duyulan veri kümesi sağlanmış oldu. Donanım dünyasında yaşanan gelişmelerle birlikte (Yüksek kapasiteli CPU'lar, GPU'lar, bellek kapasitelerinin artması vs.) elde edilen yüksek miktarda verinin işlenebilmesi daha da mümkün hale geldi. Geçmiş daha öncelere dayanan derin öğrenmenin son yıllarda popülaritesinin hızla artmasının temelinde bahsettiğimiz bu iki neden vardır. Bol miktarda veri ve bu veriyi işleyebilecek uygun donanıma erişimin artmasıyla derin öğrenme yöntemlerinin kullanımı ve geliştirilmesi üssel olarak artış sağlamıştır. Derin öğrenmeye olan yönelimin artmasında bir diğer neden; Şekil 2.1.'de [10] görüldüğü gibi geleneksel makine öğrenme yöntemlerinin başarımlarının artan veri miktarına paralel olarak artmamasıdır. Dolayısıyla araştırmacıların bu geniş ölçekli veriyi etkin bir biçimde kullanacak olan yöntem arayışlarına girmeleri bu alandaki çalışmaların artmasında rol oynamıştır.



Şekil 2.1. Derin öğrenme performans veri miktarı ilişkisi

En az bir veya birden fazla gizli katmana sahip Yapay Sinir Ağlarına derin yapay sinir ağı denir. Derin öğrenme terimi literatürde çoğunlukla çok katmanlı yapay sinir

ağlarının eğitimini ifade eder [25]. YSA'ların derinliğinin artışına paralel olarak eğitimleri için gerekli veri ve donanım ihtiyacı da artış gösterir. Bir veya daha fazla gizli katmana sahip olan derin YSA'lar geçmişte de mevcut iken günümüzde başarı oranları ve popülaritesi bir hayli artmıştır. Bu durumun en önemli sebepleri derin YSA'ların eğitimleri için gerekli olan veri miktarı ile birlikte donanım ihtiyaçlarının da günümüzde daha kolay ulaşılabilir ve karşılanabilir olmasıdır. Görüntüleme sistemlerinin ucuzlaması ve internet kullanımında yaşanan gelişmeler son yıllarda makine öğrenmesinde kullanılan veri miktarının ciddi miktarda artmasını sağlarken, GPU gibi aygıtların yaygın kullanımıyla YSA'ların eğitim süreleri daha makul hale gelmiştir. Ayrıca öğrenme algoritmalarında yapılan yenilikler de Derin YSA'ların başarısını artırmada temel faktörlerden biridir.

Şekil 2.1.'deki grafikte artan veri miktarına bağlı olarak geleneksel makine öğrenme metotları ve YSA'ların veri miktarına bağlı olarak performans artışları gösterilmiştir. Grafikte yatay eksenimiz bir görevi gerçekleştirmek için sahip olduğumuz veri miktarını gösterirken dikey eksenimiz ise yine medikal görüntü sınıflandırıcısı veya bir veri seti üzerinden hastalık tahmini gibi bir görevi icra eden makine öğrenme tekniğinin başarı durumunu göstermektedir. Elimizde olan bu geniş ölçekli veri destek vektör makinesi (Support Vector Machine, SVM), lojistik regresyon (Logistic Regression) vs. gibi geleneksel makine öğrenme metotlarında başarı oranını belirli bir seviyeye kadar artırırken daha derin YSA'ların kullanılmasıyla başarı oranlarında ciddi artışlar gözlemlenmiştir.

Özel bir makine öğrenme tekniği olan derin öğrenme çoğunlukla çok geniş olan yapay sinir ağlarının eğitimini ifade etmektedir [25]. Bu bölümde derin öğrenme kavramının anlaşılabilmesi için makine öğrenmesinden başlayarak gerekli olan temel kavramlar açıklanacaktır. Makine öğrenmesine ait temel kavramlar açıklandıktan sonra YSA'ların temelleri, mimarisi ve eğitiminden bahsedilerek görüntü sınıflandırmada sıklıkla kullanılan bir derin YSA olan CNN'in mimarisi ve eğitiminden bahsedilecektir. Son olarak ta bölge önerileri ile CNN özelliklerini kombine ederek verilen bir görüntü içerisinde nesne tespit etme işlemini gerçekleştiren R-CNN'in mimarisi ve tasarımından bahsedilecektir.

2.1. Makine Öğrenmesi

Arthur Samuel Makine öğrenmesini 1959'da bilgisayarlara açıkça kod yazmadan öğrenme kabiliyetini kazandıran çalışma alanı olarak tanımladı [9]. Daha yeni ve formal bir tanımlama olarak Tom Mitchell makine öğrenmesini şu şekilde tanımlar; bir bilgisayar programının bir T görevini E tecrübesinden P performans ölçümü ile öğrenmesidir. Eğer P ile ölçülen performans T görevi üzerinde E ile iyileşiyorsa bu işlem makine öğrenmesi ile gerçekleştirilebilir [26]. Bu iki tanımı birleştirerek makine öğrenmesini şu şekilde tanımlayabiliriz: Bir görevi açıkça kod yazarak gerçekleştirmek yerine performans ölçümüne bağlı tecrübe ile bilgisayarın öğrenmesini sağlamaktır. Veri tabanlı yaklaşım da diyebileceğimiz bu yaklaşımda bilgisayar gerekli tecrübeyi bir veri seti üzerinden öğrenir.

Makine öğrenme algoritmaları; veri setinden öğrenme tipine göre denetimli (supervised), denetimsiz (unsupervised), yarı denetimli (semi-supervised) ve pekiştirmeli (reinforcement) olarak sınıflandırılır.

2.1.1. Denetimli Öğrenme (Supervised Learning)

Denetimli öğrenme etiket adı da verilen cevap değişkeninin bulunduğu veri setlerine uygulanır. Burada cevap değişkeni sürekli veya kategorik olabilir. Denetimli öğrenme bir dizi eğitim örneğinden oluşan bu etiketli eğitim verilerinden bir fonksiyon çıkarımı yapar [27]. Örneğin bir dizi hastaya ait olan veri setinde cevap değişkenimiz her bir hastanın kanser olup olmadığı olabilir veya verilen bir ülke veya şehirdeki ev fiyatlarını veren veri kümesinde cevap değişkenimiz ev fiyatları olabilir [28]. Denetimli öğrenme örnek girdi cevap çiftlerine göre bir girdiyi bir cevaba eşleyen fonksiyonu öğrenerek eğitimde kullanılmayan yeni örnekler için kullanılabilecek fonksiyon çıkarımını üretir [29].

2.1.2. Denetimsiz Öğrenme (Unsupervised Learning)

Cevap değişkenlerinin diğer adıyla etiketli verilerin olmadığı veri setlerinde öğrenme denetimsiz olarak gerçekleştirilir. Denetimsiz öğrenme veri kümesi içerisinde bazı benzerlik veya mesafe ölçümlerine göre gerçekleşir. Denetimsiz öğrenmede en çok kullanılan teknik kümelemedir. Denetimsiz öğrenmeye örnek olarak sosyal ağ

analizi, görüntü segmentasyonu, klimatoloji, pazarlamada müşteri segmentasyonu ve daha birçok uygulama sayılabilir [28].

2.1.3. Yarı Denetimli Öğrenme (Semi-Supervised Learning)

Yarı denetimli öğrenme geniş miktarda etiketsiz verinin yanı sıra az miktarda etiketli veriyi kullanan öğrenme algoritmasıdır. Geniş miktardaki etiketsiz veri az miktarda etiketli veri ile birlikte kullanıldığında denetimsiz öğrenmeye göre daha yüksek başarı elde edilirken denetimli öğrenmenin zaman ve maliyet gereksinimlerinden de kaçınılmış olur [30]. Denetimli öğrenmede veri setinin hazırlanması ve veri setindeki her bir verinin etiketlenmesi zaman ve maliyet gerektiren işlemlerdir.

2.1.4. Pekiştirmeli Öğrenme (Reinforcement Learning)

En iyi sonucu üretebilmek için denetimli ve denetimsiz öğrenme algoritmalarının her ikisi de temiz ve doğru veriye ihtiyaç duyar. Algoritmanın görmediği örnekler üzerinde de çalışabilmesi için eğitim setinde kullanılan verinin kapsayıcı olması gerekir. Pekiştirmeli öğrenme, yalnızca verilerin başlangıç durumunun bir girdi olarak mevcut olduğu ve tek bir olası cevabın olmadığı ancak çok fazla sonucun mümkün olduğu durumlarda ideal bir seçimdir [28]. Robotik ve sürücüsüz araçlarda uygulanan makine öğrenme algoritmaları pekiştirmeli öğrenme algoritmaları sınıfındandır. Pekiştirmeli öğrenme algoritmaları sürekli olarak çevresinden öğrenen algoritmalarlardır.

2.2. Yapay Sinir Ağı Temelleri

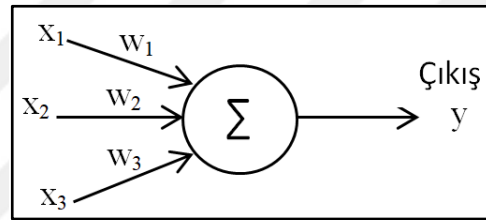
Özel bir makine öğrenme tekniği olan Yapay sinir ağları (YSA) memeli cerebral korteksinden esinlenerek tasarlanmış daha küçük ölçekli işleme aygıtlarıdır [31]. YSA'lar değişkenler arasındaki doğrusal olmayan ilişkileri öğrenebilme ve yüksek dereceli ilişkileri tanıma konusunda oldukça yeteneklidir. YSA'lar denetimli ve denetimsiz öğrenme algoritmalarının her ikisiyle de uygulanabilirler [28].

Olabilecek en basit sinir ağı tek bir nöronun hesaplamalı modeli olan Perceptron'dur. Perceptron Warren McCulloch ve Walter Pitts'in daha önceki çalışmalarından esinlenilerek 1950'lerde Frank Rosenblatt tarafından geliştirilmiştir [32]. Günümüzde YSA çalışmalarında kullanılan temel nöron modeli sigmoid nöron

olarak adlandırılır. Sigmoid nöronun yapısına bakmadan önce Perceptron'un yapısını anlamak sigmoid nöronun çalışmasını anlama noktasında faydalı olacaktır.

2.2.1. Perceptron

Birkaç tane binary girişe sahip olan Perceptron bir adet binary çıkışa sahiptir. Rosenblatt Perceptron'un çıkışının hesaplanması için şu basit kuralı önerir, ilgili girdilerin her biri önemini ifade eden ağırlıklarla (weights) çarpılır ve tüm bu çarpımlar toplanır. Eğer sonuç belirli bir eşiğin üzerindeyse Perceptron 1 değerini değilse 0 değerini üretir. Aşağıda şekil 2.2.'de 3 adet girişe sahip örnek bir Perceptron modeli görülmektedir. N adet girişe sahip Perceptron'un çıkışının hesaplanması ise eşitlik 2.1.'de verilmiştir.



Şekil 2.2. Üç girişli örnek perceptron modeli

$$Çıkış, y = \begin{cases} 0 & \text{eğer } \sum_{j=1}^N w_j x_j \leq \text{threshold (eşik)} \\ 1 & \text{eğer } \sum_{j=1}^N w_j x_j > \text{threshold (eşik)} \end{cases} \quad (2.1.)$$

w : Ağırlık vektörü
 x : Giriş vektörü
 j : İndis (1 ... N)

Eşitlik 2.1.'de görülen tanımı daha da basitleştirmek için $\sum_j w_j x_j$ ifadesi yerine noktasal çarpım kullanır ve eşitlik 2.1.'deki gibi eşik değeri (threshold) eşitliğin diğer tarafına atıp negatif eşik değere bias (b) dersek Perceptron çıkışımız eşitlik 2.2.'deki hale gelir.

$$Çıkış, y = \begin{cases} 0 & \text{eğer } w \cdot x + b \leq 0 \\ 1 & \text{eğer } w \cdot x + b > 0 \end{cases} \quad (2.2.)$$

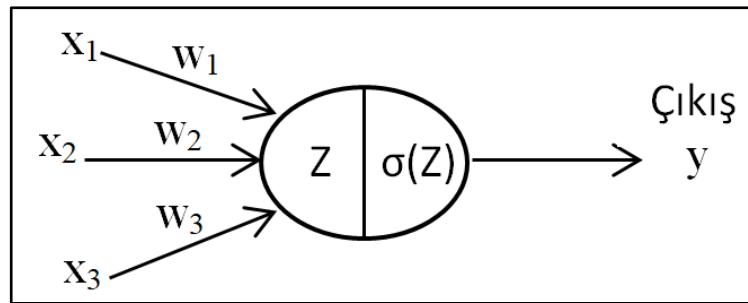
Burada bias'i Perceptron çıkışının hangi kolaylıkta lojik 1 değerini üreteceğinin ölçüsü olarak tanımlayabiliriz veya biyolojik olarak Perceptron'un hangi kolaylıkta ateşleyeceğinin ölçüsü diyebiliriz [32]. Eşitlik 2.2.'yi şekil 2.2.'deki 3 girişli örnek perceptron modeli için yazacak olursak ifademiz eşitlik 2.3.'te görüldüğü gibi olacaktır.

$$\text{Çıkış} = \begin{cases} 0 & \text{eğer } w_1x_1 + w_2x_2 + w_3x_3 + b \leq 0 \\ 1 & \text{eğer } w_1x_1 + w_2x_2 + w_3x_3 + b > 0 \end{cases} \quad (2.3.)$$

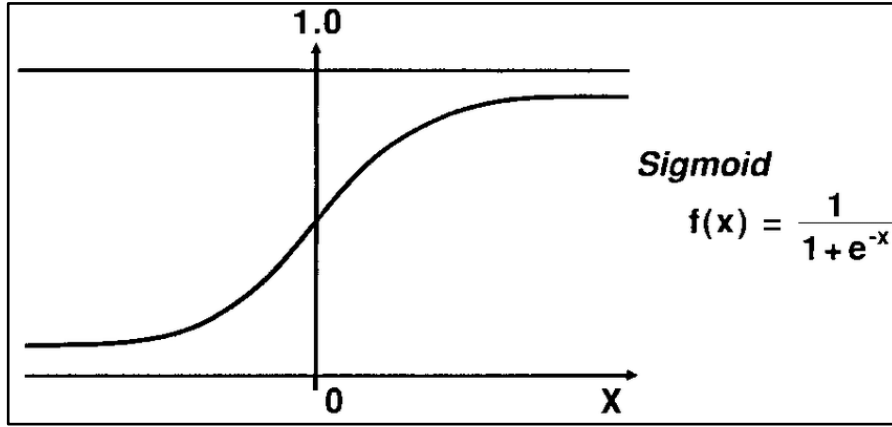
Eşitlik 2.3.'te görüldüğü gibi Perceptron yalnızca iki değerli (0 ve 1) çıkış üretebilen en ilkel yapay nöron modelidir. Çarpımların toplamı şeklinde hesaplanan bu binary çıkış yapısıyla Perceptron doğrusal (linear) bir sınıflandırıcıdır. Doğrusal olmayan daha karmaşık problemlerin çözümü için doğrusal olmayan (non-linear) bir aktivasyon fonksiyonuna sahip olan sigmoid nöron modeli geliştirilmiştir.

2.2.2. Sigmoid Nöron

Sigmoid Nöron, Perceptron'a benzer şekilde her bir girişi için ayrı ayrı ağırlıklara ve bir bias'e sahiptir. Perceptron'dan farklı olarak Sigmoid Nöron'un giriş değerleri sadece lojik 0 ve lojik 1 değerleri değil 0 ile 1 arasında herhangi bir reel sayı olabilir. Sigmoid Nöron çıkış değeri ise Perceptron'a benzer şekilde hesapladığımız çıkışın sigmoid fonksiyona (Bkz. Şekil 2.3.) uygulanmış halidir. Bu şekilde sigmoid nöron Perceptron'da çıkış olarak üretilen sadece 0 veya 1 değerleri yerine 0 ile 1 arasındaki herhangi bir reel sayı değerini de çıkış olarak üretebilir ve sigmoid fonksiyonun uyguladığı doğrusal olmayan aktivasyon sayesinde doğrusal olmayan bir sınıflandırıcı olarak kullanılabilir.



Şekil 2.3. Üç girişli sigmoid nöron modeli



Şekil 2.4. Sigmoid fonksiyon

Direk olarak bakıldığında anlamsız veya karmaşık gibi gözükebilen Sigmoid Nöron yapısı Perceptron’la beraber ele alındığında daha anlaşılır hale gelmektedir. Perceptron’un lojik değer alabilen giriş ve çıkışları yerine Sigmoid Nöron’un 0-1 arası reel sayı değerlerini alabilen giriş ve çıkışları Sigmoid Nöron’u daha hassas ayarlanabilir bir model olarak karşımıza çıkarmaktadır. Şekil 2.3. ve şekil 2.4.’te sırasıyla üç girişli bir Sigmoid Nöron modeli ve Sigmoid Fonksiyon’un şekli görülmektedir. Sigmoid Nöron’un çıkışının hesaplanması eşitlik 2.4. ve 2.5.’de görüldüğü gibi gerçekleşir. İlk olarak her bir nöron girişi kendisine ait ağırlık değeriyle çarpılır ve tüm bu çarpımlar toplanarak üzerine bir bias değeri eklenir. Elde edilen bu değer sigmoid fonksiyona uygulanarak nöron çıkışı elde edilir. Eşitlik 2.6.’da şekil 2.3.’te görülen modele ait z değerinin hesaplanması gösterilmiştir.

$$z = \sum_{j=1}^N w_j x_j + b \quad (2.4.)$$

$$\text{Çıkış}, y = \sigma(z) = \frac{1}{1+e^{-z}} \quad (2.5.)$$

$$z = w_1 x_1 + w_2 x_2 + w_3 x_3 + b \quad (2.6.)$$

w_j : Ağırlık matrisi

x_j : Giriş vektörü

j : İndis (1 ... N)

σ : Sigmoid Fonksiyon

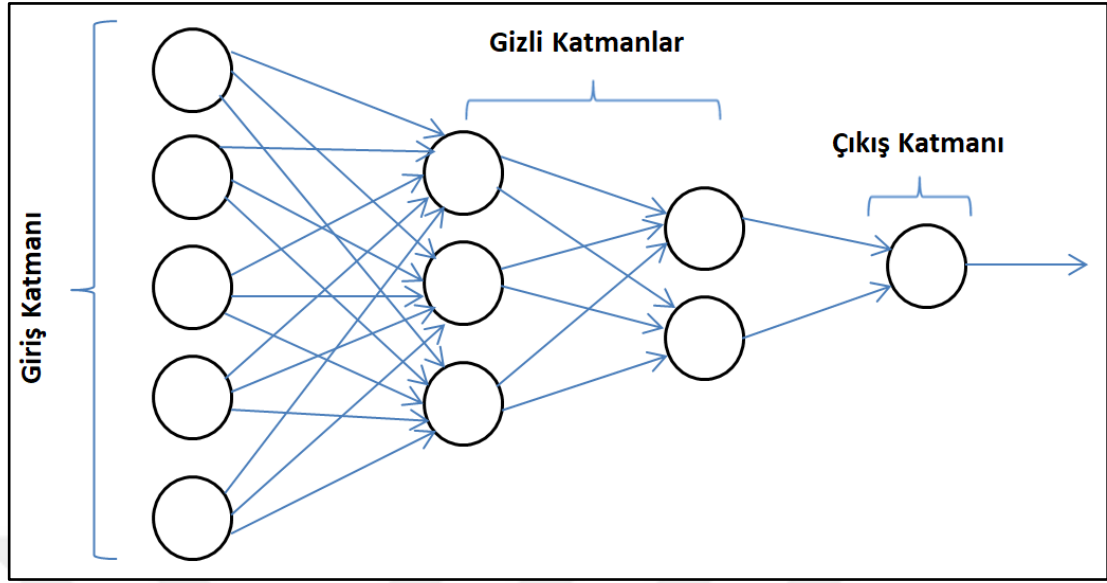
z : Toplama Fonksiyonu

Eğer sigmoid fonksiyon gibi doğrusal olmayan bir fonksiyonu nöron çıkışına uygulamazsa nöron çıkışı basit bir doğrusal fonksiyon olacağından oldukça kısıtlı bir öğrenme kapasitesine sahip olur. Yapay sinir ağları ile karmaşık ve lineer olmayan problemlerin çözümü hedeflendiğinden nöronlar için lineer olmayan aktivasyon fonksiyonları tercih edilir. Sigmoid nöron temel bir model olup YSA’larda nöronlar için aktivasyon fonksiyonu olarak sadece sigmoid fonksiyon kullanılmaz. Hiperbolik tanjant fonksiyonu, ReLU (Rectified Linear Units) fonksiyonu, Leaky ReLU fonksiyonu ve softmax fonksiyonu gibi doğrusal olmayan fonksiyonlar da YSA’larda sıklıkla kullanılırlar.

2.3. Yapay Sinir Ağı Mimarisi

Tipik olarak YSA’lar her birinde belirli sayıda nöron bulunan katmanlardan oluşur. Verinin alındığı katmana giriş katmanı en son işlenmiş verinin çıktığı katmana ise çıkış katmanı adı verilir. Giriş ve çıkış katmanı arasında kalan katmanlara ise gizli katman adı verilir. Giriş katmanı aracılığıyla YSA tarafından alınan veriler bir veya daha fazla sayıdaki gizli katmanda işlenerek çıkış katmanına aktarılır. Aşağıda şekil 2.5.’te görülen örnek bir YSA’da en soldaki katman giriş katmanı en sağdaki ise çıkış katmanıdır. Şekil 2.5.’te görülen YSA iki adet gizli katmana sahip olup gizli katman sayısı uygulamaya göre değişiklik arz edebilir ve daha fazla sayıda gizli katman içerebilir. Literatürde çoğunlukla YSA’ların katman sayıları verilirken giriş katmanı dâhil edilmez ve şekil 2.5.’te görülen YSA üç katmanlı bir YSA olarak ifade edilir. Perceptronlar yerine Sigmoid nöronlardan oluşan modern çok katmanlı ağlar literatürde yine de zaman zaman MLP (Multi-Layer Perceptron) olarak adlandırılabilirler.

YSA tasarımında giriş ve çıkış katmanlarındaki nöronların sayısı eldeki problemin tipine göre direk olarak kolayca belirlenirken gizli katman sayısını ve her bir gizli katmandaki nöron sayısını belirlemek biraz daha karmaşıktır. Örneğin 2 girişli bir XOR kapısı problemini çözen bir YSA otomatik olarak 2 giriş ve 1 çıkışa sahiptir. Veya 128x128 bit renkli bir resmin bir insana ait olup olmadığına karar veren bir YSA, 49152 (128x128x3) giriş ve bir çıkışa sahiptir ve çıkış değeri belirlenen eşik değerden büyük ise giriş imgesi insan, küçük ise insan değil olarak etiketlenir.



Şekil 2.5. İki gizli katmanlı YSA

2.3.1. İleri Yayılım (Forward Propagation)

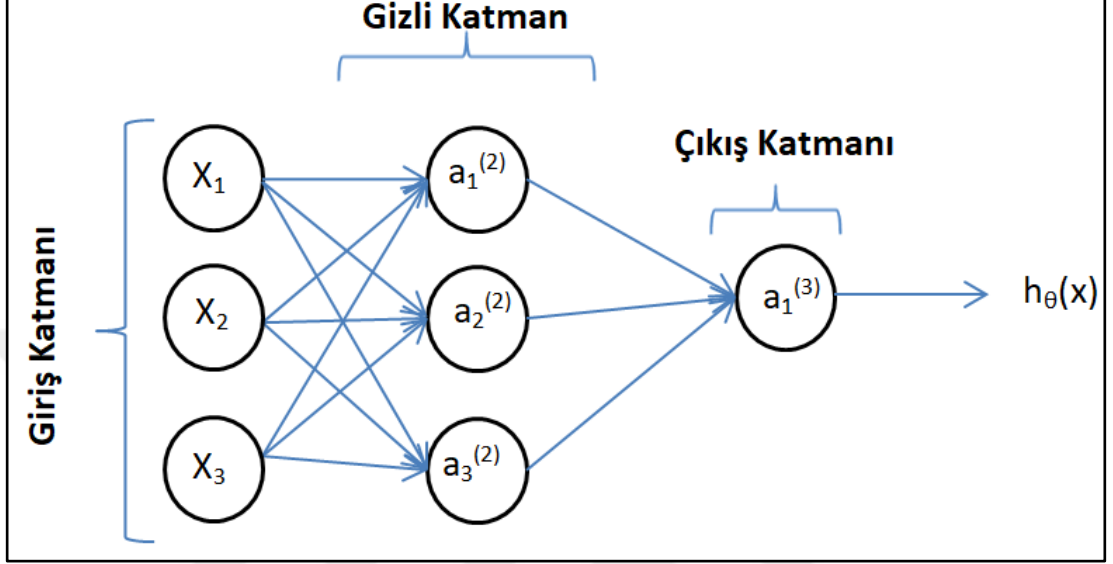
İleri beslemeli bir yapay sinir ağında (Feed Forward Neural Network) çıkış değerinin hesaplanması giriş katmanından çıkışa doğru ilerleyerek gerçekleştirilir. Bir katmanda üretilen değerler kendisine ait ağırlıklarla çarpılarak bir sonraki katmana iletilir. Sıradan YSA yapısında bir katmandaki her bir nöron kendinden sonra gelen katmandaki her bir nörona bir ağırlık değeri ile bağlıdır. Bu bağlantı şekline tam bağlantılı mimari (fully connected architecture) denir. Şekil 2.6.'daki YSA'da üç adet nörona sahip giriş katmanındaki her bir nöron kendisinden sonra gelen gizli katmandaki her bir nörona bir ağırlık ile bağlıdır. Bu nedenle bu iki katman arasında 9 adet (3x3) ağırlık bulunur. Gizli katmandaki her bir nöron için de bir adet bias değeri olacağından bu iki katman arası toplam 12 adet (9+3) parametreye sahip oluruz. Şekil 2.6.'daki ileri beslemeli YSA için giriş vektörümüzü ve giriş katmanı ile gizli katman arasındaki ağırlık matrisimizi aşağıdaki şekilde yazabiliriz.

$$\theta^{(1)} = \begin{bmatrix} \theta_{10}^{(1)} & \theta_{11}^{(1)} & \theta_{12}^{(1)} & \theta_{13}^{(1)} \\ \theta_{20}^{(1)} & \theta_{21}^{(1)} & \theta_{22}^{(1)} & \theta_{23}^{(1)} \\ \theta_{30}^{(1)} & \theta_{31}^{(1)} & \theta_{32}^{(1)} & \theta_{33}^{(1)} \end{bmatrix}, \quad X = \begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{bmatrix}$$

$\theta^{(l)}$: Ağırlık matrisi (l katmanı ile $l + 1$ katmanı arasındaki ağırlıklar)

X : Giriş vektörü

3 adet girişe sahip olan YSA'nın 4 elemanlı olan giriş vektörünün ilk elemanı (X_0) ve $\theta^{(1)}$ ağırlık matrisinin ilk sütunu gizli katmana ait bias değerlerini üretmek içindir.



Şekil 2.6. İleri beslemeli YSA

YSA mimarisinde aktivasyon değerlerinin hesaplanması girişten çıkışa doğru gerçekleştirilir. Bir katmana ait aktivasyonlar hesaplanırken sırasıyla eşitlik 2.7. ve 2.8.'de görülen işlemler uygulanır. Bir katmana ait ağırlıklar o katmanın aktivasyonları ile çarpılarak bir sonraki katman için toplam fonksiyonu z hesaplanır. Elde edilen toplam fonksiyonuna bir aktivasyon fonksiyonu uygulanarak ilgili katmana ait aktivasyonlar elde edilmiş olur.

$$\theta^{(l)} * X = z^{(l+1)} \quad (2.7.)$$

$$\sigma(z^{(l)}) = a^{(l)} \quad (2.8.)$$

Eşitlik 2.7. ve 2.8.'den faydalanarak şekil 2.6.'daki gizli katmana ait aktivasyonları, $\theta^{(1)}$ ağırlık matrisini X giriş vektörü ile çarpıp bir sigmoid aktivasyon fonksiyonundan geçirerek elde ederiz.

$$z^{(2)} = \theta^{(1)} * X = \begin{bmatrix} (X_0 * \theta_{10}^{(1)} + X_1 * \theta_{11}^{(1)} + X_2 * \theta_{12}^{(1)} + X_3 * \theta_{13}^{(1)}) \\ (X_0 * \theta_{20}^{(1)} + X_1 * \theta_{21}^{(1)} + X_2 * \theta_{22}^{(1)} + X_3 * \theta_{23}^{(1)}) \\ (X_0 * \theta_{30}^{(1)} + X_1 * \theta_{31}^{(1)} + X_2 * \theta_{32}^{(1)} + X_3 * \theta_{33}^{(1)}) \end{bmatrix}$$

$z^{(2)}$: 2.katmana ait toplam matrisi

$$\sigma(z^{(2)}) = a^{(2)} = \begin{bmatrix} a_1^{(2)} \\ a_2^{(2)} \\ a_3^{(2)} \end{bmatrix}$$

$a^{(2)}$: 2.katmana ait aktivasyon vektörü

σ : Sigmoid aktivasyon fonksiyonu

$a_i^{(j)}$: j katmanın i nöronuna ait aktivasyon

Şekil 2.6.'daki YSA için gizli katmana ait aktivasyon değerlerini hesapladıktan sonra $\theta^{(2)}$ ağırlık matrisi bu aktivasyonlar ile çarpılır ve bir sigmoid aktivasyon fonksiyonu uygulanarak YSA çıkışı elde edilir. Burada $a^{(2)}$ vektörü ve $\theta^{(2)}$ matrisine ait ilk elemanlar ($a_0^{(2)}$ ve $\theta_{10}^{(2)}$) çıkış nöronunun bias değerini elde etmek için matrislere ilave edilmişlerdir.

$$\theta^{(2)} = \begin{bmatrix} \theta_{10}^{(2)} & \theta_{11}^{(2)} & \theta_{12}^{(2)} & \theta_{13}^{(2)} \end{bmatrix}, \quad a^{(2)} = \begin{bmatrix} a_0^{(2)} \\ a_1^{(2)} \\ a_2^{(2)} \\ a_3^{(2)} \end{bmatrix}$$

$$z^{(3)} = \theta^{(2)} * a^{(2)} = \left(\theta_{10}^{(2)} * a_0^{(2)} \right) + \left(\theta_{11}^{(2)} * a_1^{(2)} \right) + \left(\theta_{12}^{(2)} * a_2^{(2)} \right) + \left(\theta_{13}^{(2)} * a_3^{(2)} \right)$$

$$h_{\theta}(x) = \sigma(z^{(3)}) = a_1^{(3)}$$

2.3.2. Maliyet Fonksiyonu (Cost Function)

Günümüzde YSA'lar çok kuvvetli öğrenme algoritmalarına sahiptirler. Maliyet fonksiyonu YSA'ların parametrelerinin ayarlanması için gerekli olan temel kavramlardan biridir. Maliyet fonksiyonu YSA'nın ürettiği sonuçlardan memnuniyetimizi ölçmeye ve buna bağlı olarak parametreleri güncellememize olanak sağlar. Eğitim setinde m adet örneğe sahip olan ve k adet sınıf için sınıflandırma yapan (k adet çıkışa sahip) bir YSA için maliyet fonksiyonu $J(\theta)$, eşitlik 2.9.'daki hesaplanabilir.

Çıkış katmanında bir adet nöron bulunan tek sınıflı (single class) bir YSA sınıflandırıcı için, maliyet fonksiyonu, eğitim setindeki her bir örneğin YSA çıkışında oluşturacağı maliyeti hesaplar. Çıkış katmanında birden fazla nöron bulunan çok sınıflı (multi class) YSA'lar için, maliyet fonksiyonu, eğitim setindeki

her bir örneğin YSA çıkışındaki tüm sınıflar için ürettiği maliyeti hesaplar. Her bir örnek için ait olduğu çıkış sınıfına ait bir maliyet değeri olacağı gibi ait olmadığı sınıflara ait de maliyet değerleri üretilecektir. Sezgisel olarak çok sınıflı bir sınıflandırma bir giriş örneğinin ait olduğu sınıfa ait çıkış değeri mümkün olduğunca büyük olmalıdır. Verilen örnek için maliyet değerinin ise ait olduğu sınıfa ait çıkış değeri büyüdükçe küçülmesi beklenir. YSA'nın verilen giriş örneğinin ait olmadığı sınıflar için de çıkışında daha küçük değerler üretmesi beklenirken bu sınıflar için çıkış değeri küçüldükçe maliyet değerinin azalması istenir. Bu durumun sağlanması için eşitlik 2.9.'un ilk kısmında çıkışın doğal logaritması alınırken ikinci kısımda çıkış değeri 1'den çıkarıldıktan sonra doğal logaritması alınır. Eğitim setindeki i numaralı örnek eğer gerçekte k sınıfına ait ise $y_k^{(i)} = 1$ ve $(1 - y_k^{(i)}) = 0$ olacağından ait olduğu sınıf için ürettiği maliyet, eşitlik 2.9.'daki maliyet fonksiyonun ilk kısmı ile hesaplanır. i numaralı örneğin ait olmadığı sınıfların her biri için üreteceği maliyet ise fonksiyonun ikinci kısmı ile hesaplanır. Bu şekilde her bir örneğin tüm sınıflar için ürettiği maliyetler toplanarak örnek sayısına bölünür ve ortalama maliyet bulunur.

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m \sum_{k=1}^K \left[y_k^{(i)} \log((h_\theta(x^{(i)}))_k) + (1 - y_k^{(i)}) \log(1 - (h_\theta(x^{(i)}))_k) \right] \quad (2.9.)$$

m : Eğitim setindeki örnek sayısı

k : YSA çıkış sayısı (sınıf sayısı)

$y_k^{(i)}$: i numaralı örnek k sınıfına ait ise $y_k^{(i)} = 1$ değilse $y_k^{(i)} = 0$

$(h_\theta(x^{(i)}))_k$: i numaralı örneğin YSA'nın k çıkışı için ürettiği değer

$\log$: Doğal logaritma

2.3.3. Geri Yayılım Algoritması (Back Propagation Algorithm)

YSA eğitimi, eğitim seti için maliyet fonksiyonunu minimize eden ağ parametrelerinin bulunmasıdır. YSA'ların eğitiminde kullanılan geri yayılım algoritması maliyet fonksiyonunu minimize etmeye yarayan bir algoritmadır. Geri yayılım algoritmasında elde edilen kısmi türevler ile gradyan iniş (gradient descent) veya daha gelişmiş bir optimizasyon algoritması bir arada kullanılarak maliyet fonksiyonunu minimize eden ağ parametreleri bulunur. Geri yayılım algoritması

YSA'nın eğitim setinde bulunan örnekler için YSA'nın çıkış katmanında ürettiği değerleri olması gereken değerlerle kıyaslayarak çıkış hatasını bulur. Bu çıkış hatasını son katmandan ilk katmana doğru yayarak her bir aktivasyon için kısmi türevleri hesaplar.

Geri yayılım algoritmasının çalışmasında ilk olarak ileri yayılım ile YSA çıkışı hesaplanır. Eğitim setindeki örnekler için YSA çıkışı hesaplandıktan sonra, YSA çıkışında elde edilen değerlerden örneklere ait olan gerçek değerler çıkartılarak çıkış katmanı için hata değerleri 2.10.'daki gibi hesaplanır. Elde edilen bu hata değerleri YSA'nın çıkışından girişine doğru yayılarak kısmi türevler elde edilir.

$$\delta^L = a^{(L)} - y^{(t)} \quad (2.10.)$$

δ^L : Çıkış katmanı için hesaplanan hata değerleri

$a^{(L)}$: Eğitim seti için çıkış katmanın ürettiği aktivasyon değerleri

$y^{(t)}$: Eğitim setinde bulunan gerçek çıkış değerleri

Çıkış katmanından önceki katmanlar için kısmi türev değerleri ise eşitlik 2.11.'deki gibi hesaplanır. Bir l katmanın δ değerinin hesaplanması için ilk olarak l katmanına ait θ ağırlık matrisiyle bir sonraki katmanın δ değeri çarpılır. Elde edilen bu çarpım l katmanının aktivasyon fonksiyonunun türevi ile noktasal çarpılarak l katmanına ait δ değeri bulunmuş olur. Bu işlem sırasıyla çıkış katmanında girişe doğru tüm katmanlar için gerçekleştirilir. Giriş katmanına gelindiğinde ise giriş katmanı için herhangi bir hata değeri hesaplanmaz.

$$\delta^l = (\theta^{(l)})^T \delta^{(l+1)} * g'(z^{(l)}) \quad (2.11.)$$

δ^l : l katmanı için geri yayılmış hata değerleri

g' : Sigmoid aktivasyon fonksiyonunun türevi

$\theta^{(l)}$: l katmanına ait ağırlık matrisi

Tüm katmanlara ait δ değerleri hesaplandıktan sonra, YSA'daki tüm katmanlar ve eğitim setindeki tüm örnekler için kümülatif δ değeri (Δ) eşitlik 2.12.'deki gibi hesaplanır. Algoritmanın başlangıcında $\Delta^{(l)}$ ilk olarak 0 değeri ile başlatılır. Eğitim

setindeki örnekler birer birer YSA üzerinden geçirilerek her bir örneğe ait δ değerleri hesaplanarak bir döngü içerisinde toplanır ve Δ değeri elde edilir.

$$\Delta^{(l)} := \Delta^{(l)} + \delta^{(l+1)}(a^{(l)})^T \quad (2.12.)$$

$\Delta^{(l)}$: Kümülatif δ değerlerini içeren matris

2.4. Konvolüsyonel Sinir Ağı (Convolutional Neural Network, CNN)

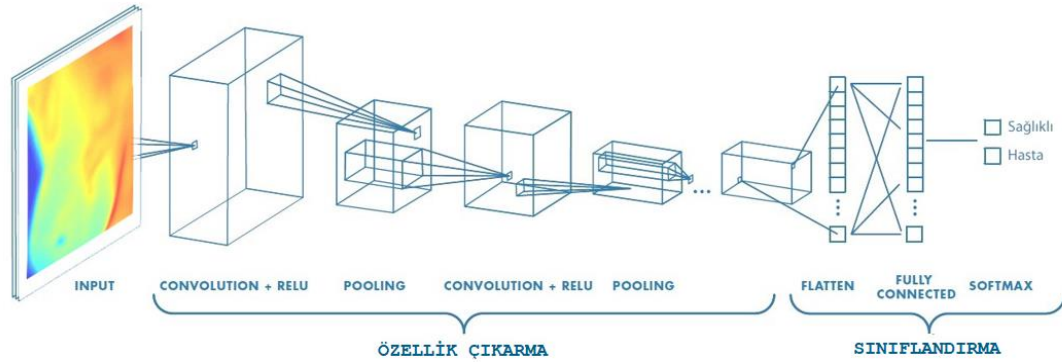
Konvolüsyonel sinir ağı, nöronları arasındaki bağlantı şekli hayvan görsel korteksinden ilham alınmış bir tür ileri beslemeli yapay sinir ağıdır. CNN bir sınıflandırma işlemini gerçekleştirmeyi direk olan görüntülerden, videolardan, metin veya ses dosyalarından öğrenen en popüler derin öğrenme algoritmalarından biridir. Sıradan YSA'lara oldukça benzeyen CNN tıpkı sıradan YSA'lar gibi öğrenilebilen ağırlık ve bias değerlerine sahip nöronlardan oluşur [13, 14]. CNN'in sıradan YSA'lardan en büyük farkı doğası gereği girişlerini iki veya üç boyutlu görüntü olarak varsaymasıdır. Bu durum ağ parametrelerinin sayısında ciddi miktarda azalma sağlarken görüntü işleme problemlerinde aşırı öğrenmenin önüne geçerek verim artışı sağlar.

2.4.1. CNN Mimarisi

Sıradan YSA'lar tek bir vektör şeklinde uygulanan bir girişi bir dizi katman aracılığıyla işleyerek sonucu hesaplarlar. Her bir katman bir dizi nörondan oluşurken bir katmandaki her bir nöron kendinden önce ve sonra gelen katmanlardaki tüm nöronlara belirli ağırlık değerleri ile tamamen bağlıdır (fully-connected). 32x32 boyutunda gri ölçekli görüntülerden oluşan bir veri setimiz olduğunu düşünürsek, bu veri seti için tasarlanmış herhangi bir sınıflandırma işlemini gerçekleştiren YSA'nın giriş vektörü 1024x1 (32x32) şeklinde olacaktır. Tamamen bağlı yapıda bir katmandaki tüm nöronlar bir sonraki katmandaki bütün nöronlara bağlı olacağından örneğimiz için giriş katmanından sonra gelen ilk gizli katmandaki her bir nöron için 1024 adet parametremiz olacaktır. Tamamen bağlı mimarinin doğası gereği YSA derinliği (katman sayısı) ve katmanlardaki nöron sayısının artışı parametre sayısını ciddi miktarlarda artıracak ve hızlı bir şekilde problemi aşırı öğrenmeye (over-

fitting) götürecektir. Yani eğitim setine aşırı bir şekilde adapte olmuş olan YSA eğitim seti haricindeki örnekleri doğru şekilde sınıflandıramayacaktır.

CNN'ler görüntü sınıflandırma problemlerine iyi uyarlanabilen özel mimarisiyle hızlı bir şekilde eğitilebilirler. Sıradan YSA'ların aksine CNN'lerin nöronları üç boyutlu olarak dizilmiştir. CNN'de de sıradan YSA'larda olduğu gibi giriş katmanındaki her bir nöron gizli katmandaki nöronlara bağlanır fakat sıradan YSA'lardaki tamamen bağlantılı yapının aksine her bir giriş nöronu gizli katmandaki tüm nöronlara bağlanmaz. Bunun yerine gizli katmandaki her bir nöron giriş katmanındaki küçük bir bölgeye (örneğin 5x5) bağlanır. Giriş katmanındaki (giriş görüntüsü) bu bölgeye lokal reseptif alan denir. Giriş görüntüsündeki her bir lokal reseptif alan için gizli katmanda bir nöron bulunur. Her bir lokal reseptif alanın tepkisi ise konvolüsyon işlemi ile hesaplanır. Lokal reseptif alanlar gizli katmandaki nöronların bağlanırken aynı ağırlıkları kullandığından bu ağırlıklara paylaşılmış ağırlıklar (shared weights) da denilir. Paylaşılmış ağırlık mimarisi sayesinde CNN görüntü sınıflandırmada parametre sayısını sıradan YSA'lara göre ciddi miktarda azaltarak oldukça uygun bir çözüm olarak karşımıza çıkar.



Şekil 2.7. Örnek CNN mimarisi

Şekil 2.7.'de [33] görüldüğü gibi CNN bir giriş katmanı ile başlayıp tam bağlantılı bir katmanla (fully connected layer) son bulan sıralı bir modeldir. Giriş katmanı, giriş görüntülerinin ham piksel değerlerini tutar. Giriş katmanının boyutları, bir CNN'de kullanılacak veri kümesine göre belirlenir. Tipik bir konvolüsyonel sinir ağı giriş katmanı ile tam bağlantılı katman arasında farklı sayılarda ve kombinasyonlarda tekrarlanan üç tip katmana sahiptir. Bu katmanlar sırasıyla konvolüsyon katmanı,

ReLU katmanı ve Pooling katmanıdır (Pooling layer). Bu üç tip katman CNN oluşturulurken defalarca tekrarlanarak ağın derinliği ayarlanır.

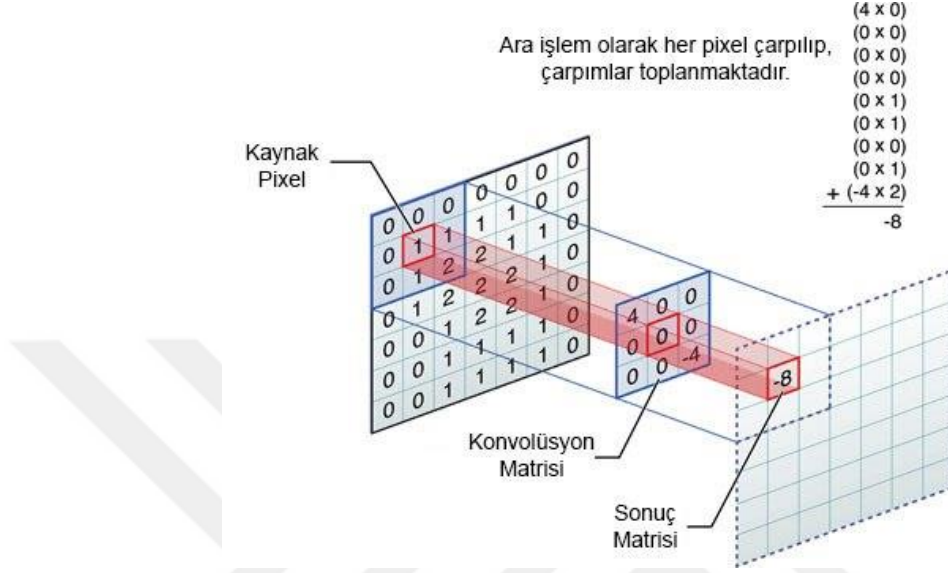
2.4.2. Konvolüsyon Katmanı (Convolutional Layer)

CNN’de bulunan konvolüsyon katmanının amacı girişini sahip olduğu konvolüsyon filtrelerle işleyerek özellik çıkarmaktır. Konvolüsyon katmanının giriş ve çıkışları tensör şeklindedir. Konvolüsyon işlemi bir tür çarpımların toplamı işlemi olup giriş katmanında bulunan reseptif alanların çıkışını hesaplamak için kullanılır. Belirlenecek olan konvolüsyon filtrenin boyutu reseptif alanın boyutunu belirler ve bu filtre giriş görüntüsü üzerinde gezdirilerek her bir reseptif alanın çıkışları hesaplanır. Ağ tasarımı esnasında konvolüsyon katmanı için seçilecek üç adet hiper parametre (hyper-parameters) vardır. Bunlar konvolüsyon filtrenin boyutları, konvolüsyon filtrenin giriş görüntüsü üzerinde gezerken ki adım büyüklüğü ve giriş görüntüsüne herhangi bir dolgu yapılıp yapılmayacağıdır. Şekil 2.8.’de [34] verilen bir giriş görüntüsüne herhangi bir dolgulama yapmadan 3x3 bir konvolüsyon filtre ile gerçekleştirilen konvolüsyon işlemi gösterilmiştir.

Şekil 2.8.’de görüldüğü gibi konvolüsyon işlemi giriş görüntüsünün sol üst köşesinden başlayarak gerçekleştirilir. Konvolüsyon filtrenin merkezi filtrenin görüntüden taşmasına izin vermeyen ilk piksele ayarlanacak şekilde konvolüsyon işlemi başlatılır. Bu durumda görüntünün sınırında kalan pikseller konvolüsyon filtrenin merkezine alınarak işlenemeyecektir. Eğer görüntü sınırlarında ilgilenilen önemli detaylar varsa ve bu piksellerin de konvolüsyon işlemine tabi tutulması isteniyorsa giriş görüntüsüne dolgulama (padding) yapılarak bu problemin üstesinden gelinebilir. Konvolüsyon filtrenin merkezi görüntünün sol üst köşesindeki piksel ile karşılaştırıldığında filtrenin diğer elemanlarının karşılığının boşta kalması engellenecek şekilde giriş görüntüsünün çevresi 0 değerli pikseller ile kaplanır ve bu işleme sıfır dolgulama (zero padding) denir.

Konvolüsyon katmanı CNN’in temel yapı taşıdır. Katmanın parametreleri giriş görüntüsü üzerinde reseptif alanları temsil eden belirli boyuta sahip filtrelerin değerleridir. Konvolüsyon filtre değerleri (parametreler) ağ eğitimi sırasında ayarlanarak ağı ilgili sınıflandırma işlemini başarılı şekilde yapması sağlanır.

Konvolüsyon filtre sayısı artırılarak da giriş görüntüsünden çıkarılacak olan özellik sayısı artırılabilir. Konvolüsyon katmanındaki tüm filtreler giriş görüntüsü üzerinde gezdirilerek sonuç matrisi elde edilir.



Şekil 2.8. Konvolüsyon işlemi

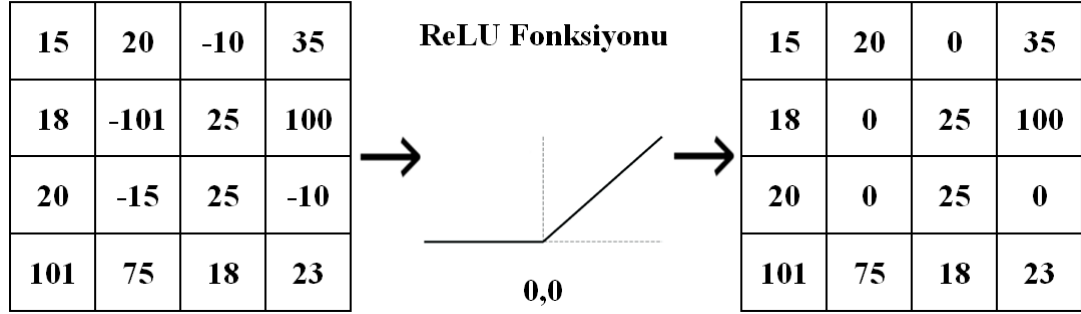
2.4.3. ReLU Katmanı (Rectified Linear Unit Layer)

ReLU katmanı, girişindeki her bir öğeye $f(x) = \max(0, X)$ şeklinde bir aktivasyon fonksiyonunu uygulayan katmandır. Doğrusal olmayan bir aktivasyon fonksiyonu olan ReLU sıfırdan küçük ve eşit olan girişlerini sıfıra eşitlerken sıfırdan büyük girişlerini olduğu gibi bırakır. Şekil 2.7.'de görüldüğü gibi CNN modellerinde konvolüsyon katmanlarından sonra ReLU katmanı kullanılır. Şekil 2.9.'da görüldüğü gibi ReLU katmanı girişinin her bir öğesi için tek tek uygulanıp 0'dan küçük olan değerleri 0'a çekerken 0'dan büyük değerleri olduğu gibi bırakır.

ReLU fonksiyonunun kullanımı diğer aktivasyon fonksiyonlarına göre (sigmoid veya hiperbolik tanjant gibi) genelleme doğruluğunda anlamlı bir fark oluşturmamasına rağmen birkaç kat daha hızlı olması nedeniyle tercih edilir. Bu fark hesaplama yükünün oldukça fazla olduğu derin yapay sinir ağlarında çok büyük uygulama kolaylığı sağlar. Eşitlik 2.13. ve 2.14.' te görüldüğü gibi sigmoid fonksiyona göre türevinin daha basit olması geri yayılım gibi algoritmaları kullanırken çok büyük kolaylık ve hız sağlar.

$$f(x) = \max(0, x) \quad f'(x) = \begin{cases} x > 0 \text{ ise } 1 \\ x \leq 0 \text{ ise } 0 \end{cases} \quad (2.13.)$$

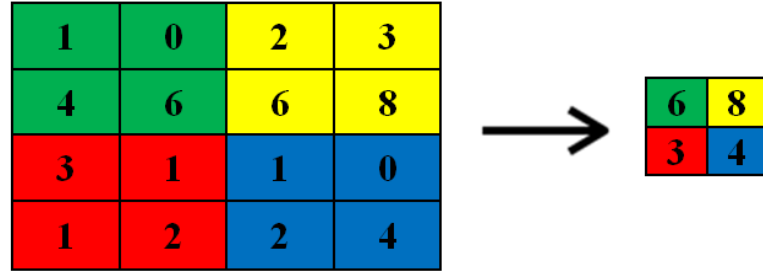
$$\sigma(z) = \frac{1}{1+e^{-z}} \quad \sigma'(z) = \sigma(z)(1 - \sigma(z)) \quad (2.14)$$



Şekil 2.9. ReLU işlemi

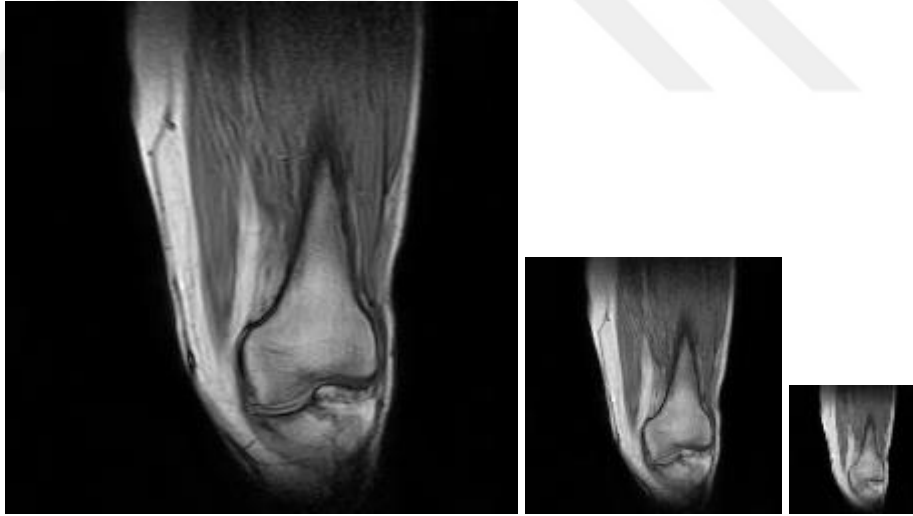
2.4.4. Pooling Katmanı (Pooling Layer)

CNN de bir diğer önemli kavram da doğrusal olmayan (Non-linear) aşağı örnekleme (downsampling) yapan pooling katmanıdır. Pooling katmanı non-parametrik bir katman olup herhangi bir parametre içermez. Pooling işlemi için birkaç farklı fonksiyon kullanılsa da en yaygın olanı max-pooling fonksiyonudur. Bu işlem giriş imgesini çakışmayan dikdörtgenlere ayırır ve her bir alt bölge için o bölgenin en yüksek değerini çıkış olarak üretir. Burada faydalanan prensip bir özellik bir kez bulunduğu diğer özelliklere göre tam lokasyonunun kaba lokasyonu kadar önemli olmadığıdır. Pooling katmanının fonksiyonu parametrelerin miktarı ve hesaplamayı azaltmak için girdinin mekânsal boyutunu kademeli olarak azaltmak ve böylece aşırı öğrenmeyi de kontrol altına almaktır. CNN mimarisinde ardışık konvolüsyon katmanları arasına pooling katmanı koymak yaygın bir uygulamadır. Pooling işlemi bir çeşit çeviriden bağımsızlığı sağlar. Şekil 2.10.'da 2x2 boyutunda bir filtre kullanılarak 2 birimlik kayma adımıyla gerçekleştirilen max-pooling [35] işlemi görülmektedir. 16 piksellik görüntü alanı max-pooling işleminden sonra 4 piksellik bir alana indirgenmiştir.



Şekil 2.10. 2x2 Filtre ile max pooling işlemi

Şekil 2.11.'de 8-bit derinliğinde 256x256 boyutunda bir diz MR görüntüsünün 2x2 boyutunda bir filtre kullanılarak 2 kayma adımıyla max-pooling işleminin 2 kez uygulanmış hali görülmektedir. Görüntü çözünürlüğü ilk filtreleme işleminden sonra 128x128'e ikinci filtrelemeden sonra da 64x64'e düşmektedir. Max-pooling'in en önemli görevi görüntü özelliklerini kabaca koruyarak parametre sayısını azaltıp ezberlemeyi kontrol altına almasıdır. Şekil 2.11.'deki görüntü son haliyle 16 kat küçülmüş olsa da görüntü içerisindeki nesne hala kabaca özelliklerini korur durumdadır.

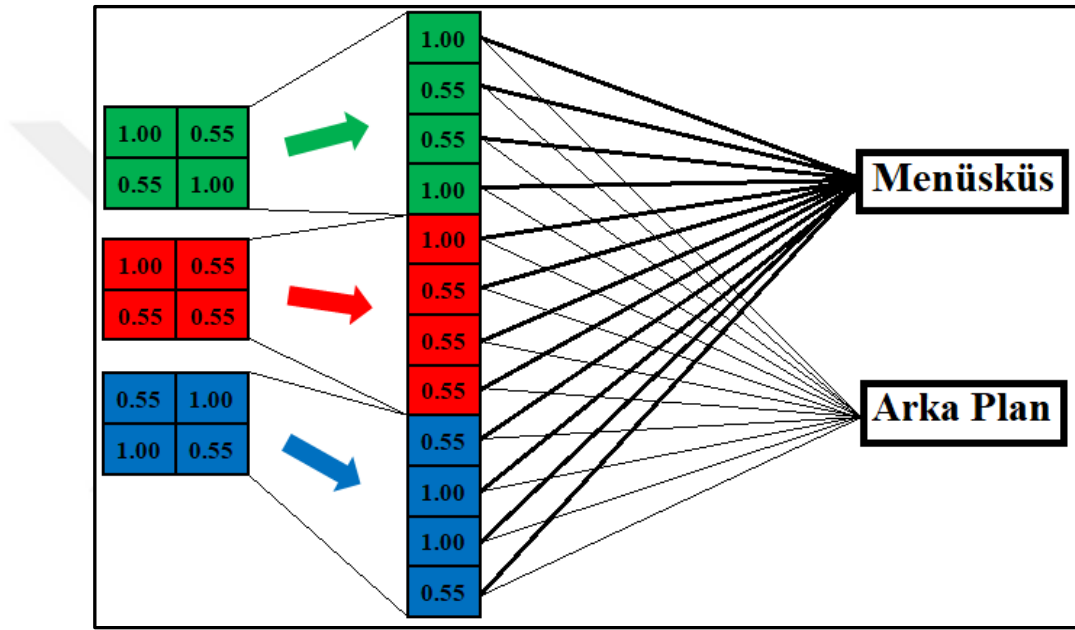


Şekil 2.11. MR görüntülerinde max-pooling işlemi

2.4.5. Tam Bağlantılı Katman (Fully Connected Layer)

Tam bağlantılı katman CNN'de sınıf skorlarını üreten son katmandır. Konvolüsyon ve pooling katmanlarından sonra son olarak yüksek seviye çıkarım tam bağlantılı katmanda gerçekleştirilir. Şekil 2.7.'de görüldüğü gibi CNN girişine bir dizi konvolüsyon ve pooling işlemi uygulandıktan sonra üç boyutlu tensör şeklinde bir çıkış elde edilir. Bu çıkışlar düz bir vektör haline getirilerek tam bağlantılı katmana

aktarılır. Tam bağlantılı katmanın giriş boyutu bir önceki katman tarafından belirlenirken çıkışı veri setindeki sınıf sayısı tarafından belirlenir. Tam bağlantılı katman üzerindeki her bir nöron belirli ağırlık değerleri ile çıkış nöronlarının her birine bağlanır. Bu bağlantı şekli sıradan YSA’lardaki tam bağlantılı mimarinin aynısıdır. Şekil 2.12.’de [36] görüldüğü gibi tam bağlantılı katman bir önceki katmandan gelen girişlerini tek bir vektör şeklinde birleştirir ve kendisine ait ağırlıklarla çarparak sınıflara ait çıkış değerlerini üretir.



Şekil 2.12. Tam bağlantılı katman

2.4.6. Softmax Fonksiyonu

Literatürde bazen ayrı bir katman olarak da karşımıza çıkan softmax fonksiyonu derin öğrenme modellerinin çıkışında sıklıkla kullanılır. Sigmoid fonksiyona benzer bir yapıya ve iyi bir sınıflandırma performansına sahiptir. Softmax fonksiyonu tam bağlantılı katmanda üretilen sınıf skorlarını 0 ile 1 arasındaki olasılık tabanlı değerlere ayarlar. Softmax fonksiyonu eşitlik 2.15. ve 2.16.’da görüldüğü gibi N boyutlu bir giriş vektörünü alarak her bir elemanı 0 ile 1 arası değerlere sahip ikinci bir N boyutlu vektör üretir. Derin öğrenme modellerinin çıkış katmanında genellikle softmax fonksiyonu kullanılsa da destek vektör makinesi (Support Vector Machine, SVM) gibi bir sınıflandırıcı da kullanılabilir. Ekspansiyel bir fonksiyon olması dolayısıyla softmax fonksiyonu sınıflar arasındaki farkı daha da belirginleştirir.

$$S(a_j): \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \rightarrow \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{bmatrix} \quad (2.15.)$$

$$S(a_j) = \frac{e^{a_j}}{\sum_{k=1}^n e^{a_k}} \quad (2.16.)$$

a_j : Softmax fonksiyonu giriş vektörü

$S(a_j)$: Softmax fonksiyonu

2.4.7. CNN Eğitimi

Uygun parametre değerlerine (ağırlıklar ve bias'ler) sahip bir CNN girdi olarak verilen bir görüntüyü çıkışında sahip olduğu sınıflardan bir tanesine doğru şekilde atayacaktır. CNN eğitiminin amacı bir sınıflandırma problemi için doğru sınıflandırma işlemini gerçekleştirecek en uygun CNN parametrelerinin bulunmasıdır. CNN eğitimi gerçekleştirilirken ilk olarak ağırlıklar ve bias'ler rastgele değerlerle başlatılırlar veya öğrenme transferi (transfer learning) dediğimiz benzer görevi gerçekleştiren bir sinir ağının parametre değerleriyle başlatılabilirler. Gerçekte öğrenme transferini bir çeşit tecrübe aktarımına benzetebiliriz. Benzer bir görevi gerçekleştiren uygun bir CNN'den öğrenme transferi ile parametrelerin alınması halinde daha az sayıda veri ve iterasyon ile CNN eğitimi gerçekleştirilebilir. Özellikle derin sinir ağlarının eğitimi için gerekli olan çok fazla veri ihtiyacı ve hesaplama yükleri öğrenme transferi ile azaltılabilir.

Parametre değerlerinin başlangıç değerleri belirlendikten sonra eğitim setindeki her bir örnek CNN'e girdi olarak verilerek çıkışındaki her bir sınıf için ürettiği sınıf skorları (class scores) hesaplanır. Üretilen bu sınıf skorları kısım 2.3.2.'de anlatılan maliyet fonksiyonu gibi bir fonksiyona uygulanarak mevcut parametrelerle CNN'in ne kadar uygun sonuçlar üretilip üretilmediği görülür. Sezgisel olarak eğer parametre değerlerimiz uygun değilse maliyet fonksiyonumuz görece olarak yüksek değerler üretecektir. Aslında CNN eğitimi maliyet fonksiyonunu minimize eden CNN parametrelerini bulma işlemidir.

Literatürde maliyet fonksiyonu (cost function) ve kayıp fonksiyonu (loss function) kavramları sıklıkla karıştırılarak birbirlerinin yerine kullanılmaktadır. Kayıp fonksiyonu eğitim setindeki tek bir örnek için hatayı hesaplarken maliyet fonksiyonu

eğitim setinin tamamı için toplam hatayı hesaplar. Maliyet fonksiyonu literatürde karşımıza bazen de kayıp katmanı (loss layer) olarak çıkabilir.

Maliyet fonksiyonu CNN eğitimi sırasında eğitim setini kullanarak CNN'in o anki parametreleri ile ürettiği sınıf skorlarının olması gereken gerçek değerlerinden ne kadar saptığını hesaplar. CNN parametrelerinin bir sonraki eğitim adımında ne şekilde güncelleneceğinin belirlenmesinde maliyet fonksiyonu dikkate alınır. Literatürde farklı farklı maliyet fonksiyonları bulunmakla beraber kısım 2.3.2.'de anlatılan maliyet fonksiyonu CNN eğitiminde sıklıkla kullanılır.

CNN eğitimi için eğitim seti üzerinde toplam hatayı hesaplayan bir maliyet fonksiyonuna sahip olduktan sonra sıradaki adım bu maliyet fonksiyonunu minimize edecek CNN parametrelerini bulan bir eğitim algoritmasına sahip olmaktır. CNN eğitiminde sıklıkla kullanılan gradyan iniş (gradient descent) algoritması bu optimizasyonu, maliyet fonksiyonunun gradyanının negatifi yönünde belirli bir öğrenme oranı ile ilerleyerek yapar. Kısım 2.3.3.'de anlatılan geri yayılım algoritması ile elde edilen kısmi türevler gradyan iniş algoritması ile kullanılarak maliyet fonksiyonunu minimize eden parametre değerlerine ulaşılır.

Döngüsel bir işlem olan gradyan iniş algoritması eşitlik 2.17.'de görüldüğü gibi uygulanır. Mevcut parametrelerle elde edilen maliyet fonksiyonunun gradyanı belirli bir öğrenme oranı ile çarpılarak parametrelerden çıkartılır ve bu şekilde yeni parametre değerleri elde edilmiş olur. Elde edilen bu yeni parametre değerleri ile algoritmaya devam edilir. Parametrelerin her güncellenişi ile maliyet fonksiyonun kısmi türevleri daha da küçüleceğinden şekil 2.13.'te görüldüğü gibi öğrenme adımları da küçülür. Şekil 2.13.'te görülen maliyet fonksiyonunun minimum noktasına ulaşıldığında maliyet fonksiyonunun türevi 0'a eşit olacağından eşitlik 2.17.'de görülen gradyan iniş algoritmasında parametrelerin güncellenmiş değeri mevcut değerine eşit olur ve algoritma sonlandırılır. Burada öğrenme oranı α eğer çok büyük bir değer seçilirse öğrenme adımı çok büyüyeceğinden gradyan iniş algoritması minimum noktasını sürekli olarak ıskalayabilir. Eğer öğrenme oranı çok küçük bir değer seçilirse bu kez de öğrenme adımı oldukça küçüleceğinden minimuma ulaşma süresi bir hayli artacaktır. Bu yüzden α öğrenme oranının optimum bir değerde seçilmesi gerekir.

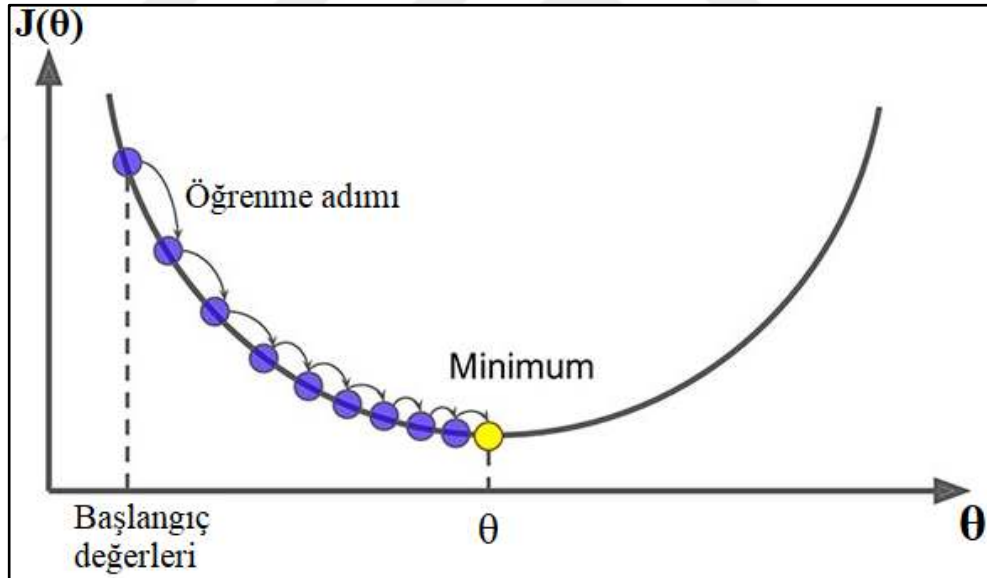
$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1) \quad (2.17.)$$

l : İterasyon sayısı

α : Öğrenme oranı

$J(\theta)$: Maliyet fonksiyonu

Sıradan bir derin öğrenme uygulamasında kullanılan eğitim seti binlerce veriden oluşabilir. Böyle bir uygulamada tüm eğitim seti üzerinden toplam hatayı hesaplayarak parametreleri güncellemek ağır bir hesaplama yükü getirir. Eğitim setini daha küçük yığınlar (mini-batch) bölerek her bir yığın üzerinden gradyan iniş algoritması ile parametreleri güncellemek derin öğrenmede sıklıkla kullanılan bir yaklaşımdır [13]. Algoritmanın bir tek yığın üzerinden parametreleri güncellenmesi işlemine bir iterasyon denirken tüm yığınların yani eğitim setinin tamamının algoritma üzerinden bir sefer geçmesine bir epoch adı verilir.

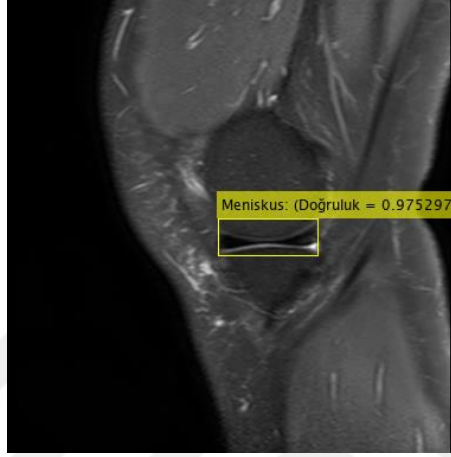


Şekil 2.13. Gradyan iniş

2.5. Bölgesel Tabanlı CNN (Regions with CNN, R-CNN)

Sıradan YSA'lar ve CNN sınıflandırma işlemlerini gerçekleştiren modellerdir. CNN'in gerçekleştirdiği görüntü sınıflandırma işlemi bir görüntü içerisinde nesne tespit işlemi için de kullanılabilir. Bu amaçla Girshick ve arkadaşları bölge önerilerini CNN özellikleri ile kombine ederek verilen bir görüntü içerisinde nesne

tespitini gerçekleştirdiler [17]. Önerdikleri bu metoda CNN özellikleri ile bölgeler (Regions with CNN features, R-CNN) adını verdiler. R-CNN'in amacı bir giriş görüntüsünde aranan nesnenin görüntü içerisindeki lokasyonunu bulmaktır. R-CNN'in çıkışları ise şekil 2.14.'te görüldüğü gibi sınırlayıcı kutular (bounding box) ve bu kutuların ilgili nesne sınıflarına ait olma olasılıklarıdır.



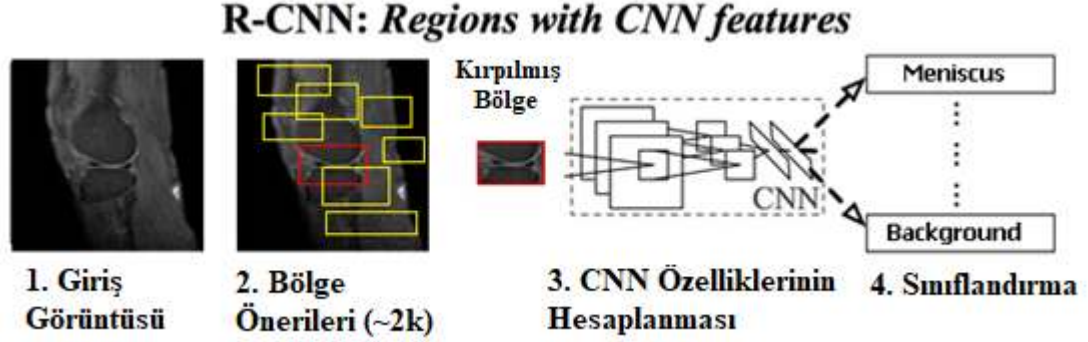
Şekil 2.14. R-CNN ile elde edilmiş çıkış görüntüsü

Şekil 2.15'te R-CNN'in dört bölümden oluşan mimari yapısı görülmektedir. Aşağıda nesne tespitinde R-CNN'in yürüttüğü işlem adımları sırasıyla anlatılmıştır.

1. İlk olarak daha önce benzer bir sınıflandırma işlemini gerçekleştirmek için eğitilmiş olan bir CNN alınır.
2. Bu model yeni problemde tespit edilecek sınıf sayısına göre tekrar eğitilir. Yani kaç çeşit nesne tespit edilecek ise çıkış sayısı o kadar olacaktır. Çıkış sayısı artacak ise yeni çıkışlar ilave edilir azalacaksa bazı çıkışlar budanır.
3. Her bir görüntü için bölge önerileri çıkartılır (yaklaşık 2000 adet) ve CNN'in giriş boyutlarına getirilir.
4. Bölgeler alındıktan sonra her bir sınıf için nesneler ile arka planı ayıracak bir SVM (Support Vector Machine) eğitilir.
5. Son olarak daha sıkı bounding box'lar elde etmek için bir lineer regresyon modeli eğitilir.

Girshick ve arkadaşları çalışmalarında bölge önerisi için seçici arama (selective search) metodunu kullanmışlardır [17, 37]. Yine de farklı bölge önerisi metotları da

CNN ile kombine edilerek nesne tespiti için kullanılabilir. Aslında R-CNN'in yaptığı nesne tespit problemini bir sınıflandırma problemine indirgemektir.



Şekil 2.15. R-CNN mimarisi

Nesne tespiti için R-CNN ne kadar kullanışlı olsa da bazı kısıtlamaları beraberinde getirmektedir. Aşağıda açıklanan nedenlerden dolayı tasarlanan bir R-CNN modeli hız açısından yavaştır.

1. Seçici arama algoritmasına bağlı olarak her bir resim için 2000 adet bölge önerisi çıkartılır.
2. Çıkartılan her bir bölge için CNN kullanılarak özellik çıkartıldığından N tane görüntümüz olduğunu varsayarsak çıkaracağımız CNN özelliği $N \times 2000$ olacaktır.
3. Tüm R-CNN nesne algılama prosesi üç farklı modelin eğitimi gerektirir.
 - a. Özellik çıkarmak için CNN eğitimi
 - b. Nesne tespiti için doğrusal SVM sınıflandırıcı
 - c. Bounding Box'ı sıkılaştırmak için regresyon modeli

Tüm bu işlem sıralaması R-CNN'in görece olarak yavaş bir model olmasına neden olsa da online olmayan ve çok fazla hız gereksinimi olmayan medikal görüntü işleme problemlerinde R-CNN rahatlıkla kullanılabilir. R-CNN'i daha hızlı hale getirmek için literatürde önerilmiş çeşitli modeller mevcuttur. Bunlardan biri Ross Girshick'in önerdiği Fast R-CNN [38] modelidir. Bu modelin R-CNN'den farkı giriş görüntüsünü direk olarak bölge önerilerine ayırmak yerine giriş görüntüsüne CNN uygulandıktan sonra bölge önerilerine ayırmasıdır. Ren ve arkadaşlarının sunduğu daha hızlı bir model olan Faster R-CNN [39] ise Fast R-CNN'e benzer şekilde giriş

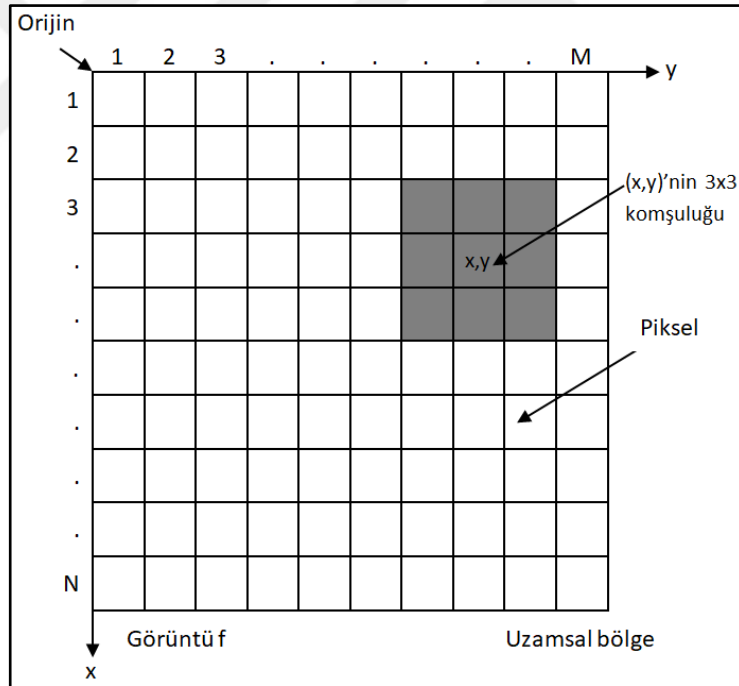
görüntüsüne CNN uygulandıktan sonra bölge önerilerini çıkartır. Faster R-CNN'deki fark ise bölge önerilerini seçici arama ile değil bir bölge önerisi ağı oluşturarak yapmasıdır.



3. MORFOLOJİK GÖRÜNTÜ İŞLEME

3.1. Görüntü İşleme Altyapı

Bir sayısal görüntü $f(x,y)$ şeklinde iki boyutlu bir fonksiyon olarak tanımlanabilir. Burada x ve y uzamsal (düzlemsel) koordinatlardır ve herhangi bir (x, y) koordinat çiftindeki f 'in genliğine görüntünün o noktadaki yoğunluğu veya gri seviyesi denir. Eğer bu tanımdaki x , y ve yoğunluk değerlerinin tamamı sonlu ve ayrık ise bu görüntüye sayısal görüntü denir. Bilgisayar biliminde sayısal görüntü işleme, sayısal bilgisayarlar ve bir algoritma kullanılarak sayısal görüntülerin işlenmesidir. Tanımda da belirtildiği gibi bir sayısal görüntü belirli bir lokasyonu ve değeri olan sonlu sayıda elemandan oluşur ve bu elemanlar piksel olarak adlandırılır [40, 41]. M sütun ve N satırdan oluşan bir sayısal görüntüde $M \times N$ adet piksel bulunur.



Şekil 3.1. $N \times M$ sayısal görüntü

Uzamsal bölge (spatial domain) terimi görüntü düzleminin kendisini ifade eder ve bu kategoride uygulanan görüntü işleme metotları piksellerin direk olarak manipülasyonu şeklinde gerçekleştirilir. Uzamsal görüntü işleme (spatial processing) yoğunluk transformasyonu (intensity transformation) ve uzamsal filtreleme (spatial filtering) olarak iki ana başlığa ayrılabilir. Yoğunluk transformasyonu tekli pikseller

üzerinde işlem yaparak kontrast manipölasyonu ve eşikleme (thresholding) gibi işlemleri gerçekleştirir. Uzamsal filtreleme ise her bir pikselin belirli bir komşuluğu ile çalışarak görüntü keskinleştirme gibi işlemleri gerçekleştirir.

Morfoloji kelimesi genellikle hayvanların ve bitkilerin şekil ve yapısıyla ilgilenen biyolojinin branşını işaret eder. Matematiksel morfoloji veya basitçe morfoloji uzamsal yapıları analiz eden bir teori olarak tanımlanabilir [42]. Matematiksel morfoloji sadece bir teori olmayıp güçlü bir görüntü analiz tekniğidir. Morfoloji olarak tanımlanmasının nedeni nesneleri şekil ve biçim yönünden analiz etmesidir ve analiz yöntemi küme teorisine dayandığı için matematiksel bir yöntemdir.

Genellikle karmaşık olan görüntü işleme problemlerinin çözümünde temel morfolojik işlemler tek başına değil bir dizi işlem kombinasyonu şeklinde uygulanır. Bu bölümde temel morfolojik işlemler ve bu işlemlerin kombinasyonu ile gerçekleştirilen sınır temizleme ve morfolojik rekonstrüksiyon gibi bazı sık kullanılan işlemler anlatılacaktır. Morfolojik işlemlere girmeden önce konunun anlaşılması için bir altyapı sağlayacak şekilde görüntü çeşitleri ve pikseller arasındaki bazı temel ilişkilerden bahsedilecektir.

3.2. Görüntü Çeşitleri

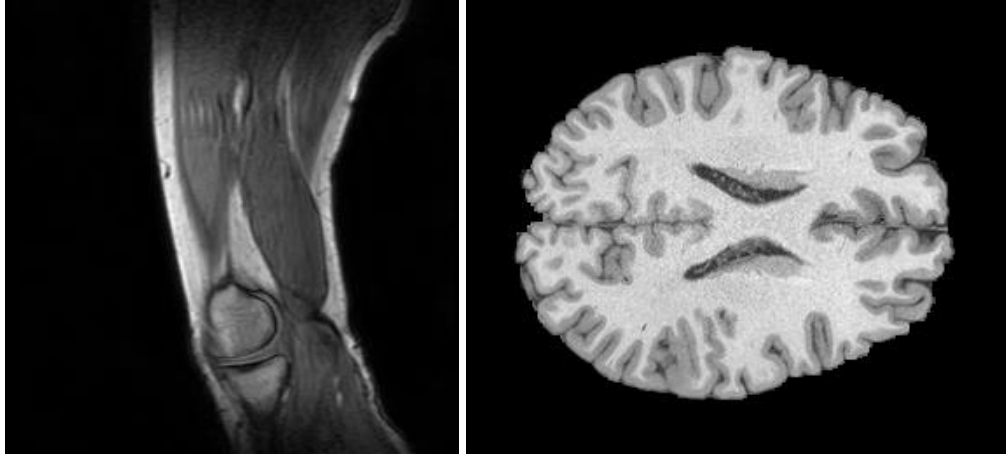
Bu çalışmada 8-bit derinliğe sahip gri ölçekli MR görüntüleri kullanılmıştır. Menüsküs bölgesinin tespiti direk olarak gri ölçekli görüntüler üzerinde yapılmış olup morfolojik işlemlerin uygulanması gri ölçekli olan bu görüntülerin binary'e çevrilmesinden sonra gerçekleştirilmiştir. Ayrıca bazı binary morfolojik işlemlerin uygulanması da üç boyutlu olarak gerçekleştirilmiştir. Medikal görüntülemelerde çeşitli alanlarda renkli görüntülerin kullanımı da mevcuttur. Uygulanacak olan görüntü işleme yöntemlerine bağlı olarak istenildiğinde çeşitli yöntemlerle görüntü tiplerinin birbirlerine dönüşümü sağlanabilir. Renkli görüntüler gri ölçekli görüntülere, gri ölçekli görüntülerde binary görüntülere çevrilerek üzerlerinde çeşitli görüntü işleme yöntemleri uygulanabilir. Ayrıca aynı oryantasyonda belirli bir adım ile ilerlenerek elde edilmiş olan görüntü dizileri birleştirilerek üç boyutlu görüntüler elde edilebilir. Bu şekilde elde edilmiş olan MR görüntüleri üç boyutlu görüntülerdir. Bu kısmın

devamında Gri ölçekli, Binary, Renkli ve Üç boyutlu (3D) görüntüler hakkında bilgi verilecektir.

3.2.1. Gri Ölçekli Görüntü (Grey Scale Image)

Gri ölçekli görüntüleme grinin tonlarının kullanıldığı bir görüntüleme çeşididir. Sayısal görüntülemelerde gri tonlarının sunumu için sınırlı sayıda ton kullanılabilir. En çok kullanılanlardan bir tanesi 256 tonlu gri ölçekli görüntü olmakla beraber daha az ve daha çok tonların kullanıldığı görüntüleme metotları da mevcuttur. Görüntüde kullanılan gri tonlarının sayısının kaç bit ile ifade edildiği görüntünün derinliğini verir. 2-bit derinliğinde bir görüntüde gri tonların sunumu için (2^2 , 0-3) dört farklı kombinasyon mevcut iken şekil 3.2.'de görülen 8-bit görüntülerde (2^8 , 0-255) 256 farklı gri tonu sunulabilir.

Gri ölçekli görüntüler medikal görüntülemelerde MR görüntüleme ve BT (Bilgisayarlı Tomografi) gibi görüntüleme teknikleri kullanılarak direk olarak elde edildikleri gibi renkli görüntülerden dönüşüm yoluyla da elde edilirler. Gri ölçekli görüntüler yoğunluk transformasyonu ve uzamsal filtreleme gibi görüntü işleme yöntemleri ile direk olarak işlenebildiği gibi binary görüntüye çevrilerek temel morfolojik işlemlerle de işlenebilirler.

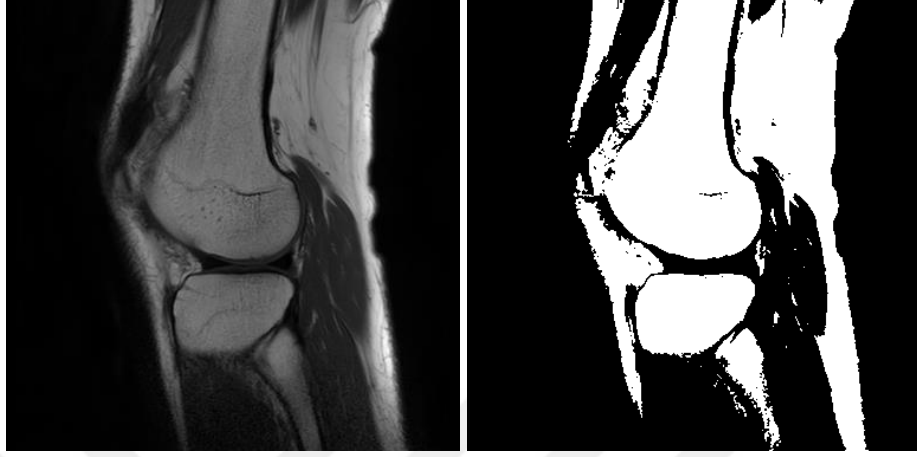


Şekil 3.2. 8-Bit gri ölçekli diz ve beyin MR görüntüsü

3.2.2. Binary Görüntü (Binary Image)

Binary görüntülerdeki piksel değerleri, bir pikselin bir nesneye veya arka plana ait olup olmadığına bağlı olarak 1 veya 0'dır. Nesne sunumu en yaygın olarak

kullanıldığı şekliyle siyah zemin üzerine beyaz nesneler şeklinde veya tam tersi olabilir. Morfolojide görüntü içerisindeki nesneler kümeler olarak kabul edilir.



Şekil 3.3. 8-Bit gri ölçekli görüntü ve T=55 ile eşiklendirilmiş binary görüntüsü

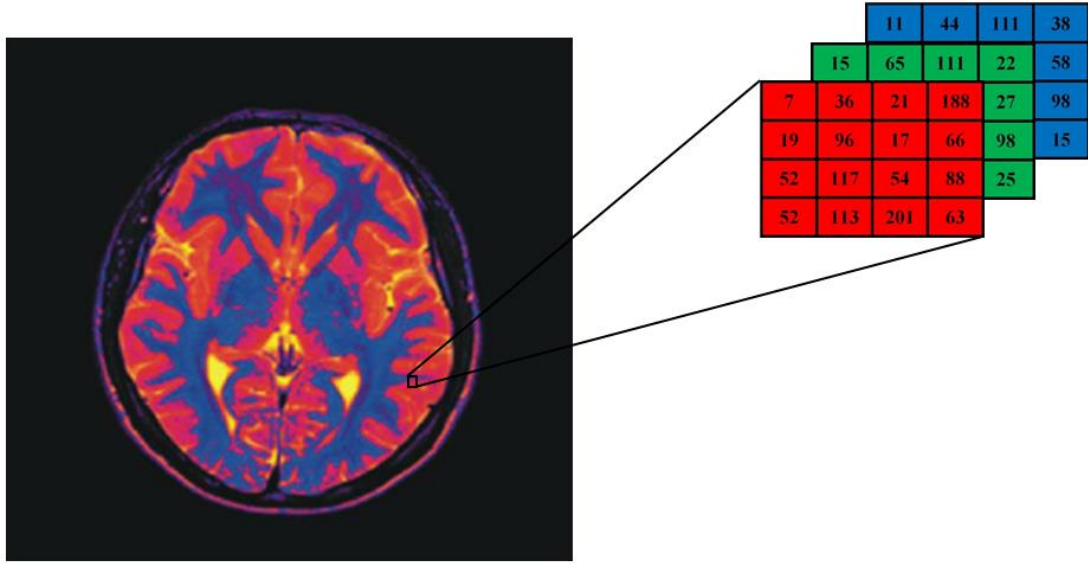
Binary görüntüler sıklıkla gri tonlamalı görüntülerden eşikleme (thresholding) yöntemi ile elde edilirler. Görüntü işlemede en çok kullanılan temel tekniklerden birisi eşiklemedir. En çok tercih edilen siyah arka plan üzerine beyaz nesneler şeklindeki görüntü sunumunda gri ton eşiklemesi eşitlik 3.1.'de görüldüğü gibi uygulanır. Burada nesneleri arka plandan çıkarmak için bir T eşik değeri seçilerek T değerinin üzerindeki yoğunluk değerine sahip olan pikseller 1, T değerinin altındaki yoğunluk değerine sahip olan pikseller 0 olacak şekilde binary görüntü elde edilir. Şekil 3.3.'de 8-bit gri ölçekli bir diz MR görüntüsü ve bu görüntüden T=55 eşik değeri ile elde edilmiş binary görüntü görülmektedir. Görüntü eşiklemede çok çeşitli yöntemler mevcuttur. T'nin bir görüntü boyunca sabit bir değer olarak uygulandığı yöntemlere global eşikleme denir.

$$g(x, y) = \begin{cases} 1 & \text{eğer } f(x, y) > T \\ 0 & \text{eğer } f(x, y) \leq T \end{cases} \quad (3.1.)$$

3.2.3. Renkli Görüntü (Color Image)

Her bir piksel için renk bilgisi de içeren sayısal görüntüye renkli görüntü denir. Bilgisayar ekranlarında genellikle RGB (Red, Green, Blue) renk modeli kullanılsa da HSI (Hue, Saturation, Intensity), CMY (Cyan, Magenta, Yellow) ve CMYK (Cyan,

Magenta, Yellow, Black) gibi diğer modellerde farklı ortamlarda kullanılırlar. RGB renk modelinde her renk kendisini oluşturan kırmızı yeşil ve mavi bileşenleri ile ifade edilir. Gri ölçekli görüntüler iki boyutlu bir matris şeklindeyken RGB görüntüler üç boyutlu bir yapıya sahiptir. RGB görüntüler çeşitli yöntemlerle gri ölçekli görüntülere dönüştürülebilirler. Renkli görüntüler renkli görüntü işleme teknikleri ile direk olarak işlenebildikleri gibi gri ölçekli görüntüye çevrilerek de işlenebilirler.



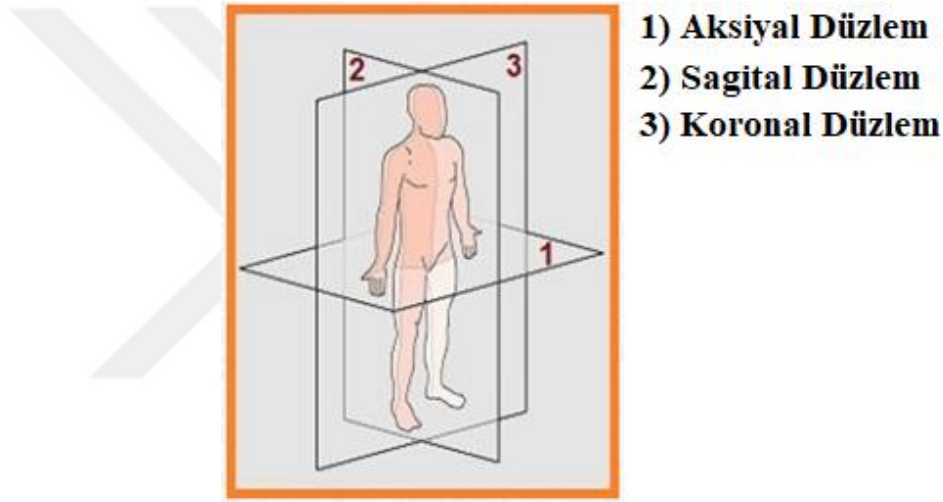
Şekil 3.4. RGB renkli görüntü

3.2.4. Üç Boyutlu görüntü (3D Image)

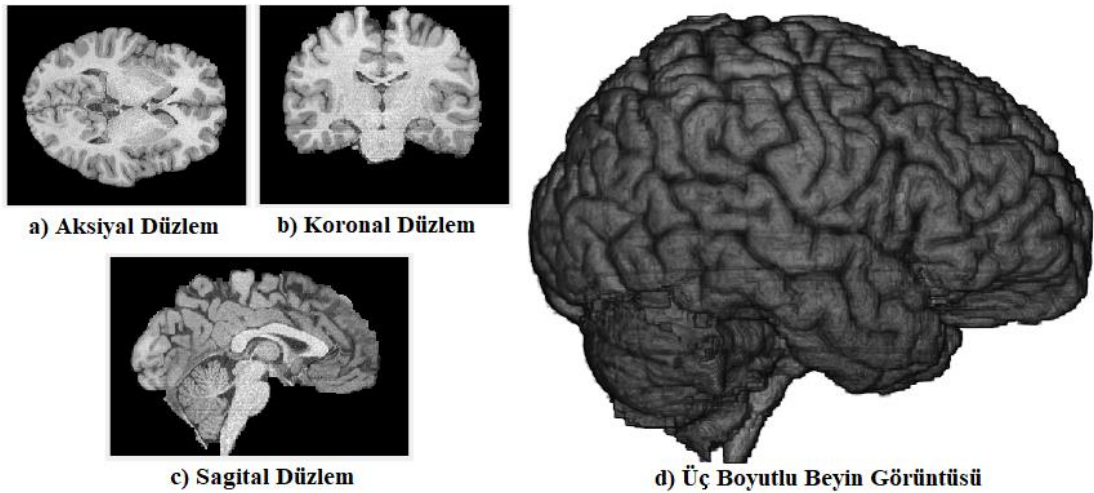
Şekil 3.1.'de görülen X ve Y eksenlerinden oluşan iki boyutlu bir görüntüye ilave olarak bir Z ekseninin gelmesiyle elde edilir. Medikal görüntülemelerde belirli bir dilim aralığı ile aynı oryantasyonda çekilen iki boyutlu görüntülerin bir dizi şeklinde üst üste konulmasıyla üç boyutlu görüntüler elde edilebilir. Medikal görüntüler elde edildikleri düzlemlere göre şekil 3.5.'te görüldüğü gibi adlandırılırlar. Vücudu alt ve üst şeklinde ortadan ikiye ayıran düzleme aksiyal düzlem, vücudu sağ ve sol şeklinde ikiye ayıran düzleme sagittal düzlem, ön ve arka şeklinde ayırana ise koronal düzlem denir.

Şekil 3.6.'da 8-bit derinlikli 311x260 çözünürlüğünde 209 adet aksiyal düzlem beyin MR görüntüsünün [45] üst üste konulmasıyla elde edilmiş üç boyutlu beyin görüntüsü görülmektedir. Görüntülerin bu şekilde birleştirilmesiyle 209x260

çözünürlüğünde 311 adet koronal ve 311x209 çözünürlüğünde 260 adet sagittal düzlem görüntüsü elde edilmiş olur. Üç boyutlu görüntüleme ile yalnızca bir düzlem boyunca elde edilen görüntülerden diğer düzlem görüntüleri de elde edilebilir. Bu şekilde özellikle nesnelerin kenar kısımlarında kalan bir düzlemde fark edilmesi zor olan detaylar diğer düzlemlerden kolaylıkla incelenebilir. Herhangi bir vücut düzleminde elde edilen üç boyutlu görüntünün belirli bir kalitede olabilmesi için o düzleme ait yeterli sayıda görüntü karesinin bulunması gerekir. Bu şekilde şekil 3.6.'daki gibi üç boyutlu görüntü bir düzlemden elde edildikten sonra diğer düzlemlere ait görüntüler makul çözünürlüklerle elde edilebilirler.



Şekil 3.5. Medikal görüntülemde anatomik düzlemler



Şekil 3.6. Aksiyal düzlemden elde edilmiş üç boyutlu beyin görüntüsü

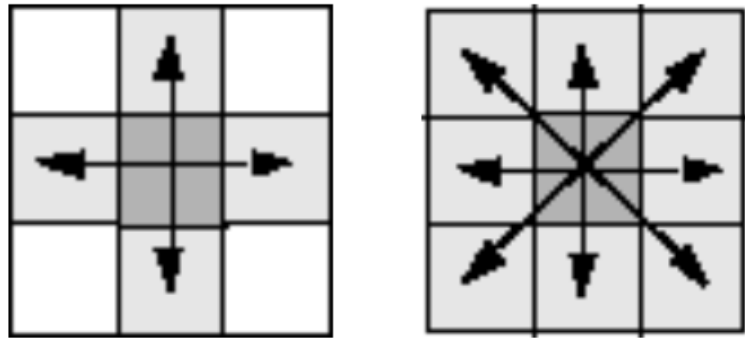
3.3. Pikseller Arası İlişkiler

$f(x,y)$ fonksiyonu şeklinde tanımlanmış olan şekil 3.1.'deki görüntüde (x,y) lokasyonundaki p pikselinin iki adet yatay, iki adet dikey ve dört adet köşegen olmak üzere toplam sekiz adet komşusu bulunmaktadır. P 'nin dikey ve yatay komşuları p 'nin dört komşusu olarak tanımlanır ve $N_4(P)$ şeklinde gösterilir. P 'nin 4 köşegen komşusu ise $N_D(p)$ şeklinde gösterilirken p 'nin iki boyutlu eksendeki tüm komşuları $N_8(p)$ şeklinde gösterilir.

Sayısal görüntülerde bitişiklik kavramı ise komşuluğa ilave olarak piksellerin yoğunluk değerlerini de ilişkilendirmede kullanır. Bitişikliği tanımlamak için bir V yoğunluk değeri kümesi kullanacak olursak binary bir görüntüde 1 değerine sahip piksellerin bitişikliğini tanımlamak için $V=\{1\}$ olacaktır ve 4-Bitişiklik ve 8-Bitişiklik kavramları aşağıdaki şekillerde tanımlanır.

4-Bitişiklik: Eğer bir q pikseli p 'nin 4 komşuluğu içerisinde ($N_4(P)$) ve değeri V 'den ise p ve q pikselleri 4-bitişiktir.

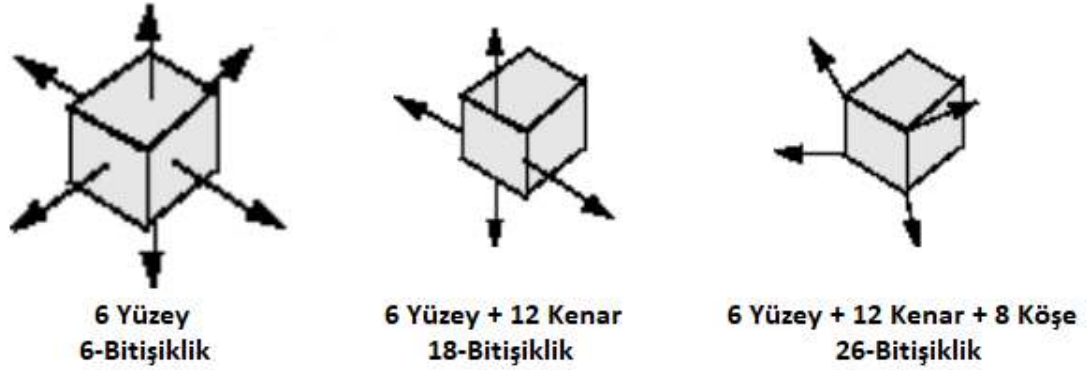
8-Bitişiklik: Eğer bir q pikseli p 'nin 8 komşuluğu içerisinde ($N_8(p)$) ve değeri V 'den ise p ve q pikselleri 8-bitişiktir.



Şekil 3.7. İki boyutlu görüntüde sırasıyla 4 ve 8-bitişiklik

İki boyutlu görüntülere ilave olarak belirli bir derinliğe de sahip olan üç boyutlu görüntülerde daha karmaşık bitişiklik ilişkileri mevcuttur. Üç boyutlu bir görüntü içerisindeki bir pikseli küp şeklinde hayal edecek olursak bu küpün yüzeylerine temas eden ve onunla aynı değerde olan komşuları onun 6-Bitişikliği olacaktır. Küpün yüzeyleri birlikte 12 kenarına da temas eden pikseller $(12+6)$ 18-Bitişikliği

olarak adlandırılırlar. Eğer küpün köşeleri de hesaba katılacak olursa (12+6+8) tüm bu pikseller bir pikselin 26-Bitişikliğini oluşturur. Şekil 3.8.'de gördüğümüz 6, 18 ve 26-Bitişiklik tanımları [46] üç boyutlu görüntülerde morfolojik işlemleri derinlemesine de uygulamamızı sağlar.



Şekil 3.8. Üç boyutlu görüntülerde 6, 18 ve 26-Bitişiklik

3.4. Aşınma (Erosion)

Aşınma ve genişleme işlemleri morfolojik görüntü işlemenin temellerini oluştururlar. Morfolojik algoritmaların çoğu bu iki operasyon temellidir. Aşınma görüntü içerisindeki bir nesne üzerinde daraltma ve inceltme işlemlerini gerçekleştirirken genişleme görüntü içerisindeki nesneleri kalınlaştıran veya büyüten bir işlemdir. Bu kalınlaştırma ve inceltme işlemleri yapılandırma elemanı denilen bir model tarafından kontrol edilir [40].

Z bir tamsayılar kümesi olsun ve Z^2 'nin içerisinde X ve B kümeleri olsun. X 'in B ile aşınması $X \ominus B$ ile gösterilir ve eşitlik 3.2.'deki gibi z kadar öteleme durumunda B 'nin X içerisinde yer alabileceği bütün z noktaları kümesi olarak ifade edilir.

$$X \ominus B = \{z | (B)_z \subseteq X\} \quad (3.2.)$$

Daha açık bir ifade ile anlatacak olursak B yapılandırma elemanının merkezinin X ile çakıştığı durumlarda yapılandırma elemanı arka plan ile herhangi bir eleman paylaşmıyorsa yapılandırma elemanının merkezinin çakıştığı piksel aşındırmadan geriye kalır. Yapılandırma elemanının merkezinin X ile çakıştığı durumlarda

yapılandırma elemanı nesnenin dışarısına taşıp arka plan ile çakışıyorsa merkezinin çakıştığı piksel aşındırılarak nesneden kaldırılır.

0	0	0	0	0	0	0		0	0	0	0	0	0	0	0
0	0	0	0	0	0	0		0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0	0	x	x	x	x	x	x	0
0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0	0	x	x	x	x	x	x	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X								B		X⊖B					

Şekil 3.9. B yapılandırma elemanı ile X'in aşınma işlemi

3.5. Genişleme (Dilation)

Z bir tamsayılar kümesi olsun, X ve B, Z^2 'nin içerisindeki kümeler olsun. B'nin bir yapılandırma elemanı olduğunu ve X'in genişlemeye uğratılacak görüntü kümesi olduğunu farz edersek X'in B ile genişlemesi $X \oplus B$ şeklinde gösterilir ve şu şekilde tanımlanır: Yapılandırma elemanının görüntü kümesi X üzerinde z kadar ötelenirken yapılandırma elemanının merkezinin görüntü ile kesiştiği her bir nokta için yapılandırma elemanının kapsadığı alanların tamamıdır [42].

Şekil 3.10.'da 5 birim uzunluğundaki yatay bir nesnenin 3 birim uzunluğunda dikey bir yapılandırma elemanı ile genişletilmiş hali görülmektedir. Yapılandırma elemanının merkezinin nesne ile her kesiştiği noktada kapladığı alanlar genişletilmiş görüntüde x ile işaretlenmiştir. Morfolojik genişleme nesneleri büyötmek için kullanılır ve belirli noktalardan görüntü içerisindeki nesneleri tekrar inşa eden morfolojik geriatmanın da adımlarından biridir.

Binary morfolojide aşınma ve genişleme işlemleri birbirinin çiftesi (duality) işlemlerdir. Şekil 3.9. ve 3.10.'a baktığımızda görüntü içerisindeki bir nesnenin genişlemesinin arka planın inceltilmesi veya görüntü içerisindeki nesnenin aşındırılmasının arka planın genişletilmesi ile aynı olduğunu görebiliriz.

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

X

1
1
1

B

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	x	x	x	x	x	0
0	1	1	1	1	1	0
0	x	x	x	x	x	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

$X \oplus B$

Şekil 3.10. B yapılandırma elemanı ile X 'in genişleme işlemi

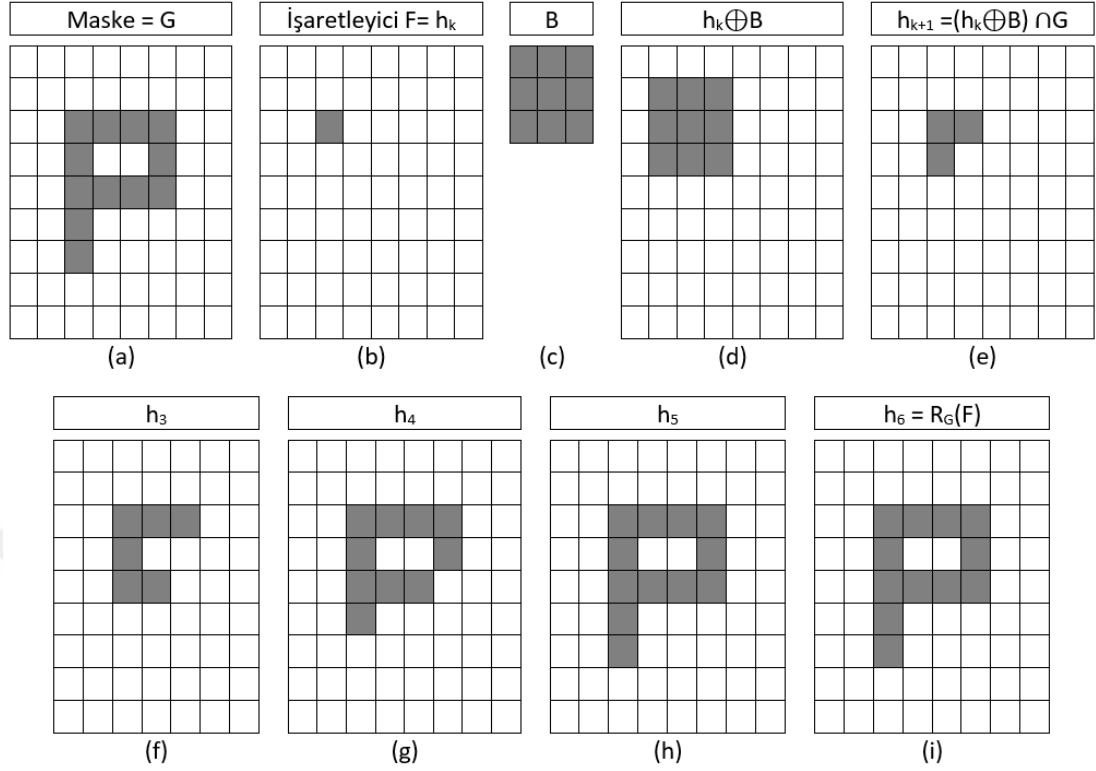
3.6. Morfolojik Geriçatma (Morphological Reconstruction)

Morfolojik geriçatma iki görüntü ve bir yapılandırma elemanını içeren güçlü bir morfolojik dönüşümdür. Morfolojik geriçatma görüntü içerisindeki belirli noktalardan nesneleri tekrar inşa ederek görüntüden çıkarmayı veya etiketlemeyi sağlar. İşaretleyici olarak adlandırılan görüntü dönüşümün başlangıç noktalarını belirlerken maske olarak adlandırılan görüntü dönüşümün sınırlarını belirler. Yapılandırma elemanı ise bitişikliği tanımlamak için kullanılır. Aşağıda binary morfolojik geriçatma için kullanılan bir algoritma verilmiştir [40, 41].

G maskemiz ve F ise işaretleyicimiz olsun. G 'nin F 'den geriçatması $R_G(F)$ ile gösterilir ve aşağıdaki adımlarla ilerler:

1. İşaretleyici F 'den h_1 'i başlat (3.11. (b))
2. Bir B yapılandırma elemanı oluştur (3.11. (c))
3. $h_{k+1} = (h_k \oplus B) \cap G$ (3.11. (d) ve (3.11. (e))
 $h_{k+1} = h_k$ olana kadar üçüncü adımı tekrar et.
4. $h_{k+1} = h_k$ ise $R_G(F) = h_{k+1}$ (3.11. (i))

Şekil 3.11.'de görüldüğü gibi işaretleyici görüntü yapılandırma elemanı ile büyütülür ve maske ile çarpılarak yeni işaretleyici elde edilir. Bu işlem ile elde edilen yeni işaretleyici bir önceki ile aynı olana kadar devam ettirilir. İşaretleyicinin değişmemesi bize dönüşümün sınırlarına ulaştığımızı ve geriçatma işleminin tamamlandığını gösterir.

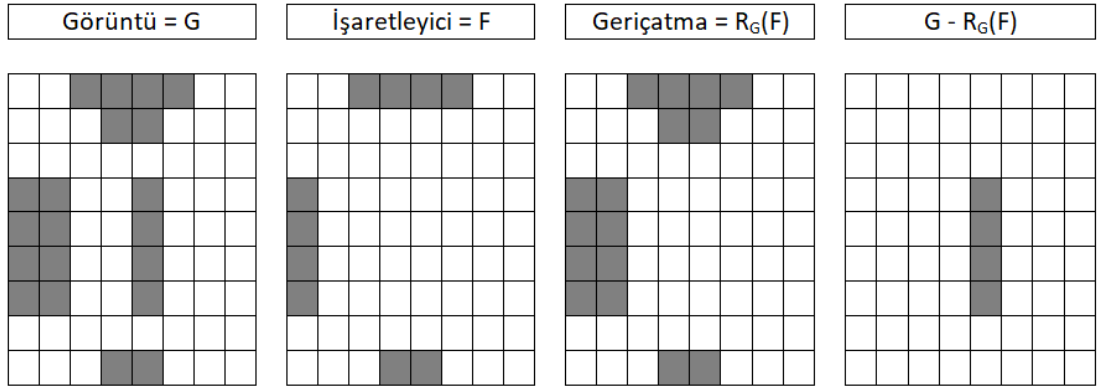


Şekil 3.11. 3x3 yapılandırma elemanı ile morfolojik geriçatma işlemi

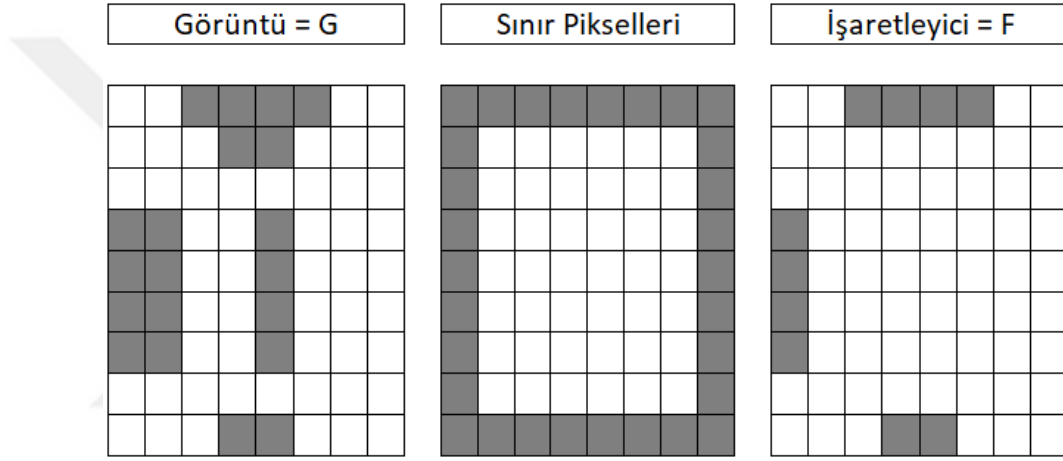
3.7. Sınır Temizleme (Border Clearing)

Birçok uygulamada görüntü sınırına temas eden nesnelerin kaldırılması bir problem olarak karşımıza çıkmaktadır. Bu problem morfolojik geriçatma veya genişleme ile geriçatma (reconstruction by dilation) denilen yöntem kullanılarak aşılabılır. Burada görüntünün kendisi maske, görüntü sınırında bulunan nesnelerin sınıra temas ettiği lokasyonları gösteren görüntü ise işaretleyici olarak kullanılacaktır. Şekil 3.12.'de görüldüğü gibi işaretleyici, görüntü kenarlarına temas eden tüm nesneleri büyütmek için kullanılan bir tohum niteliğindedir. Görüntü sınırına temas eden nesneler bu işaretleyiciden genişleme ile geriçatma kullanılarak büyütülür ve ana görüntüden çıkarılarak temizlenirler [42].

Sınır temizleme için otomatik olarak işaretleyici üretmenin yolu şekil 3.13.'te görüldüğü gibidir. İşaretleyicisi üretilmek istenen görüntü ile aynı boyutlarda olan ve sınır piksellerinin tamamının 1 değerine sahip olduğu bir görüntü, görüntünün kendisi ile noktasal olarak çarpılarak işaretleyici elde edilir. Elde edilen bu işaretleyici yardımı ile de sınır temizleme işlemine başlanılır.



Şekil 3.12. Sınır temizleme işlemi



Şekil 3.13. Sınır temizleme işlemi için işaretleyici üretimi

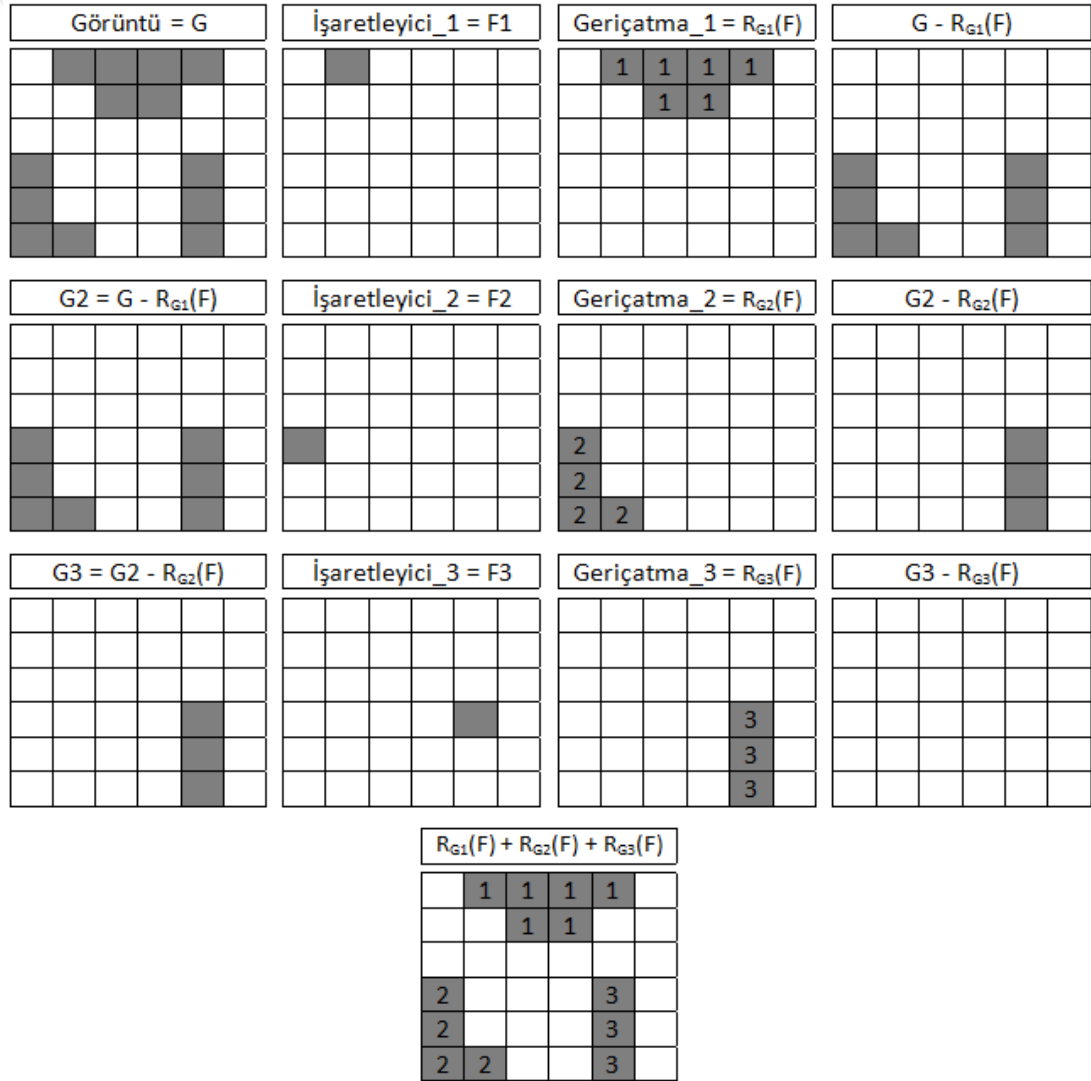
3.8. Bağlı Bileşenler (Connected Components)

Eğer bir küme içerisindeki her bir nokta komşuluk ilişkisi ile birbirine bağlı ise bu küme bağlı bir kümedir. Binary morfolojik görüntü işlemede kısım 3.3.'de bahsettiğimiz bitişiklik tanımlarıyla bağlı olan pikseller kümesine bağlı bileşenler denir. Binary görüntülerde bağlı bileşenlerin her birini ayrı birer nesne olarak düşünürsek bağlı bileşenleri etiketleme nesne özelliklerine göre filtreleme yapabilmek için gerekli ilk adımdır.

Bağlı bileşenleri etiketlemek veya görüntüden çıkarmak için farklı farklı yöntemler kullanılabilir [42, 44]. Morfolojik geriçatma ile bağlı bileşenleri çıkarma, bağlı bileşenleri etiketlemek için kullanılacak yöntemlerden biridir [43]. Morfolojik geriçatma ile bağlı bileşenleri çıkarma işlemine şekil 3.14.'te görüldüğü gibi görüntü içerisindeki ön plan piksellerinden herhangi biri ile başlanır. Pikselin bağlı

bulunduğu nesne morfolojik geriçatma ile büyütülerek etiketlenir ve görüntüden çıkarılır. İşleme görüntüde geriye kalan nesnelerden herhangi birine ait bir ön plan pikseli ile devam edilir. Görüntüdeki tüm ön plan pikselleri etiketlenene kadar bu işleme devam edilir.

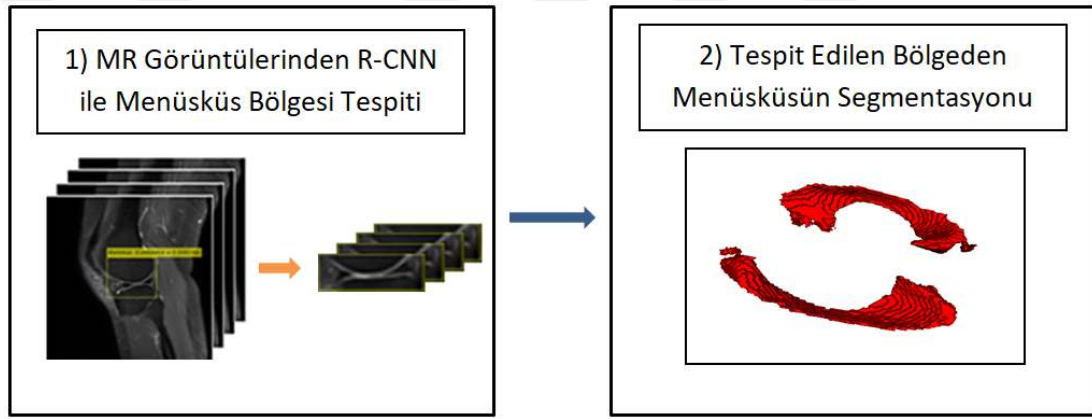
Görüntü içerisindeki bağlı bileşenlerin tamamı çıkartıldıktan sonra her bir bağlı bileşen üzerinde nesne özelliğine göre filtreleme için tek tek işlem yapılabilir. Örneğin bu çalışmada yapacağımız gibi nesneleri yatay aks uzunluğuna göre filtreleme, belirli bir piksel sayısının altında kütleyle sahip olan küçük nesneleri temizleme gibi işlemler nesne tabanlı olarak kolaylıkla uygulanabilir.



Şekil 3.14. Bağlı bileşenlerin çıkarılması

4. MENÜSKÜS SEGMENTASYONU

Bu tez çalışmasında derin öğrenme ve morfolojik görüntü işleme yöntemleri bir arada kullanılarak MR görüntülerinde otomatik olarak menüsküs dokusunun segmentasyonu gerçekleştirilmiştir. Çalışmanın ilk aşamasında PD ağırlıklı MR görüntü dizisinde R-CNN kullanılarak menüsküs bölgesinin tespiti gerçekleştirilmiştir. İkinci aşamada, tespit edilen menüsküs bölgesinden menüsküs dokusunun morfolojik görüntü işleme yöntemleri ile segmentasyonu, PD ve T1 ağırlıklı görüntü dizileri kullanılarak gerçekleştirilmiştir. İki aşamadan oluşan bu çalışmanın ana adımları aşağıda şekil 4.1.'de gösterilmiştir.



Şekil 4.1. Çalışmanın ana aşamaları

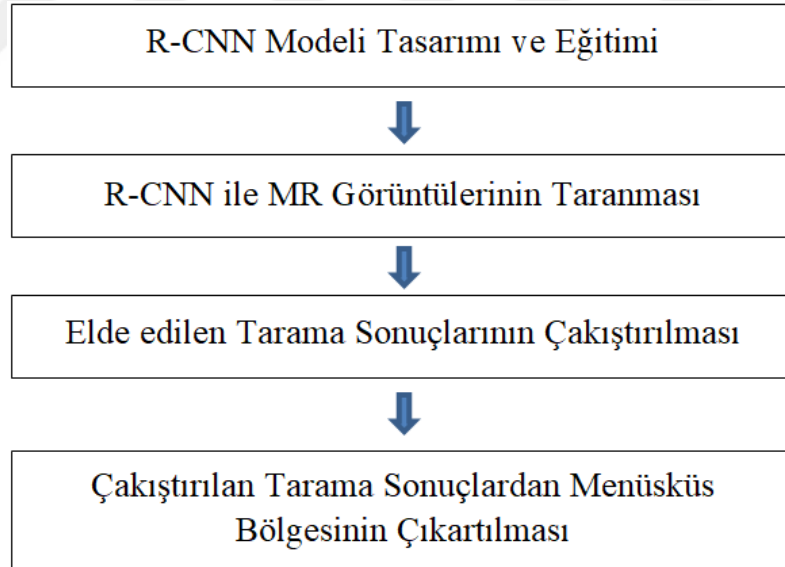
Bu çalışmada sagittal düzlem PDW_SPAIR ve T1W_aTSE diz MR sekansları kullanılmıştır. Tez çalışmasında kullanılan diz MR görüntülerinin tamamı Yozgat Bozok Üniversitesi Araştırma Uygulama Hastanesinden, Yozgat Bozok Üniversitesi Klinik Araştırmalar Etik Kurulunun 21/02/2018 toplantı tarihi ve 2017-KAEK-189_2018.02.21_14 karar numaralı etik kurul onayı ile alınmıştır. Çalışmada kullanılan PDW_SPAIR sekansı 30ms echo time (TE) ve 3000 - 3250ms tekrarlanma zamanı (repetition time (TR)) değerlerine sahiptir. T1W_aTSE sekansı ise 20ms TE ve 450 - 515ms TR değerlerine sahiptir. Her iki sekans için ortak olan parametreler ise şu şekildedir: 1.5 Tesla, 0.18mm x 0.18mm çözünürlük (resolution), 90⁰ kayma açısı (flip angle) ve 1.5mm kesim kalınlığı (slice thickness).

Bu bölümün devamında kısım 4.1.'de MR görüntülerinden derin öğrenme ile menüsküs bölgesinin tespiti anlatılırken kısım 4.2.'de tespit edilen bölgeden

morfolojik görüntü işleme yöntemleri kullanılarak menüsküs dokusunun segmentasyonu anlatılmıştır.

4.1. R-CNN ile Menüsküs Bölge Tespiti

Çalışmanın bu aşamasında PD ağırlıklı MR görüntü dizisinde menüsküs bölgesinin tespiti gerçekleştirilmiştir. Şekil 4.2.'de menüsküs bölgesi tespit işleminin ana adımları görülmektedir. İlk olarak MR görüntülerinde menüsküs bölgesini tespit etmek için bir R-CNN modeli tasarlanmıştır. Tasarlanan R-CNN modelinin eğitimi öğrenme transferi kullanılarak gerçekleştirilmiştir. Sıfırdan bir ağ eğitmek yerine öğrenme transferi kullanımı R-CNN eğitiminin kısıtlı sayıda veri ve donanıyla gerçekleştirilebilmesine olanak sağlamıştır. R-CNN modelinin eğitiminden sonra PD ağırlıklı görüntü dizisindeki her bir görüntü karesinde R-CNN ile menüsküs bölgesi tespiti gerçekleştirilmiştir. Her bir görüntü karesi için elde edilen bu sonuçların karşılaştırılması ile de menüsküs bölgesinin tespiti gerçekleştirilmiştir.



Şekil 4.2. R-CNN ile menüsküs bölgesinin tespit aşamaları

4.1.1. Öğrenme Transferi (Transfer Learning)

Bir makine öğrenmesi tekniği olan öğrenme transferi derin öğrenmede sıklıkla kullanılır. Öğrenme transferi YSA eğitimine başlarken sıfırdan, rastgele ağırlık ve bias değerleri ile başlamak yerine benzer bir sınıflandırma görevini gerçekleştiren bir YSA'nın parametreleri ve mimarisini kullanarak başlamaktır. Öğrenme transferinde

kullanılacak olan YSA'nın çıkış sayısı mevcut sınıflandırma probleminin çıkış sayısından fazla ise çıkış sayılarının fazla kısmı budanır veya daha az çıkışa sahip ise ilave çıkışlar eklenerek mimari uygun hale getirilir. Bu şekilde benzer bir sınıflandırma işlemini yerine getiren bir YSA'nın tecrübesi aktararak bir ince ayar ile yeni sınıflandırma görevlerine adaptasyonu sağlanabilir. Bir ince ayar (fine tuning) işlemi olan öğrenme transferi derin öğrenme algoritmalarında gerekli olan çok sayıda veri ve donanım ihtiyacının üstesinden gelmemizde bize yardımcı olabilir. Bu şekilde elimizde bulunan az sayıda veri ile de derin yapay sinir ağlarının eğitiminde başarı sağlanabilir.

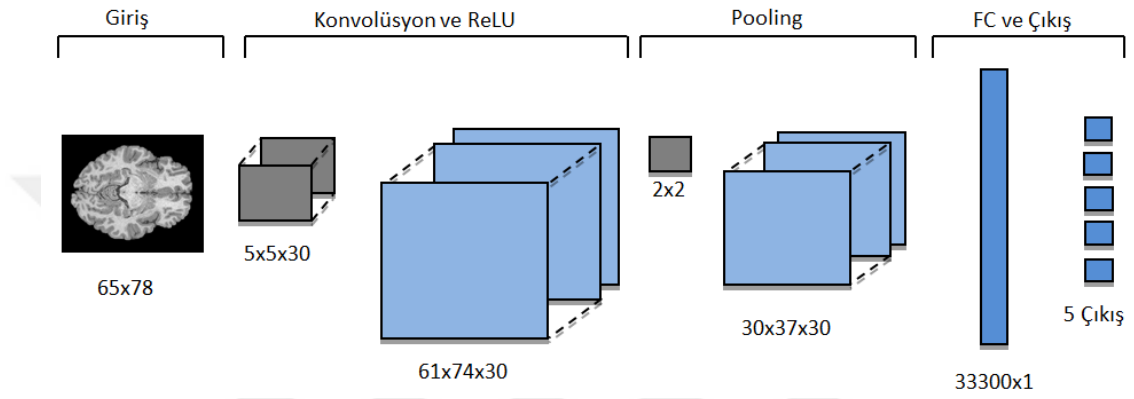
Bu çalışmada PD ağırlıklı MR görüntüleri içerisinde menüsküs bölgesini tespit edecek olan R-CNN'in eğitimi için öğrenme transferi kullanılmıştır. Öğrenme transferinde kullanılan CNN (ConvNetBrain) axial beyin MR görüntülerini sınıflandırmak için çok sayıda veri ile eğitilmiş bir ağıdır. Menüsküs bölgesini tespit eden R-CNN'in eğitimi öğrenme transferi kullanılarak elle çıkartılmış sadece 150 adet menüsküs görüntüsünden 100 adedi ile gerçekleştirilirken 50 adet görüntü de test işlemlerinde kullanılmıştır.

4.1.2. Öğrenme Transferinde Kullanılan CNN (ConvNetBrain)

ConvNetBrain aksiyal beyin MR görüntülerini 5 farklı lokasyona göre sınıflandırabilen bir konvolüsyonel sinir ağıdır. Tablo 4.1.'de görüldüğü gibi ConvNetBrain için kullanılan 1982 adet verinin 985 adedi eğitim için 997 adedi ise test işlemi için kullanılmıştır. ConvNetBrain 997 adet görüntüden oluşan test seti üzerinde 0.982 doğruluğunda bir başarı oranına ulaşmıştır. ConvNetBrain eğitiminde kullanılan MR görüntüleri Human Connectome projesinin [45] veri tabanından sağlanmıştır.

Şekil 4.3.'te mimarisi görülen ConvNetBrain 65x78 piksel boyutunda bir görüntü girişine ve 5 adet sınıflandırma çıkışına sahiptir. ConvNetBrain'in eğitiminde kullanılan Human Connectome veri setindeki beyin MR görüntülerinin orijinal boyutları 260 x 311 piksel çözünürlüğünde olmasına rağmen çok fazla olan hesaplama yükünden kurtulmak adına görüntüler 65x78 boyutlarına sıkıştırılarak kullanılmış ve ağ tasarımı ona göre yapılmıştır. ConvNetBrain birer adet

konvolüsyon, ReLU ve pooling katmanlarına sahiptir. Konvolüsyon katmanında 30 adet 5x5 genişliğinde konvolüsyon filtreye sahiptir. Beyin MR görüntülerinin sınır pikselleri herhangi bir bilgi barındırmadığı için bu görüntülere konvolüsyon işleminden önce herhangi bir kenar dolgulama (padding) işlemi uygulanmamıştır. ConvNetBrain'in pooling katmanında ise 2x2 boyutunda filtre 2 kayma adımı ile max pooling işlemini gerçekleştirmektedir.



Şekil 4.3. R-CNN eğitiminde kullanılan CNN (ConvNetBrain)

Tablo 4.1. ConvNetBrain Hiper parametreleri ve eğitim sonuçları

Eğitim Veri Sayısı	Test Veri Sayısı	Epoch	Mini Batch Boyutu	Öğrenme Oranı	Test Seti Üzerinde Doğruluk
985	997	10	128	0.0001	0.779
985	997	50	128	0.0001	0.788
985	997	100	128	0.0001	0.788
985	997	200	128	0.0001	0.788
985	997	200	128	0.00005	0.981
985	997	200	128	0.00002	0.981
985	997	200	128	0.00001	0.978
985	997	250	128	0.00001	0.979
985	997	300	128	0.00001	0.981
985	997	400	128	0.00001	0.982
985	997	500	128	0.00001	0.982

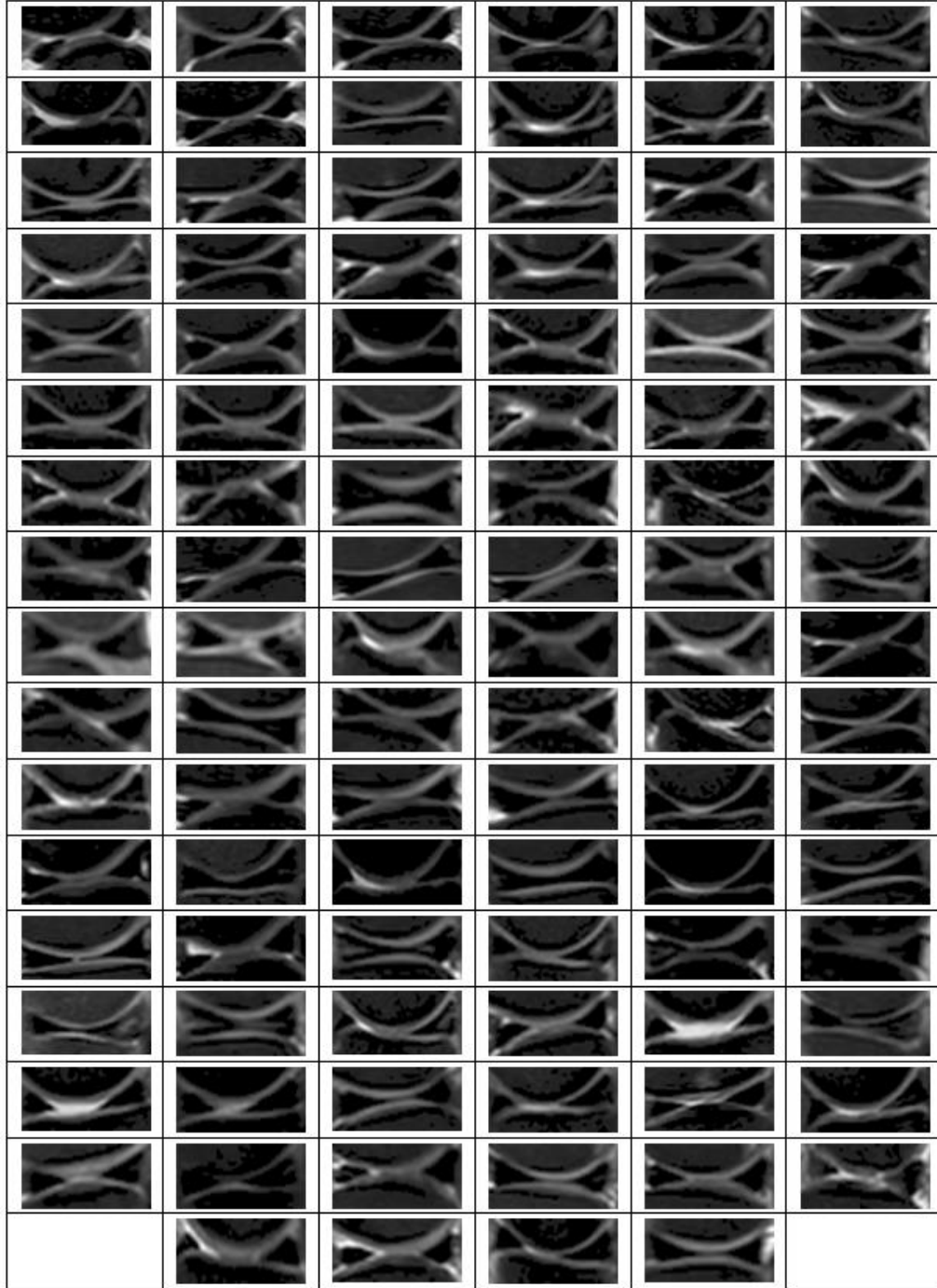
CNN eğitiminde parametre kavramı ağ eğitimi ile öğrenilen ağırlıklar ve bias değerlerini ifade ederken hiper parametre kavramı öğrenme adımı epoch sayısı gibi eğitim sırasında ayarlanan değerleri ifade eder. ConvNetBrain eğitimi tablo 4.1.'de görülen hiper parametreler ile gerçekleştirilmiştir. Eğitim için maliyet fonksiyonu olarak cross-entropy maliyet fonksiyonu kullanılırken optimizasyon algoritması olarak kısım 2.4.7.'de anlatılan mini-batch gradyan iniş algoritması kullanılmıştır. Mini-batch gradyan iniş algoritması için eğitim seti tablo 4.1.'de görüldüğü gibi her biri 128 adet görüntü içeren küçük yığınlar ayrılmıştır. Mini-batch gradyan iniş algoritmasında iterasyon denilen parametrelerin güncellenmesi her bir yığın üzerinden tek tek yapılırken yığınların tamamının optimizasyon algoritması üzerinden geçmesine epoch adı verilir.

Tablo 4.1.'de görüldüğü gibi ConvNetBrain'in eğitimine 0.0001 öğrenme oranı ve 10 epoch sayısı ile başlanarak, test seti üzerinde 0.779 başarı oranı elde edilmiştir. Daha sonra öğrenme oranı 0.0001'de sabit tutularak epoch sayısı sırasıyla 50, 100 ve 200 olacak şekilde artırılmıştır. 0.0001 olan öğrenme oranı ile epoch sayısını artırmanın başarıyı daha fazla artırmadığı görülünce daha küçük bir öğrenme oranı seçilerek CNN eğitimine devam edilmiştir. 200 epoch sayısı ile öğrenme oranı 0.00005'e düşürüldüğünde 0.981 başarı oranına ulaşılmıştır. Öğrenme oranı 0.00005'ten daha küçük bir değer seçilerek devam edildiğinde ise başarı oranının epoch sayısının yetersiz kalması nedeniyle düşüşe geçtiği görülmüştür. Bu nedenle 0.00001 öğrenme oranında epoch sayısı kademeli olarak 500'e kadar artırılmıştır. ConvNetBrain eğitimi 500 epoch sayısı ve 0.00001 öğrenme oranı ile test seti üzerinde 0.982 başarı oranına ulaştığında sonlandırılmıştır.

4.1.3. Bölge Tespiti için R-CNN Modeli Tasarımı ve Eğitimi

Tez çalışmasının bu kısmında PD ağırlıklı MR görüntüleri içerisinde menüsküs bölgesini tespit eden R-CNN modeli tasarlanarak eğitimi gerçekleştirilmiştir. R-CNN tasarımı öğrenme transferi ile kısım 4.1.2.'de anlatılan ConvNetBrain kullanılmıştır. R-CNN eğitime ConvNetBrain mimarisi ve parametreleri ile başlanırken ConvNetBrain'in beş adet olan çıkışı budanarak R-CNN çıkışı menüsküs ve arka plan olmak üzere 2 adede indirilmiştir. Öğrenme transferi ile R-CNN'in az miktarda veri ile eğitilmesi mümkün hale gelmiştir. Bu çalışmada R-CNN eğitimi

iin el ile ıkartılmıř 150 adet mensks blgesi grnts kullanılmıřtır. Bu grntlerin 100 adedi R-CNN eēitiminde kullanılırken 50 adedi test iřleminde kullanılmıřtır. Ařaēıda řekil 4.4.'te PD aēırlıklı MR grntlerinden el ile ıkartılmıř 100 adet grntden oluřan R-CNN eēitim setinin tamamı grlmektedir.



řekil 4.4. R-CNN eēitim setinin tamamı

Aşağıda Tablo 4.2.'de R-CNN eğitime ait hiper parametreler ve başarı oranları gösterilmektedir. R-CNN eğitimi için maliyet fonksiyonu olarak cross-entropy maliyet fonksiyonu kullanılırken parametreleri güncelleyecek olan optimizasyon algoritması için mini-batch gradyan iniş algoritması kullanılmıştır. Her bir mini-batch tablo 4.2.'de görüldüğü gibi 16 adet görüntüden oluşacak şekilde gruplandırılmıştır. Öğrenme transferi gerçekte bir ince ayar (fine-tuning) işlemi olduğundan R-CNN eğitimi 0.000001 gibi oldukça küçük bir öğrenme oranı ile gerçekleştirilmiştir. Öğrenme işlemine rasgele değerler ile başlamadığımız için maliyet fonksiyonu transfer ettiğimiz başlangıç parametreleri sayesinde minimum noktasına daha yakın bir değerde olacaktır. Bu nedenle daha az epoch sayısı ve daha küçük öğrenme adımları ile maliyet fonksiyonunun minimumuna ulaşılabilir.

R-CNN eğitimi 150 adet görüntüden oluşan menüsküs veri setinin tablo 4.2.'de görüldüğü şekliyle eğitim ve test olmak üzere çeşitli oranlarda bölünmesi ile gerçekleştirilmiştir. İlk olarak 50 adet görüntü ile eğitime başlanmış ve test seti üzerinde 0.86 başarı oranına ulaşılmıştır. 100 adet eğitim verisi ile test seti üzerinde 0.93 başarı oranına ulaşıldığında ise R-CNN eğitimi sonlandırılmıştır.

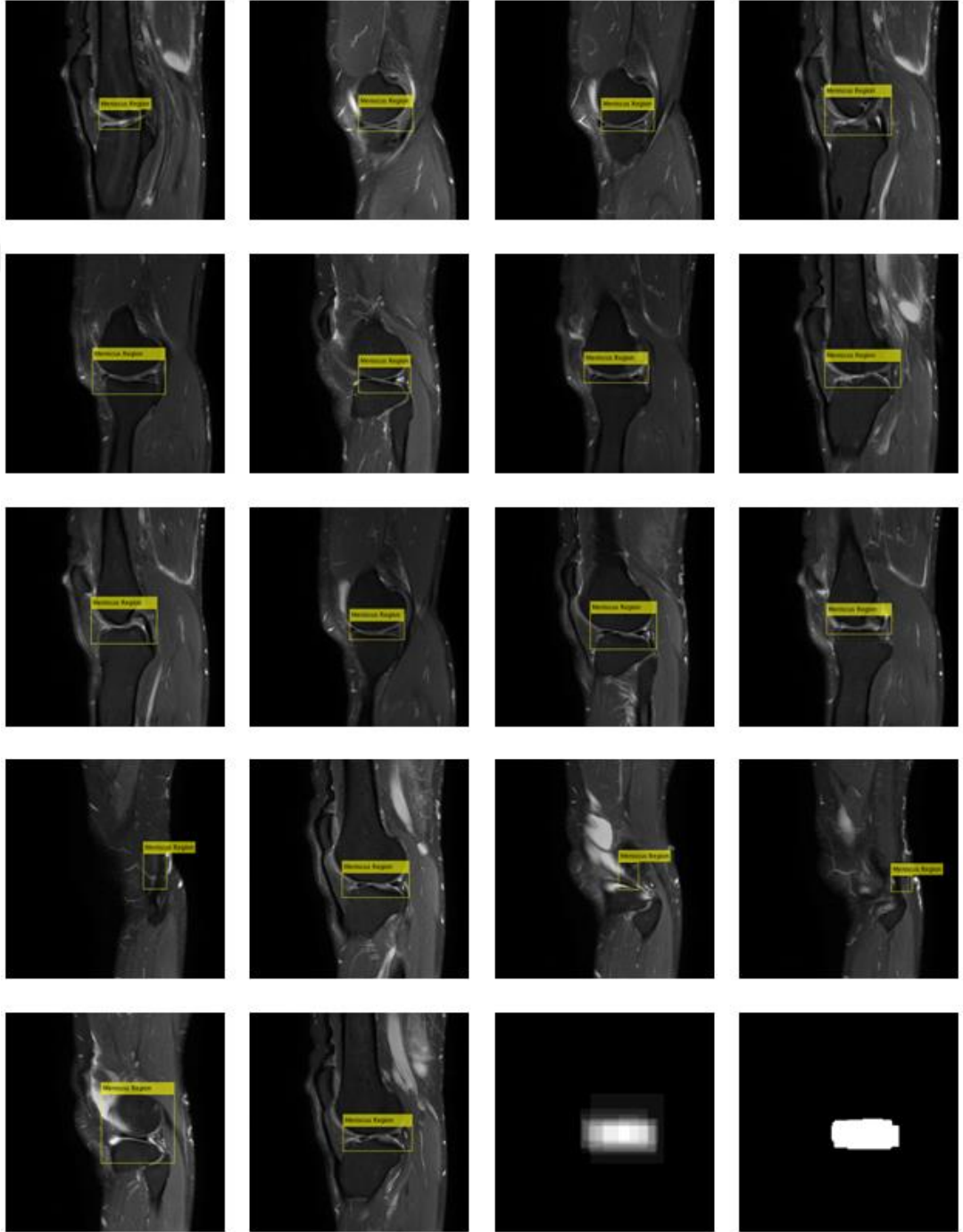
Tablo 4.2. R-CNN eğitimi sonuçları ve Hiper parametreleri

Eğitim Veri Sayısı	Test Veri Sayısı	Epoch	Mini Batch Boyutu	Öğrenme Oranı	Test Seti Üzerinde Doğruluk	Eğitim Seti Üzerinde Doğruluk
50	100	20	16	0.000001	0.860	0.830
75	75	20	16	0.000001	0.907	0.880
100	50	20	16	0.000001	0.930	0.960

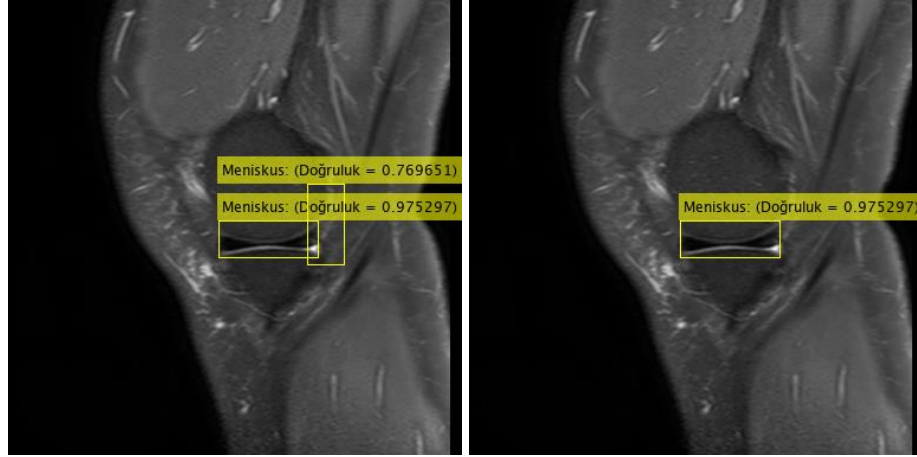
4.1.4. MR Görüntülerinin R-CNN ile Taranması

Şekil 4.5.'te PD ağırlıklı bir MR görüntü dizisi boyunca her bir görüntü karesinin R-CNN ile tarama sonuçları gösterilmiştir. Verilen bir giriş görüntüsünde yaklaşık 2000 adet bölge önerisi çıkaran R-CNN bir görüntü içerisinde bazen birden fazla alanı (genellikle birbirine yakın olan) menüsküs olarak tanımlayabilmektedir. Bu durumda menüsküs olarak tespit edilen bölgelerden en yüksek olasılık değerine sahip olan tutularak diğer sonuçlar elenir (Non-max suppression). Şekil 4.6.'da verilen

görüntüde iki farklı bölge R-CNN tarafından menüsküs bölgesi olarak tespit edilmiş fakat düşük olasılık değerine sahip olan sonuç elenmiştir. Şekil 4.5.'te görülen sonuçlarda maksimum olasılık değerine ait sonuçlar alınarak diğer sonuçlar elenmiştir.



Şekil 4.5. Bir görüntü dizisi boyunca R-CNN tarama sonuçları



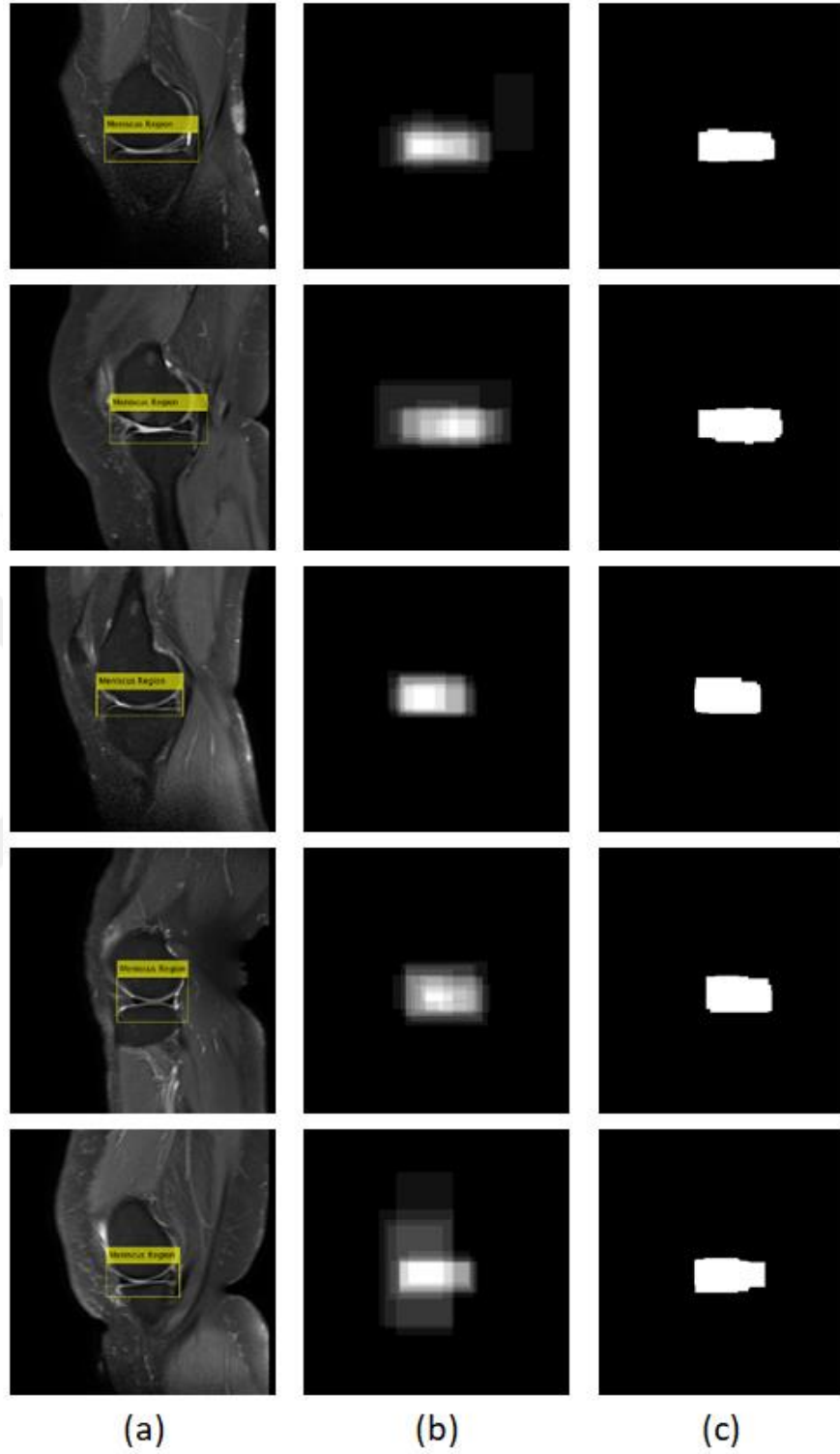
Şekil 4.6. Maksimum olmayan sonuçların bastırılması (Non-max suppression)

4.1.5. Tarama Sonuçlarından Menüsküs Bölgesinin Çıkartılması

MR görüntü dizilerinde değerlendirme yapan bir uzman bir genellikle birden fazla görüntü karesini dikkate alır. Bu çalışmada menüsküs bölgesinin tespiti için bir görüntü dizisi boyunca tüm resimlerde elde edilen tarama sonuçları topluca ele alınarak menüsküs bölgesinin tespiti yoluna gidilmiştir. Bu şekilde herhangi bir görüntü karesinde yanlış olarak tespit edilebilecek bir menüsküs bölgesi sonucunun elimine edilmesi sağlanmıştır. Tarama sonuçlarından menüsküs bölgesinin tespiti için şöyle bir yol izlenmiştir:

- 1) İlk olarak hatalı pozitiflerin (false pozitif) önlenmesi için belirli bir olasılık değerinin altında olan (0.9) menüsküs bölge tespitleri elenir.
- 2) Düşük olasılıklı sonuçlar elendikten sonra geriye kalan sonuçlar bir düzlem boyunca üst üste çakıştırılır.
- 3) Bir görüntü dizisi boyunca bir piksel lokasyonu belirli bir “m” sayısından fazla menüsküs bölgesi olarak tespit edilmişse o piksel lokasyonu menüsküs bölgesi içerisindedir.

Şekil 4.5.’in son satırındaki iki resimde sırasıyla çakıştırılmış sonuçlar ve menüsküs bölgesi olarak tespit edilmiş nihai sonuç görülmektedir. Şekil 4.7.’de ise 5 farklı deneğe ait çakıştırılmış sonuçlar ve menüsküs bölgesi tespit sonuçları gösterilmiştir.



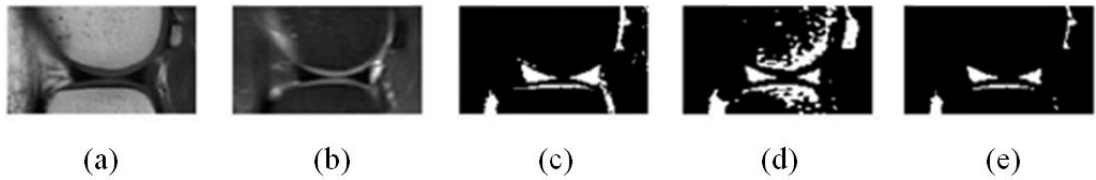
Şekil 4.7. Beş farklı deneğe ait tarama sonuçları

- (a) Her bir görüntü dizisine ait bir adet örnek tarama sonucu.
- (b) Bir görüntü dizisi boyunca elde edilen tüm tarama sonuçlarının çakıştırılması.
- (c) Çakıştırılan sonuçlardan tespit edilmiş menüsküs bölgesi.

4.2. Çıkarılan Menüsküs Bölgesinden Menüsküs Segmentasyonu

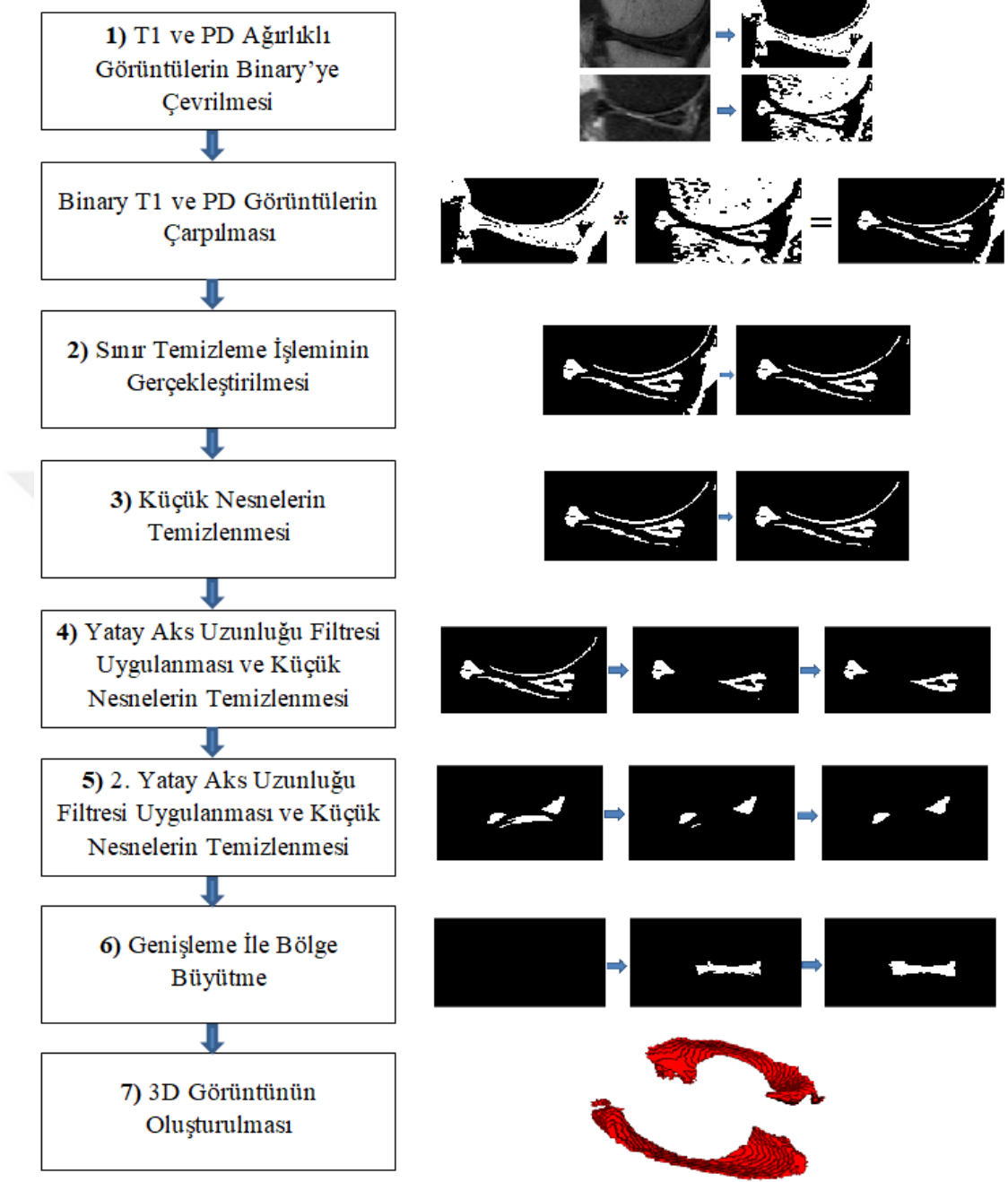
Çalışmanın bu kısmında bir önceki kısımda tespit edilen menüsküs bölgesinden menüsküs dokusunun segmentasyonu gerçekleştirilmiştir. Bu kısımda gerçekleştirilen işlemler akış şeması şeklinde şekil 4.9.'da verilmiştir. Bu iş için morfolojik görüntü işleme teknikleri ve iki farklı MR görüntü dizisi bir arada kullanılmıştır. Menüsküs dokusunu çevreleyen dokuların farklı kontrast sergilediği iki farklı görüntü dizisi, menüsküs dokusunun kendisini çevreleyen dokulardan ayrıştırılmasını sağlamak amacıyla kullanılmıştır. Nesneleri şekilsel olarak analiz edebilen oldukça kuvvetli bir görüntü işleme tekniği [42] olan morfoloji ise istenmeyen nesneleri görüntü içerisinden filtreleyerek geriye ilgilenilen nesneyi (object of interest) bırakmak için kullanılmıştır.

T1 ve PD ağırlıklı görüntü dizilerinde menüsküs dokusu benzer gri ölçek yoğunluk değerlerine sahiptir. Menüsküs dokusunu çevreleyen dokular ise her iki görüntü dizisinde de farklı gri ölçek yoğunluk değerleri sergilemektedir. Bu özellikten faydalananarak T1 ve PD ağırlıklı görüntüler uygun bir eşik değer ile binary görüntüye çevrildikten sonra birbirleri ile noktasal olarak çarpılırlarsa şekil 4.8.(e)'de görüldüğü gibi menüsküs dokusunun çevresindeki dokulardan ayrıştırıldığı binary görüntü elde edilmiş olur. Bu işlemin sonunda elde edilen sonuç görüntüsünde çevresinden ayrılmış menüsküs dokusu ile birlikte bazı istenmeyen doku parçacıkları kalır. Bu noktada morfolojik işlemler bizim bu istenmeyen dokuları görüntüden filtre ederek çıkarmamızı sağlarlar.



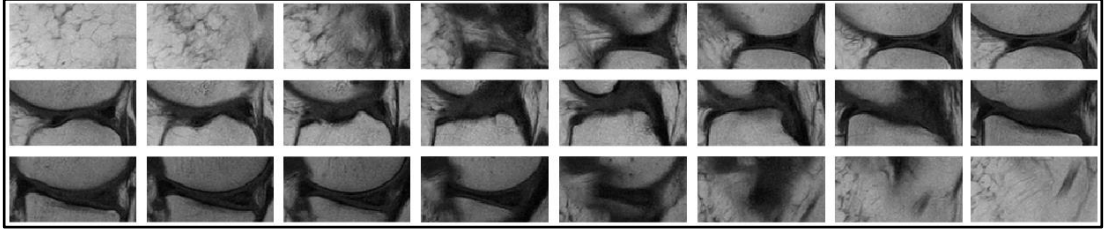
Şekil 4.8. Binary T1 ve PD ağırlıklı görüntülerin çarpımı

- (a) T1 Ağırlıklı Görüntü
- (b) PD Ağırlıklı Görüntü
- (c) Binary T1 Ağırlıklı Görüntü ($T1 < 35$)
- (d) Binary PD Ağırlıklı Görüntü ($PD < 35$)
- (e) $(c) * (d)$

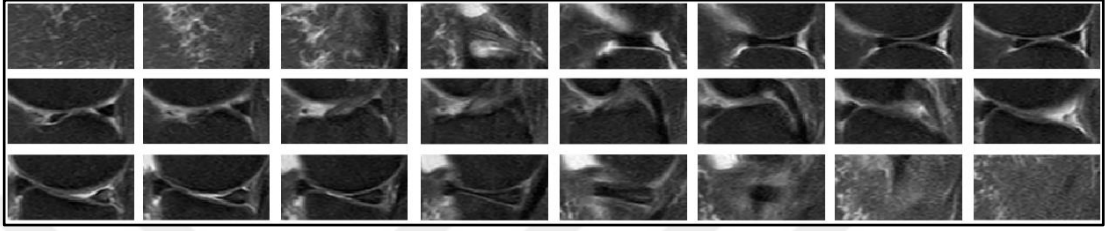


Şekil 4.9. Menüsküs bölgesinden menüsküs doku segmentasyonu

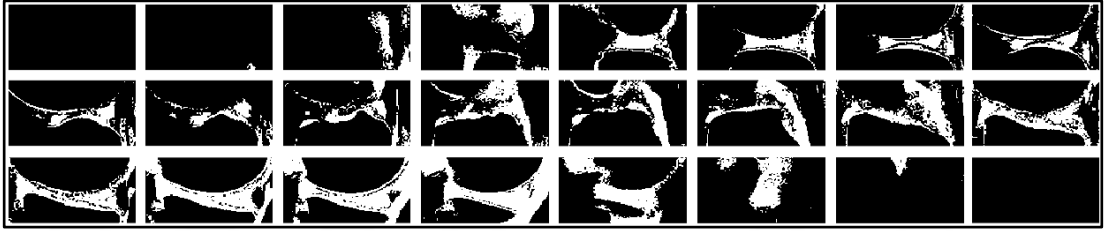
Çalışmanın bu kısmında şekil 4.9.'da adımları görülen menüsküs segmentasyonun her bir adımı sırasıyla anlatılacaktır. Bu kısımda gerçekleştirilen işlemlerin anlatımı tek bir deneğe ait 24 kareden oluşan PD ve T1 ağırlıklı görüntü dizileri üzerinden gerçekleştirilecektir. Segmentasyon işlemi tamamlandıktan sonra binary görüntü dizisi içerisindeki menüsküs dokusunun üç boyutlu modeli sunularak menüsküs segmentasyonu işlemi tamamlanmış olur.



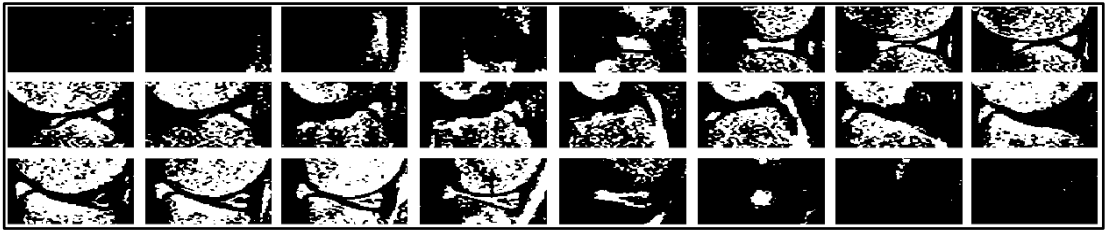
Şekil 4.10. T1 ağırlıklı görüntü dizisi



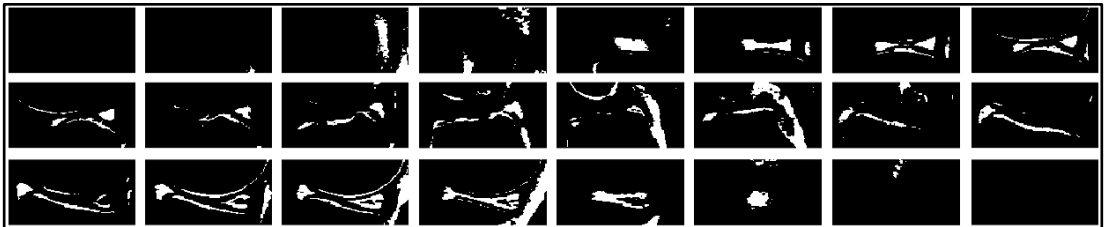
Şekil 4.11. PD ağırlıklı görüntü dizisi



Şekil 4.12. T1 ağırlıklı binary diz MR görüntü sekansı



Şekil 4.13. PD ağırlıklı binary diz MR görüntü sekansı

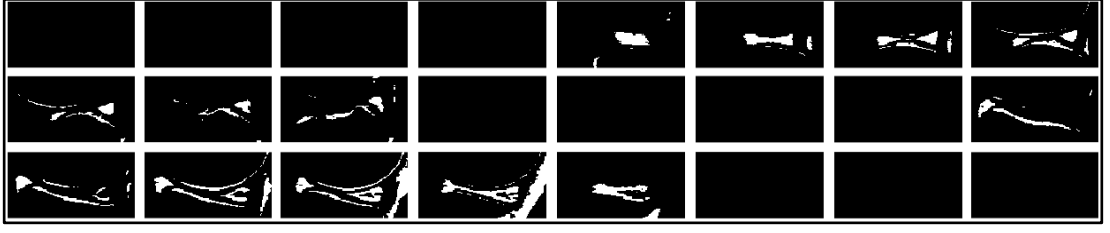


Şekil 4.14. T1 ve PD ağırlıklı binary görüntülerin çarpımı

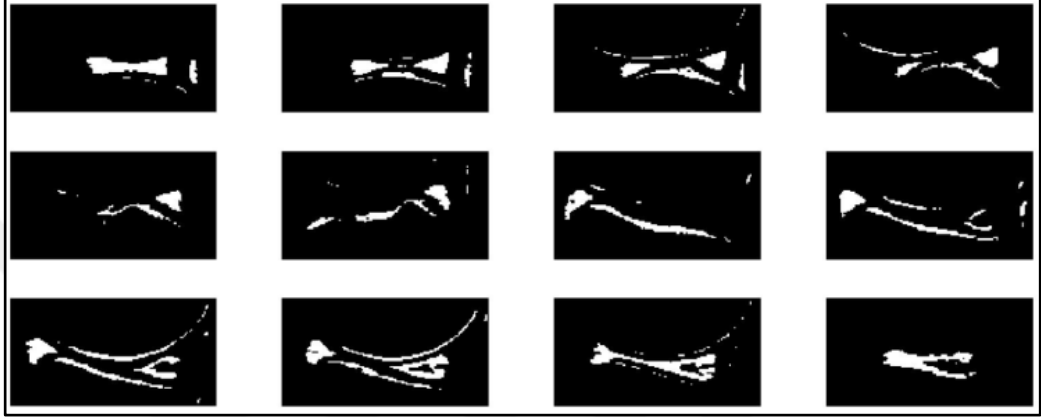
1) Şekil 4.10. ve 4.11.'de T1 ve PD ağırlıklı MR görüntü dizilerinin menüsküs bölgesinin tespitinden sonra görüntü içerisinde sadece menüsküs bölgesi kalacak şekilde kırılmış halleri görülmektedir. Bu kısımda gerçekleştirilmiş olan işlemler kırılmış T1 ve PD ağırlıklı görüntü dizileri üzerinden başlatılmıştır. İlk olarak 8-bit derinliğe sahip T1 ve PD ağırlıklı MR görüntü dizileri binary görüntülere çevrilir. T1 ve PD ağırlıklı görüntü dizilerinin her ikisinde de menüsküs dokusu düşük yoğunluk değerleri sergilediği için binary dönüşümün sonucunda menüsküs dokusunun görüntü içerisinde kalmasını garanti edecek 35 eşik değeri seçilmiştir. T1 ve PD ağırlık görüntülerde eşik değerin üzerindeki değere sahip pikseller binary görüntüde 0 eşik değerin altındaki değere sahip pikseller 1 değerini alçak şekilde dönüşüm işlemi gerçekleştirilir. Şekil 4.12. ve 4.13.'de sırasıyla T1 ve PD ağırlıklı görüntü dizisinden elde edilmiş binary görüntü dizileri verilmiştir.

Binary dönüşümün sonucunda elde edilen görüntülerde menüsküs dokusu ile birlikte bazı istenmeyen doku kısımları da görüntü içerisinde yer alır. Ayrıca bu görüntülerde menüsküs dokusu kendisini çevreleyen dokulardan tam olarak ayrılmış değildir. Bu durum morfolojik yöntemleri kullanmayı imkânsız kılar. Menüsküsü çevreleyen dokular, seçilen eşik değere göre binary dönüşüm sırasında zıt özellik gösterdiğinden menüsküs dokusunun çevresini saran dokulardan ayrıştırılması binary PD ve T1 ağırlıklı görüntülerin çarpılması ile sağlanır. Bu çarpma işleminin sonucunda elde edilen sonuç görüntüsünde çevresinden ayrıştırılmış menüsküs dokusu ile birlikte bazı istenmeyen doku parçaları bulunur. Şekil 4.14.'te binary görüntülerin noktasal çarpımlarını içeren görüntü dizisi yer almaktadır.

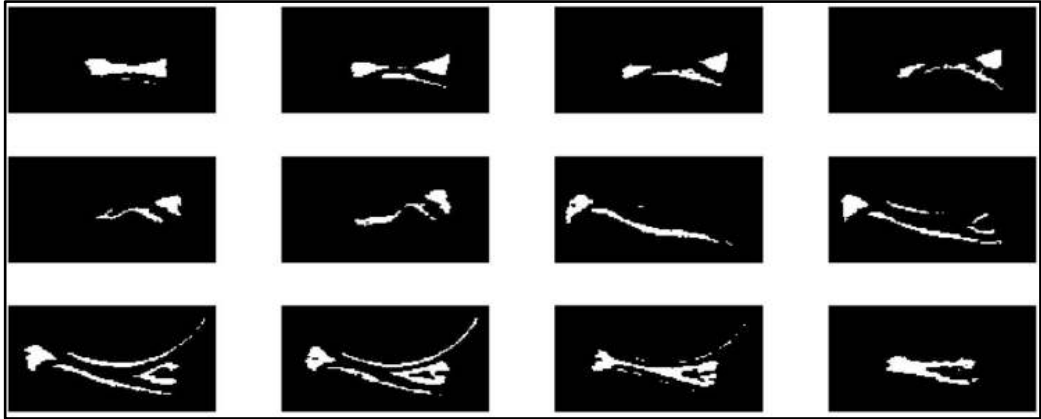
Binary çarpımların elde edilmesinden sonra görüntü dizisi içerisinde menüsküs dokusunun net olarak bulunduğu başlangıç ve bitiş görüntü kareleri seçilerek menüsküs dokusunun yer almadığı veya net olmadığı görüntü kareleri temizlenerek görüntü dizisinden çıkartılır. Böylece görüntü dizisi içerisinde sadece menüsküs dokusunun net olarak algılandığı görüntü kareleri kalır. Aşağıda şekil 4.15.'te temizlenmiş görüntü dizisi verilmiştir. Bu aşamadan sonraki adımlarda morfolojik görüntü işleme teknikleri kullanılarak istenmeyen detaylar görüntü dizisi içerisinde çıkarılır.



Şekil 4.15. Menüsküs dokusu içermeyen karelerin temizlenmesi



Şekil 4.16. Sınırları temizlenmiş görüntü sekansı



Şekil 4.17. Sınırları temizlenmiş görüntülerden küçük nesnelerin kaldırılması

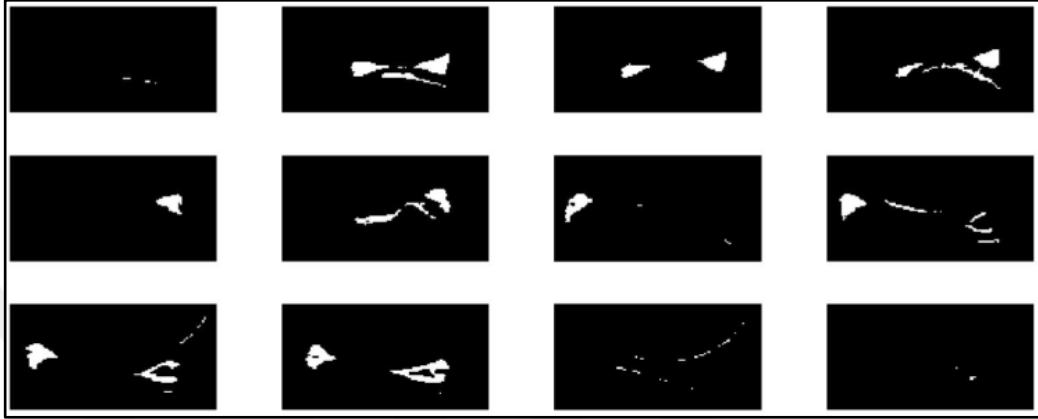
2) Binary görüntü dizisinde istenmeyen nesneleri kaldırmada ilk adım görüntülerin sınırlarına temas eden nesneleri kaldırmak olacaktır. Doğru olarak tespit edilmiş menüsküs bölgesinde görüntülerin sınırlarına temas eden nesneler menüsküs dokusuna ait olamayacakları için görüntülerden çıkartılmaları gerekir. Birçok uygulamada görüntülerin sınırlarına temas eden nesnelerin kaldırılması için farklı yöntemler kullanılmıştır. Bu çalışmada sınıra temas eden nesnelerin kaldırılması için kısım 3.6.'da detayları anlatılan genişleme ile morfolojik rekonstrüksiyon [40-42] kullanılmıştır. Genişleme ile morfolojik rekonstrüksiyon görüntü dizisi içerisindeki

her bir görüntü karesi için 2 boyutta ayrı ayrı uygulanarak sınır temizleme işlemi gerçekleştirilmiştir. Şekil 4.16.'da sınıra temas eden nesnelerin temizlendiği görüntü dizisi görülmektedir.

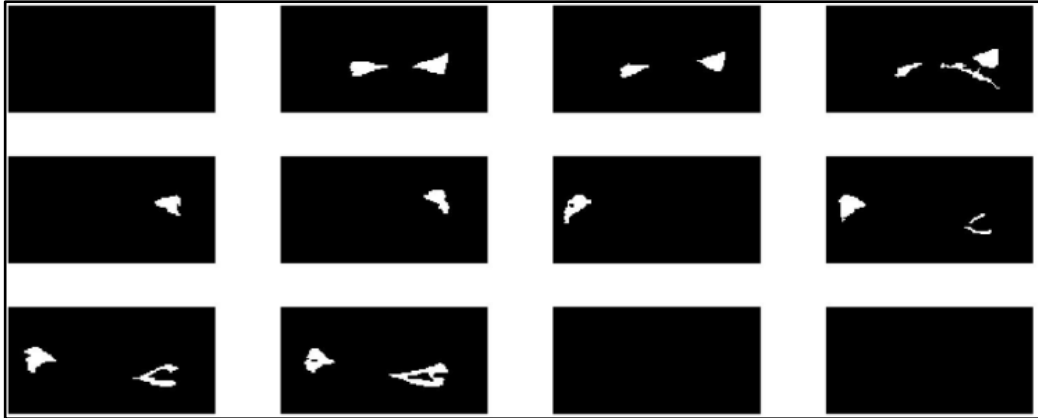
3) Bağlı bileşenler kısım 3.8.'de anlatıldığı gibi iki boyutlu görüntülerde 4 veya 8-bitişiklik üç boyutlu görüntülerde 6, 18 veya 26-bitişiklik şeklinde bağlanmış nesnelerdir [41]. Binary görüntülerde her bir bağlı bileşen ayrı bir nesneyi temsil eder. Bağlı bileşenlerin etiketlenmesinde [43] kullanılan yöntemlerden [42, 44] bir tanesi de morfolojik geriçatmadır. Görüntü içerisindeki nesneler bir kez tespit edilip etiketlendikten sonra kendilerine has özelliklerinden faydalanılarak filtreleme işlemleri gerçekleştirilebilir. Bu kısımda görüntü içerisinde kalan küçük nesnelerin filtrelenmesinde MR görüntü dizisi üç boyutlu olarak ele alınmış ve bağlı bileşenler 26-bitişiklik ile çıkarılmıştır. Her bir görüntü karesinde tek tek küçük nesnelerin filtrelenmesi yerine görüntü dizisinin üç boyutlu olarak ele alınmasının nedeni görüntü dizisinde yırtılmış olarak bulunabilecek olan menüsküs dokularına ait küçük parçaların görüntüden çıkarılmasını engellemektir. Menüsküs dokusuna ait olmayan küçük nesnelerin filtrelenmesi 26-bitişiklik ile çıkarılmış toplam 200 piksellik hacimden daha küçük nesnelerin belirlenmesi ile gerçekleştirilmiştir. Morfolojik işlemler boyunca her bir filtreleme işleminden sonra filtrelenen nesnelerden geriye kalabilecek olan küçük nesnelerin kaldırılması işlemi gerçekleştirilmiştir. Şekil 4.17.'de sınır temizleme işleminden sonra sınıra temas eden nesnelerden geriye kalan küçük nesnelerin kaldırıldığı görüntü dizisi verilmiştir.

4) Buraya kadar gerçekleştirilen işlemlerden sonra şekil 4.17.'de görüldüğü gibi görüntü dizisi içerisinde menüsküs dokusu ile birlikte kemik dokularına ait bazı kısımlar kalacaktır. Görüntü içerisinde kalan kemik dokusunun en belirgin ayırt edici özelliği görüntünün yatay eksenini boyunca en büyük uzunluğa sahip olan nesne olmasıdır. Görüntü içerisinde kalan kemik dokusunun büyük bir kısmını bir yatay aks uzunluğu filtresi kullanarak görüntüden kaldırabiliriz [40]. Bu kısımda yatay aks uzunluğu filtresi her bir görüntü karesinde 8-bitişiklik ile çıkarılmış nesnelere uygulanmış ve yatay eksen boyunca 40 pikselden fazla uzunluğa sahip olan nesneler görüntü içerisinde çıkarılmıştır. Yatay aks uzunluğu filtresinin uygulanmasından sonra filtre edilen kemik dokularından geriye kalan küçük nesneler görüntü

içerisinden filtrelenmiştir. Yatay aks uzunluğu filtresi uygulandıktan sonra elde edilen sonuçlar şekil 4.18.'de görülmektedir. Şekil 4.19.'da ise filtre edilmiş kemik dokularından geriye kalan küçük nesnelerin kaldırılmasından sonraki sonuçlar verilmiştir.



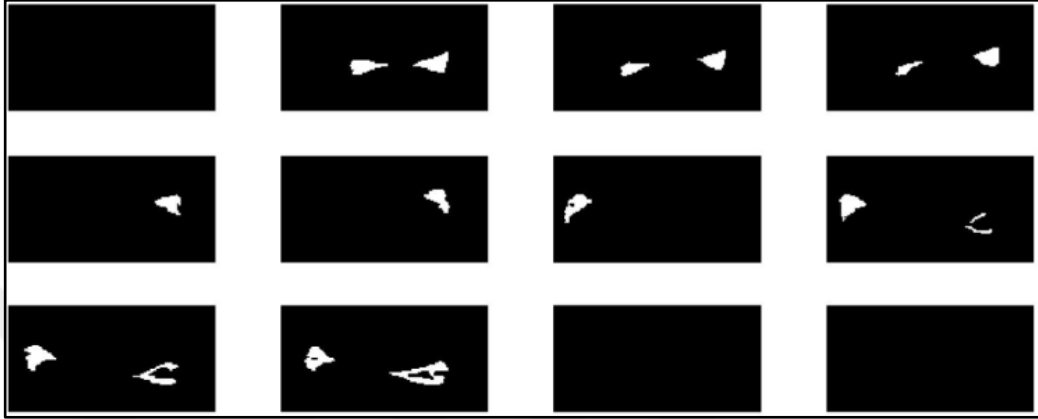
Şekil 4.18. Yatay aks filtresi ile ilk filtreleme



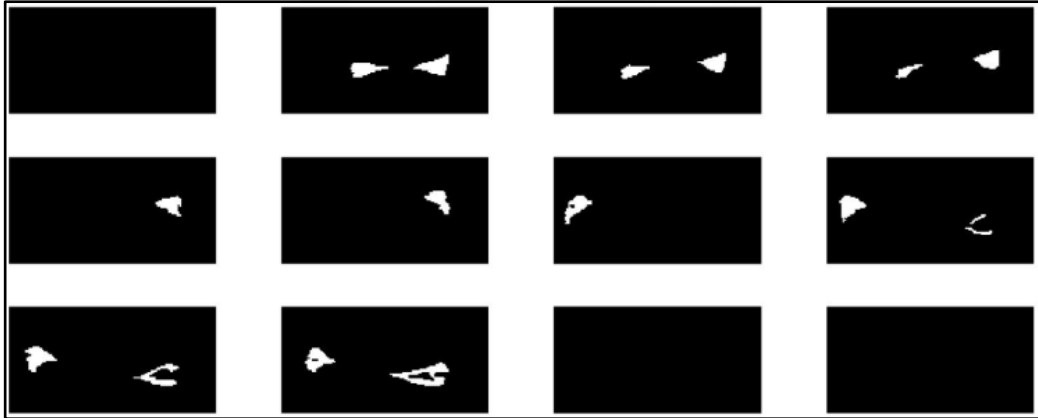
Şekil 4.19. Yatay aks filtresinden sonra küçük nesnelerin kaldırılması

5) Bu adımda uygulanan yatay aks uzunluğu filtresi ile kaldırılamayan iç ve dış menüsküs arasında kalan yatay eksen boyunca daha küçük uzunluğa sahip kemik dokusu parçalarının kaldırılması için ikinci bir filtreleme işlemi gerçekleştirilmiştir. Bu aşamada geriye kalan kemik dokuları daha kısa parçalar halinde ve iç ve dış menüsküs arasında kalacağından kemik dokusunun filtrelenmesi için şu şekilde bir yöntem izlenir: Görüntü içerisinde yatay doğrultuda menüsküs dokusunun başladığı ve bittiği noktalar arası yatay uzunluk hesaplanır ve bu uzunluğun %50'sinden daha fazla %90'ından daha az yatay aks uzunluğuna sahip nesneler filtrelenir. Bu sayede daha önce uygulanan yatay aks uzunluğu filtresinden geçmiş olan daha küçük yatay

eksen uzunluğuna sahip kemik dokuları görüntülerden kaldırılmış olur. Bu filtreleme işleminin sonucu şekil 4.20.'de görülmektedir. Şekil 4.21.'de ise ikinci yatay aks uzunluğu filtresinden sonra geriye kalan küçük nesnelerin filtrelenmiş hali görülmektedir.



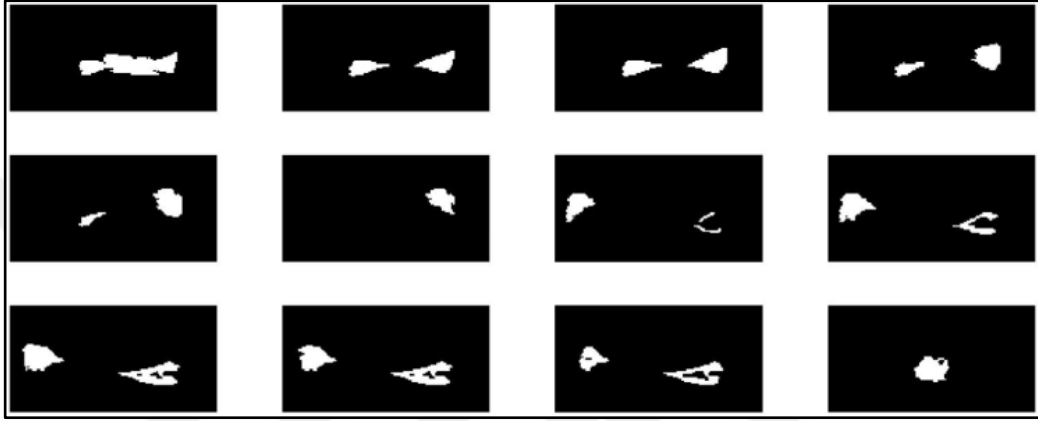
Şekil 4.20. Yatay aks filtresi ile ikinci filtreleme



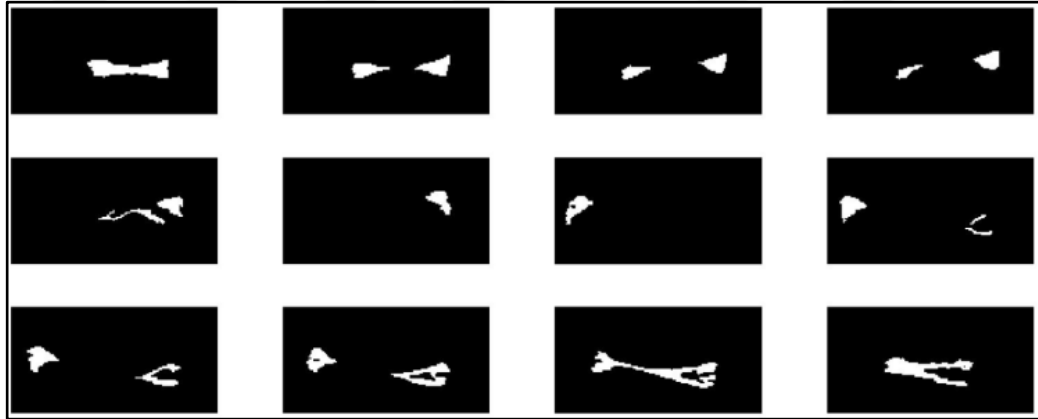
Şekil 4.21. İkinci yatay aks filtresinden sonra küçük nesnelerin kaldırılması

6) Buraya kadar gerçekleştirilen tüm bu işlemlerden sonra görüntü dizisi içerisinde sadece menüsküs dokusu kalmıştır. Fakat tüm bu filtreleme süreçleri boyunca istenmeden bazı menüsküs dokuları da görüntüden kaldırılmış olabilir. Şekil 4.21.'deki görüntü dizisinde ilk ve son iki görüntü karesindeki menüsküs dokularının filtreleme işlemleri ile silinmiş olduğu görülmektedir. İstenmeden silinen menüsküs dokularının tekrar oluşturulması morfolojik geriçatma ile gerçekleştirilebilir. Mevcut menüsküs dokuları üç boyutlu bir yapılandırma elemanı ile büyütülerek dokuların silindiği görüntü kareleri için işaretleyici görüntüsü elde edilir. Bu işaretleyici sayesinde istenmeden görüntüden kaldırılmış olan menüsküs dokuları filtre işlemleri

uygulanmamış görüntüden (şekil 4.16.) tekrar inşa edilir. Şekil 4.22.'de şekil 4.21.'den üç boyutlu bir yapılandırma elemanı (1x1x3) ile genişletilerek elde edilmiş işaretleyici görüntü dizisi görülmektedir. Şekil 4.23.'te ise şekil 4.22.'deki işaretleyici görüntü dizisinden morfolojik geriçatma ile inşa edilmiş görüntü dizisi görülmektedir. Bu adımla birlikte menüsküs dokusunun MR görüntü dizisinden segmentasyonu işlemi tamamlanmış olur.

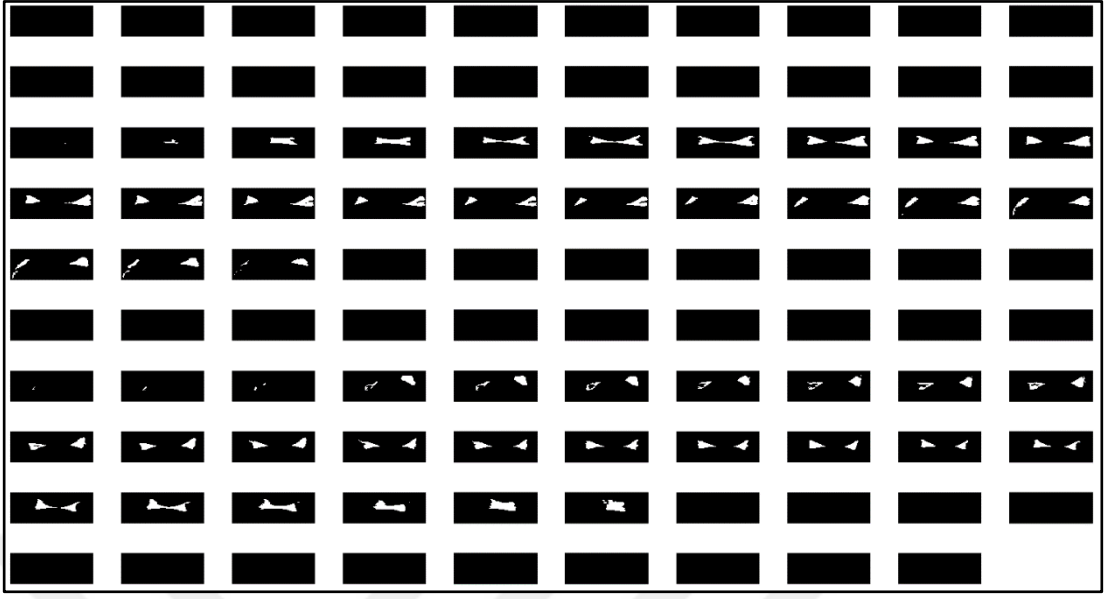


Şekil 4.22. Üç boyutlu yapılandırma elemanı ile genişleme

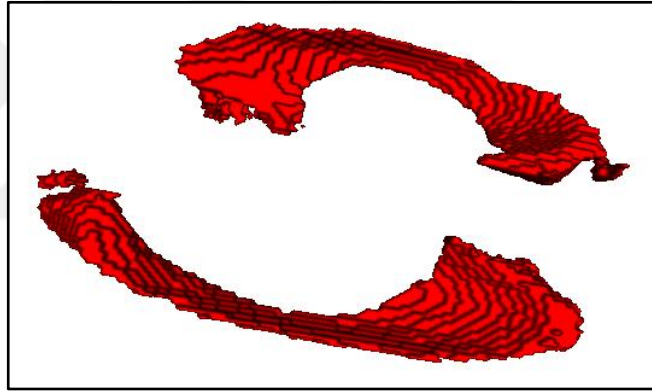


Şekil 4.23. Morfolojik rekonstrüksiyon ile bölge büyütme

7) MR görüntü dizisi içerisinde filtreleme işlemlerinin tamamlanmasından sonra elde edilen görüntüler ve bu görüntülerden elde edilmiş olan üç boyutlu menüsküs görüntüsü hekim değerlendirmesi için sunulabilir. Şekil 4.24.'de 99 adet görüntü karesine sahip T1 ve PD ağırlıklı MR görüntü dizilerinden bu çalışmada önerilen algoritma ile segmente edilmiş menüsküs dokusu görülmektedir. Şekil 4.25.te ise segmente edilmiş menüsküs dokusunun üç boyutlu olarak görselleştirilmiş hali görülmektedir.



Şekil 4.24. Segmente edilmiş menüsküs dokusu



Şekil 4.25. Segmente edilen menüsküs dokusunun üç boyutlu sunumu

5. SONUÇLAR VE DEĞERLENDİRME

Bu kısımda önerilen otomatik menüsküs segmentasyonunun genel başarısı değerlendirilmiş ve bu tez çalışmasının ileriki dönemlerde ne tür çalışmalara öncülük edebileceğinden bahsedilmiştir. Değerlendirme işleminde 10 adet deneğe ait 24 kareden oluşan T1 ve PD ağırlıklı MR görüntü sekansları kullanılmıştır. Kullanılan MR görüntü sekanslarına ait özellikler bölüm 4’te verildiği gibidir.

5.1. Değerlendirme Yöntemi

10 adet deneğe ait otomatik segmentasyon sonuçları bir uzman tarafından T1 ve PD ağırlıklı işlenmemiş sagittal düzlem görüntülere bakarak değerlendirilmiş ve segmentasyon doğruluğu hesaplanmıştır. Otomatik segmentasyonla elde edilen sonuç görüntüsündeki tüm pikseller tablo 5.1.’de görülen karışıklık matrisine göre doğru pozitif (True Positive, TP), yanlış pozitif (False Positive, FP) ve yanlış negatif (False Negative, FN) olarak tespit edilmiştir. Hesaplamalarda kullanılmayacağı için doğru negatif (True Negative, TN) değerleri dikkate alınmamıştır.

Tablo 5.1. Karışıklık matrisi

	Tahmin Edilen Pozitif	Tahmin Edilen Negatif
Gerçekte Pozitif	Doğru Pozitif (TP)	Yanlış Negatif (FN)
Gerçekte Negatif	Yanlış Pozitif (FP)	Doğru Negatif (TN)

Şekil 5.1.’de görüldüğü gibi sonuç görüntüsünde bulunan ve menüsküs dokusuna ait olduğu uzman tarafından doğrulanan dokular TP, filtre edilmesi gereken fakat sonuç görüntüsü içerisinde kalan dokular FP, sonuç görüntüsünde olması gerektiği halde bulunmayan menüsküse ait dokular ise FN olarak belirlenmiştir.

Şekil 5.1.(a)’da T1 ağırlıklı MR görüntüsünden R-CNN ile çıkarılmış menüsküs bölgesi görülmektedir ve resim içerisindeki menüsküs dokusunun sınırları bir hat şeklinde işaretlenmiştir. Şekil 5.1.(b)’ de doğru olarak segmente edilmiş menüsküs dokusu görülmektedir ve bu görüntüdeki ön plan piksellerinin tamamı TP olarak kabul edilmiştir. Şekil 5.1.(c)’de segmentasyon sonucunda görüntü içerisinde menüsküs dokusu ile birlikte kalan kemik dokusuna ait bazı nesneler ise FP olarak

kabul edilmiştir. Şekil 5.1.(d)'de menüsküs dokusu içerisinde silinmiş kısımlar ise FN olarak kabul edilmiştir.



Şekil 5.1. MR görüntüsüne ait segmentasyon sonuçları

- (a) T1 ağırlıklı MR görüntüsü
- (b) Hatasız gerçekleştirilmiş segmentasyon sonucu
- (c) FP içeren segmentasyon sonucu
- (d) FN içeren segmentasyon sonucu

Literatürdeki bazı çalışmalarda segmentasyon doğruluğu lateral menüsküs ve medial menüsküs için ayrı ayrı hesaplanırken bazılarında bütünü üzerinden hesaplanmıştır. Bu çalışmada lateral ve medial menüsküsün bütünü üzerinden tek bir segmentasyon doğruluğu hesaplanmıştır.

Çalışmada elde edilen sonuç görüntülerindeki piksellerin etiketlenmesinden sonra başarı oranının hesaplanması için küp benzerlik katsayısı (Dice similarity coefficient, DSC) ve duyarlılık (sensitivity) olmak üzere iki farklı hacim tabanlı ölçüm tekniği kullanılmıştır.

5.1.1. Küp Benzerlik Katsayısı (Dice Similarity Coefficient, DSC)

Hacim tabanlı bir ölçüm yöntemi olan DSC, Sørensen Küp Katsayısı (Sørensen Dice Coefficient) olarak ta bilinir [47]. DSC mekânsal örtüşmeye bağlı olarak otomatik segmentasyon doğruluğunun hesaplanmasında sıklıkla kullanılan yöntemlerden biridir [19-24]. Eşitlik 5.1.'de orijinal hali verilen DSC, iki ayrık kümenin kesişimlerinin birleşimlerine oranını iki ile çarparak bir katsayı üretir. Bu iki ayrık küme birbirleri ile tamamen örtüştüğü zaman DSC sayısal olarak 1 değerini üretecektir. DSC'nin binary veri kümelerinde uygulandığı ise eşitlik 5.2.'de görüldüğü gibidir.

$$DSC = 2 \frac{X \cap Y}{X + Y} \quad (5.1.)$$

$$DSC = 2 \frac{TP}{(TP + TP + FP + FN)} \quad (5.2.)$$

5.1.2. Duyarlılık (Sensitivity)

Bir diğler hacim tabanlı doğruluk ölçme yöntemi de duyarlılık ölçümüdür. DSC'den farklı olarak duyarlılık ölçümü doğruluk değerini hesaplarken FP değerlerini hesaba katmaz. Eşitlik 5.3.'te görüldüğü gibi duyarlılığın hesabında yalnızca TP ve FN değerleri hesaba katılır. Segmentasyon sonucu içerisinde eğer hiç FN yoksa duyarlılık katsayısı bire eşit olacaktır. Segmente edilen bölge olması gerekenden büyük de olsa eğer içerisinde hiç FN yoksa duyarlılık değeri yine de 1'e eşit olacaktır. Bu yönüyle duyarlılık katsayısının segmentasyon yönteminin yalnızca segmente edilmesi istenilen bölgeye olan duyarlılığını ölçtüğünü söyleyebiliriz. Segmentasyon sonucunda görüntü içerisinde kalan ve segmente edilmek istenen bölgeye ait olmayan nesneler (FP) duyarlılık katsayısını düşürmez.

$$Sensitivity = \frac{TP}{(TP+FN)} \quad (5.3.)$$

5.1.3. Sonuçlar ve Değerlendirme

Tablo 5.2.'de 10 farklı deneğe ait bu çalışmada elde edilen segmentasyon sonuçlarının DSC ve duyarlılık katsayısı ile değerlendirilmesi, ortalama (mean, μ) ve standart sapma (deviation, σ) değeri ile birlikte sunulmuştur. Tablo 5.3.'de ise ilgili çalışmalara ait segmentasyon sonuçlarının DSC ve duyarlılık katsayısı ile ölçülmüş başarı oranları sunulmaktadır.

Tablo 5.3.'de görüldüğü gibi Tack ve arkadaşları [24] lateral menüsküste DSC ile ölçülmüş 0.889 başarı oranı ile en yüksek sonucu elde etmişlerdir. Bu çalışmada 10 farklı deneğe ait yapılan segmentasyon sonuçlarının DSC ile ölçülen başarı oranı ise bütün menüsküs üzerinde 0.888'dir. Saygılı ve arkadaşlarının [22] çalışmalarında ise duyarlılık değeri oldukça yüksek iken DSC değerleri düşüktür. Bunun sebebi duyarlılık katsayısının hesaplamasında hatalı pozitif değerlerin dikkate alınmamasıdır. Duyarlılık ölçümünde daha kaba hatlara sahip veya istenmeyen bileşenlere sahip olan segmentasyon sonuçları da yüksek duyarlılık katsayısına sahip olabilirler.

Bu çalışmada segmentasyon sonucunda FP pikseller üreterek başarıyı düşüren temel neden sonuç görüntüsünde kalan filtre edilememiş menüsküs dokusuna ait olmayan dokulardır. Şekil tabanlı bir yaklaşım [24] mevcut algoritmaya ilave ederek

istenilmeyen dokuların filtrelenmesinde daha yüksek başarı oranları elde edilebilir. Ayrıca gelecek çalışmalarda R-CNN'in derinliği artırılarak ve daha yüksek miktarda verilerle eğitilerek menüsküs bölgesinin daha ince hassasiyetle tespit edilmesi sağlanabilir. FN pikseller üreterek başarıyı düşüren temel neden ise menüsküs dokusunun MR görüntülerinde başlangıç ve bitiş karelerinde doğru olarak çıkartılamamasıdır. Menüsküs bölgesinin daha hassas olarak tespit edilmesi ve morfolojik işlemlerde yapılacak olan algoritmik yenilikler ile FN piksellere bağlı olarak gerçekleşen başarı düşüşünün önüne geçilebilir.

Tablo 5.2. On farklı deneye ait segmentasyon sonuçları

Denek	DSC	Sensitivity
1	0.972	0.985
2	0.752	0.820
3	0.910	0.981
4	0.825	0.959
5	0.909	0.972
6	0.938	0.984
7	0.956	0.980
8	0.943	0.967
9	0.791	0.877
10	0.888	0.940
$\mu \pm \sigma$	0.888 ± 0.075	0.947 ± 0.052

Tablo 5.3. İlgili çalışmalara ait sonuçlar. Sonuçlar ortalama artı/eksi standart sapma (eğer varsa) şeklinde verilmiştir.

Yazar	Menüsküs	DSC ($\mu \pm \sigma$)	Sensitivity ($\mu \pm \sigma$)
Fripp et al. [19]	MM	0.770 ± 0.100	0.720 ± 0.140
	LM	0.750 ± 0.100	0.720 ± 0.180
Paprocki et al. [20]	MM	0.783	0.771
	LM	0.839	0.790
Zhang et al. [21]	Bütün	0.820 ± 0.033	0.840 ± 0.058
Saygılı et al. [22]	Bütün	0.827	0.998
Aldrin F. [23]	Bütün	0.752 ± 0.054	0.796 ± 0.072
Tack et al. [24]	MM	0.838 ± 0.061	n/a
	LM	0.889 ± 0.024	n/a
Bu Çalışma	Bütün	0.888 ± 0.075	0.947 ± 0.052

5.2. Özet ve Gelecek Çalışmalar

Paylaşılmış ağırlık mimarisi ve iki boyutlu girişi ile CNN görüntü sınıflandırma problemlerinde oldukça uygun bir çözüm olarak karşımıza çıkmaktadır. Tek bir vektör şeklinde girişe sahip sıradan bir YSA'nın ilk gizli katmanındaki her bir nöron için giriş görüntüsündeki piksel kadar ağırlık vardır. Sıradan YSA'ların gizli katmanlarındaki her bir nöron kendinden önceki ve sonraki katmanlarda bulunan tüm nöronlarla bağlantılı olacağından, sıradan YSA'larda ağ derinliği ile birlikte parametre sayısı çok hızlı artış gösterir. Bu çok fazla sayıdaki parametre hızlı bir şekilde YSA'nın eğitim setini ezberlemesine neden olur [13, 14]. Sıradan YSA'ların aksine paylaşılmış ağırlık mimarisi ile CNN bellek kullanımını azaltırken öğrenme performansı artırır ve ezberleme probleminin önüne geçer.

Sıradan bir derin öğrenme algoritmasının öğrenme için gereksinim duyduğu veri miktarı binlerle ifade edilen cinstendir. CNN eğitimi için gerekli olan bu denli çok fazla veri gereksiniminin önüne ise öğrenme transferi ile geçilebilir. Daha önce çok fazla miktarda veri ile eğitilen ve benzer bir sınıflandırma görevini gerçekleştiren CNN öğrenme transferi ile az miktarda veri kullanılarak benzer problemlere adapte

edilebilir. Bu çalışmada daha önce beyin MR görüntülerini sınıflandırmak için kullanılan bir CNN 100 adet menüsküs görüntüsü ile menüsküs bölgesi tespiti için ayarlanmıştır (fine-tuned). Böylece medikal görüntü işlemede verinin kısıtlı olabileceği alanlarda da az miktarda veri kullanılarak bir CNN eğitilebileceği gösterilmiştir.

Menüsküs bölgesi MR görüntü dizilerinde R-CNN kullanılarak tespit edildikten sonra tespit edilen bölgeden menüsküs dokusunun segmentasyonu daha kolay hale gelmiştir. Bu çalışmada menüsküs dokusunun tespit edilen bölgeden segmentasyonu için iki farklı MR görüntü sekansı ve morfolojik işlemler kullanılmıştır. Menüsküs dokusu T1 ve PD ağırlıklı görüntülerde aynı yoğunluk değerlerini gösterirken menüsküsü çevreleyen dokuların farklı yoğunluk değerleri sergilemesi özelliğinden faydalanılarak menüsküs dokusu çevresindeki dokulardan ayrıştırılmıştır. Sonrasında ise menüsküs dokusuna ait olmayan dokular morfolojik işlemler kullanılarak görüntü dizisinden filtrelenmiştir. Bu çalışmada menüsküs bölgesinin R-CNN ile tespiti segmentasyon problemini lokalize etmeyi sağlarken iki ayrı MR görüntü sekansı kullanımı uygulanacak işlem sayısının azalmasını sağlamıştır.

Bu tez çalışmasında gerçekleştirilmiş olan otomatik menüsküs segmentasyonu uzman hekime menüsküs tanılarının konulması aşamasında yardımcı olan bir karar destek sistemi niteliğindedir. Ayrıca önerilen otomatik menüsküs segmentasyonu menüsküs anomali tespitleri için yapılacak olan gelecek çalışmaları için de bir temel teşkil etmektedir. İleriki çalışmalarda otomatik olarak segmente edilmiş olan çok sayıda menüsküs dokusu menüsküs anomalilerini tespit edebilecek olan derin yapay sinir ağlarının eğitiminde kullanılabilecektir. Bu sayede el ile çıkartılması çok ciddi zaman alan menüsküs veri seti daha yüksek kenar hassasiyeti ile çok daha kısa sürede elde edilebilecektir. Gelecek çalışmaların da tamamlanması ile MR görüntülerinden otomatik olarak çıkartılacak olan menüsküs dokuları otomatik olarak teşhis konularak bir ön tanı sistemi elde edilmiş olacaktır. Ek olarak bu çalışmada yapılan menüsküs segmentasyonu diz MR görüntülerinde yapılabilecek olan kırık ve kemik dokusu segmentasyonu gibi çalışmalar için bir ön adım olacaktır.

KAYNAKLAR

1. Carleton Thomas Vangsness, Jr., M.D., <https://www.vangsnessmd.com/surgical-procedures/meniscus-surgery/>, Şubat 2020.
2. Makris, E.A., Hadidi, P., Athanasiou, K.A., The knee meniscus: structure-function, pathophysiology, current repair techniques, and prospects for regeneration, *Biomaterials*, 32(30), 7411–31, 2011.
3. Bryceland, J.K., Powell, A.J., Nunn, T., Knee Menisci, *Cartilage*, 8(2), 99–104, 2017.
4. Özer, H., Yıldırım, A., Menisküs lezyonlarında görüntüleme, *Totbid Dergisi*, 17, 114-122, 2018.
5. Liew, S.C., Liew, S.W., Zain, J.M., Tamper Localization and Lossless Recovery Watermarking Scheme with ROI Segmentation and Multilevel Authentication, *Journal of Digital Imaging*, 26, 316–325, 2013.
6. Deepa, S., Bharathi, S., Efficient ROI Segmentation of Digital Mammogram Images using Otsu's N thresholding method, *International Journal of Engineering Research & Technology*, 2(1), 1-6, 2013.
7. Nagi, J., Abdul Kareem, S., Nagi, F., Ahmed, S.K., Automated Breast Profile Segmentation for ROI Detection Using Digital Mammograms, 2010 IEEE EMBS Conference on Biomedical Engineering & Sciences, Kuala Lumpur, Malaysia, 87-92, November – December, 2010.
8. John, N.W., Segmentation of Radiological Images. In: Neri E., Caramella D., Bartolozzi C. (eds) *Image Processing in Radiology, Medical Radiology (Diagnostic Imaging)*, Berlin, Heidelberg, Springer, 2008.
9. Samuel, A.L., Some studies in machine learning using the game of checkers, *IBM Journal of Research and Development*, 3(3), 210-229, 1959.
10. Ng, A., Machine Learning Yearning: Technical Strategy for AI Engineers, In the Era of Deep Learning, Andrew Ng., Draft Version, 2018.
11. Tajbakhsh, N., et al., Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?, *IEEE Transactions on Medical Imaging*, 35(5), 1299-1312, 2016.
12. Akilan, T., Wu, Q.J., Zhang, W., Video Foreground Extraction Using Multi-View Receptive Field and Encoder-Decoder DCNN for Traffic and Surveillance Applications, *IEEE Transactions on Vehicular Technology*, 68(10), 9478-9493, 2019.

13. Lu, Le., Zheng, Yefeng., Carneiro, Gustavo., Yang, Lin., Deep learning and convolutional neural networks for medical image computing: Advances in Computer Vision and Pattern Recognition, Springer, 2017.
14. Aghdam, H.A., Heravi, E.J., Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification, Springer, 1st edn., 2017.
15. Farabet, C., Couprie, C., Najman, L., LeCun, Y., Learning hierarchical features for scene labelling, IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(8), 1915-1929, 2013.
16. Krizhevsky, A., Sutskever, I., Hinton, G.E., ImageNet classification with deep convolutional neural networks, 25th International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, 1097-1105, December, 2012.
17. Girshick, R., Donahue, J., Darrell, T., Malik, J., Rich feature hierarchies for accurate object detection and semantic segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, Ohio, 580-587, June, 2014.
18. Köse, C., Gençalioglu, O., Şevik, U., An automatic diagnosis method for the knee meniscus tears in MR images, Expert Systems with Applications, 36(2), 1208–1216, 2009.
19. Fripp, J., Bourgeat, P., Engstrom, C., Ourselin, S., Crozier, S., Salvado, O., Automated segmentation of the menisci from MR images, 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Boston, Massachusetts, 510-513, June-July, 2009.
20. Paproki, A., Engstrom, C., Chandra, S.S., Neubert, A., Fripp, J., Crozier, S., Automated segmentation and analysis of normal and osteoarthritic knee menisci from magnetic resonance images data from the osteoarthritis initiative, Osteoarthritis and Cartilage, 2(9), 1259-1270, 2014.
21. Zhang, K., Lu, W., Marziliano, P., The unified extreme learning machines and discriminative random fields for automatic knee cartilage and meniscus segmentation from multi-contrast MR images, Machine Vision and Applications, 24(7), 1459-1472, 2013.
22. Saygılı, A., Albayrak, S., A new computer-based approach for fully automated segmentation of knee meniscus from magnetic resonance images, Biocybernetics and Biomedical Engineering, 37(3), 432–442, 2017.
23. Aldrin, F., Automated segmentation of the meniscus, Master Thesis, KTH Royal Institute of Technology, 2017.

24. Tack, A., Mukhopadhyay, A., Zachow, S., Knee menisci segmentation using convolutional neural networks: data from the Osteoarthritis Initiative, Osteoarthritis and Cartilage, 26(5), 680-688, 2018.
25. Ng, A., <https://www.coursera.org/learn/neural-networks-deep-learning>, Haziran 2020.
26. Mitchell, T., Machine Learning. McGraw Hill, 1997.
27. Mohri, M., Rostamizadeh, A., Talwalkar, A., Foundations of Machine Learning, The MIT Press, 2012.
28. Ramasubramanian, K., Singh, A., Machine Learning Using R, 2nd edn., Apress, 2019.
29. Russell, S.J., Norvig, P., Artificial Intelligence: A Modern Approach, Third Edition, Prentice Hall, 2010.
30. van Engelen, J.E., Hoos, H.H. A survey on semi-supervised learning, Mach Learn, 109, 373–440, 2020.
31. Caudill, M., Neural Network Primer: Part I, AI Expert, 2(12), 46-52, 1987.
32. Rosenblatt, F., The perceptron: A probabilistic model for information storage and organization in the brain, Psychological Review, 65(3), 386-408, 1958.
33. Matlab for Artificial Intelligence, <https://www.mathworks.com/solutions/deep-learning/convolutional-neural-network.html>, Haziran 2020.
34. <https://developer.apple.com/library/content/documentation/Performance/Conceptual/vImage/ConvolutionOperations/ConvolutionOperations.html>, Haziran 2020.
35. CS231n: Convolutional Neural Networks for Visual Recognition, Stanford university course notes, <https://cs231n.github.io/convolutional-networks/>, Haziran 2020.
36. Rohrer, B., <https://end-to-end-machine-learning.teachable.com/p/how-deep-neural-networks-work>, Haziran 2020.
37. Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W., Selective search for object recognition, International Journal of Computer Vision, 104(2), 154-171, 2013.
38. Girshick, R., Fast R-CNN., Proceedings of the IEEE international conference on computer vision, Santiago, Chile, 1440-1448, 2015.

39. Ren, S., He, K., Girshick, R., & Sun, J., Faster R-CNN: towards real-time object detection with region proposal networks, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 6, 1137-1149, 2017.
40. Gonzalez, R.C., Woods, R.E., Eddins, S.L., *Digital Image Processing Using MATLAB*, 2nd edn., USA: Gatesmark Publishing, 2009.
41. Gonzalez, R.C., Woods, R.E., *Digital Image Processing*, 3rd edn., Pearson Prentice Hall, 2008.
42. Soille, P., *Morphological Image Analysis: Principles and Applications*, 2nd edn., Germany, Springer, 2002.
43. Solomon, C., Breckon, T., *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*, 1st edn., Wiley-Blackwell, 2011.
44. Sedgewick, R., *Algorithms in C*, 3rd edn., Addison-Wesley, 1998.
45. Marcus, D.S., et al., Human Connectome Project informatics: quality control, database services, and data visualization, *Neuroimage*, 80, 202–219, 2013.
46. <https://www.mathworks.com/help/images/ref/bwconncomp.html>, Haziran 2020.
47. Sørensen, T., A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons, *Kongelige Danske Videnskabernes Selskab*, 5(4), 1–34, 1948.

ÖZGEÇMİŞ

1982 yılında Kırıkkale’de doğan Emre ÖLMEZ, orta ve lise öğrenimini sırasıyla Kırıkkale Anadolu Lisesi, Yozgat Fen Lisesi ve Kırıkkale Süleyman Demirel Lisesinde tamamlamıştır. 2001 yılında kazandığı Kırıkkale Üniversitesi Mühendislik Fakültesi Elektrik-Elektronik Mühendisliği Bölümünü 2006 yılında başarıyla bitirmiştir.

2011 yılında yüksek lisans eğitimine Yozgat Bozok Üniversitesi Fen Bilimleri Enstitüsü Mekatronik Mühendisliği Anabilim Dalında başlamıştır. Doç. Dr. Halit ÖZTEKİN danışmanlığında hazırladığı “FPGA TABANLI MİKROBİLGİSAYAR MİMARİSİ KULLANILARAK DC MOTOR SÜRÜCÜ TASARIMI VE UYGULAMASI” başlıklı teziyle Yüksek Lisans öğrenimini 2013 yılında tamamlamıştır.

2013 yılında doktora eğitimine Bozok Üniversitesi Fen Bilimleri Enstitüsü Mekatronik Mühendisliği Anabilim Dalında başlamıştır. Doç. Dr. Orhan ER danışmanlığında hazırladığı “DERİN ÖĞRENME VE MORFOLOJİK GÖRÜNTÜ İŞLEME YÖNTEMLERİ İLE MR GÖRÜNTÜLERİNDE OTOMATİK MENÜSKÜS SEGMENTASYONU” başlıklı doktora tezini 2020 yılında sunmuştur. 2010 yılından beri Yozgat Bozok Üniversitesinde Mühendis olarak çalışmaktadır.

İletişim Bilgileri

Adres: Yozgat Bozok Üniversitesi Yapı İşleri ve Teknik Daire Başkanlığı YOZGAT

Telefon: (354) 212 10 70

Faks: (354) 212 10 68

E-posta: emre_olmez@hotmail.com