



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



MR GÖRÜNTÜLERİNDE GLİYOM
KARAKTERİZASYONU İÇİN 3B ANALİZ
TEMELLİ SINIFLAYICI MODEL TASARIMI

Abdulsalam HAJMOHAMAD

YÜKSEK LİSANS TEZİ

Elektrik Elektronik Mühendisliği Anabilim Dalı

Mart-2023
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Abdulsalam HAJMOHAMAD tarafından hazırlanan “MR Görüntülerinde Gliyom Karakterizasyonu İçin 3B Analiz Temelli Sınıflayıcı Model Tasarımı” adlı tez çalışması 16/03/2023 tarihinde aşağıdaki jüri tarafından oy birliği ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Elektrik Elektronik Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Prof. Dr. Salih GÜNEŞ

.....

Danışman

Dr. Öğr. Üyesi Hasan KOYUNCU

.....

Üye

Dr. Öğr. Üyesi Mehmet DEMİRTAŞ

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan KESEN
Enstitü Müdürü

TEZ BİLDİRİMİ

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

DECLARATION PAGE

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

İmza

Abdulsalam HAJMOHAMAD

Tarih: 16.03.2023

ÖZET

YÜKSEK LİSANS

MR GÖRÜNTÜLERİNDE GLİYOM KARAKTERİZASYONU İÇİN 3B ANALİZ TEMELLİ SINIFLAYICI MODEL TASARIMI

Abdulsalam HAJMOHAMAD

**Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Elektrik Elektronik Mühendisliği Anabilim Dalı**

Danışman: Dr. Öğr. Üyesi Hasan KOYUNCU

2023, 86 Sayfa

**Jüri
Prof. Dr. Salih GÜNEŞ
Dr. Öğr. Üyesi Hasan KOYUNCU
Dr. Öğr. Üyesi Mehmet DEMİRTAŞ**

Bilgisayarlı Tomografi (BT) ve Manyetik Rezonans (MR) görüntüleme teknikleri, tümörlerin tespiti için sıklıkla başvurulan tarama yöntemleridir. MR görüntüleme; beyindeki küçük bir yapıyı dahi tespit edilebildiği için, diğer yöntemlere göre beyin anormalliklerini vurgulamak için daha sık kullanılır. Literatür kapsamında Yüksek Dereceli Gliyom (YDG) ve Düşük Dereceli Gliyom (DDG) ayrımı, MR görüntülerinde yarı otomatik yaklaşımlar üzerinden gerçekleştirilmektedir. Diğer bir deyişle, iki boyutlu (2B) görüntü analizi temelli modeller ile sınıflama sağlanmaktadır. Bu noktada tam otomatik bir Bilgisayar Destekli Teşhis (BDT) sistemi, yalnızca üç boyutta (3B) tanımlı tümörün ele alınması ve bu amaca dayalı – uyarlanabilir bir sınıflama modeli ile gerçekleştirilebilir. Tez çalışması kapsamında tam otomatik bir BDT sisteminin sınıflama bölümü ele alınarak, beyin tümörlerinin 3B temelli sınıflandırılması için yenilikçi bir model geliştirilmiştir. Model kapsamında 3B MR görüntülerinde her sekans (T1, T2, T1c, FLAIR) bilgisi işlenmiş, giriş verilerini oluşturmak amacıyla bir 3B'den 2B'ye Özellik Dönüştürme Stratejisi (3t2FTS) önerilmiştir. 3t2FTS yaklaşımı içinde Birincil Derece İstatistikler (BDİ) işletilerek uzay dönüşümü sağlanmıştır. Burada temel amaç, sınıflayıcı birim olan transfer öğrenme yöntemlerine beslenecek bilgilerin verimli bir yaklaşım üzerinden sunulabilmesidir. 3t2FTS yaklaşımı ile 3B hacim bilgileri 2B görüntülere çevrilerek, tanımlayıcı 2B-ID imgeler elde edilmiştir. Çalışmada bu imgeler her bir tümör için oluşturulmuş, elde edilen veriler sekiz transfer öğrenme mimarisi (DenseNet201, InceptionResNetV2, InceptionV3, ResNet50, ResNet101, SqueezeNet, VGG19, Xception) üzerinde test edilmiştir. Uygun derin öğrenme mimarisinin belirlenmesi ve önerilen model performansının incelenmesi için, BraTS 2017/2018 veri setinde 210 YDG - 75 DDG ayrımı ele alınmıştır. Mimarilerin hiperparametreleri, ulaşılabilecek en yüksek performansları ortaya çıkarmak için kapsamlı bir şekilde analiz edilmiştir. Sistem performansının değerlendirilmesi için 2-kat çapraz geçerlilik test yöntemi işletilmiştir. Çalışma neticesinde, 3t2FTS yaklaşımı ve ResNet50 mimarisini içeren model ile YDG - DDG ayrımı için %80 sınıflandırma doğruluğu sağlandığı gözlemlenmiştir. Elde edilen başarı ve önerilen 3t2FTS temelli sınıflamanın gelişime açık olduğu tespit edilmiştir.

Anahtar Kelimeler: Beyin Tümörü, Birincil Derece İstatistikler, Evrişimli Sinir Ağları, Karar Destek Sistemi, Medikal Görüntü İşleme, Özellik Dönüşümü, Sınıflandırma, Transfer Öğrenme

ABSTRACT

MS THESIS

THE DESIGN OF 3D ANALYSIS-BASED FRAMEWORK FOR GLIOMA CHARACTERIZATION IN MR IMAGES

Abdulsalam HAJMOHAMAD

**Konya Technical University
Institute of Graduate Studies
Department of Electrical & Electronics Engineering**

Advisor: Asst. Prof. Dr. Hasan KOYUNCU

2023, 86 Pages

Jury

**Prof. Dr. Salih GÜNEŞ
Asst. Prof. Dr. Hasan KOYUNCU
Asst. Prof. Dr. Mehmet DEMİRTAŞ**

Computed Tomography (CT) and Magnetic Resonance (MR) imaging techniques are frequently used-scanning methods for the detection of tumors. MR imaging is more frequently used than other imaging modalities to highlight brain abnormalities since it can detect even a small structure in the brain tissue. In the literature, the discrimination of High-Grade Glioma (HGG) and Low-Grade Glioma (LGG) is performed using semi-automated approaches. In other words, the classification process is provided by examining Two Dimensional (2D) image analyses-based models. At this point, a fully-automated Computer-Aided Diagnosis (CAD) system can only be realized by handling the tumor in Three Dimensional (3D) and by designing an applicable – task-based classification framework. In the thesis, the classification part of a fully automated CAD is considered, and a promising model is suggested to grade the gliomas on a 3D basis. In the proposed model, all phase information (T1, T2, T1c, FLAIR) is evaluated in 3D MR images, and a 3D to 2D Feature Transform Strategy (3t2FTS) is offered to form the input data. In 3t2FTS, space transform is provided with the usage of First-order Statistics (FOS). Herein, the main purpose is to present the input information to be fed into the transfer learning architectures by using an efficient approach. Concerning this, 2D-ID images which are identifiers for tumors, are obtained by converting the 3D voxel information to 2D images. In our work, these images are formed for every tumor, and the data is tested on eight transfer learning algorithms (DenseNet201, InceptionResNetV2, InceptionV3, ResNet50, ResNet101, SqueezeNet, VGG19, Xception). To detect the most remarkable deep learning architecture and to examine the performance of the proposed model, the BraTS 2017/2018 dataset is evaluated on discrimination of 210 HGG and 75 LGG samples. Hyperparameters of architectures are comprehensively analyzed to reveal the highest performance to be reached. 2-fold cross-validation is chosen as the test method to evaluate the system performance. Consequently, it's observed that the framework operating 3t2FTS and ResNet50 can perform the HGG – LGG discrimination by achieving 80% classification accuracy. Regarding this, it's revealed that the achieved success score and the proposed 3t2FTS-based classification are promising in terms of progress.

Keywords: Brain Tumor, First-order Statistics, Convolutional Neural Networks, Decision Support System, Medical Image Processing, Feature Transform, Classification, Transfer Learning

ÖNSÖZ

Bütün çalışmalarında değerli bilgi ve tecrübeleriyle bana yol gösteren, gerekli araştırma ve geliştirme çabalarında yardımlarını esirgemeyen danışmanım Konya Teknik Üniversitesi Mühendislik Fakültesi Elektrik-Elektronik Mühendisliği öğretim üyesi Dr. Öğr. Üyesi Hasan KOYUNCU hocama, her türlü maddi ve manevi katkılarını esirgemeyen aileme ve beni yalnız bırakmayan arkadaşlarıma teşekkürlerimi sunarım.

Abdulsalam HAJMOHAMAD
KONYA-2023



İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR	ix
1. GİRİŞ	1
1.1. Tez Çalışmasının Önemi.....	2
1.2. Tez Çalışmasının Amacı	3
2. KAYNAK ARAŞTIRMASI	4
2.1. Literatür Çalışmaları	4
2.2. Kaynak Araştırması Özeti.....	6
3. MATERYAL VE YÖNTEM.....	8
3.1. Birincil Derece İstatistikler	9
3.2. Üç Boyuttan İki Boyuta Özellik Dönüşüm Stratejisi (3t2FTS).....	11
3.3. Derin Öğrenme	13
3.4. Transfer Öğrenme	14
3.5. Evrimsel Sinir Ağları	15
3.5.1. Giriş katmanı.....	17
3.5.2. Evrişim katmanı	17
3.5.3. Doğrultulmuş doğrusal birim katmanı	18
3.5.4. Havuzlama katmanı	19
3.5.5. Tam bağlantılı katman	20
3.5.6. Seyreltme katmanı	20
3.5.7. Sınıflandırma katmanı.....	21
3.5.8. Optimizasyon teknikleri.....	22
3.6. Transfer Öğrenme Yöntemleri.....	23
3.6.1. DenseNet201 mimarisi	23
3.6.2. InceptionResNetV2 mimarisi	25
3.6.3. InceptionV3 mimarisi	26
3.6.4. ResNet50 ve ResNet101 mimarileri	30
3.6.5. SqueezeNet mimarisi	30
3.6.6. VGG19 mimarisi.....	33
3.6.7. Xception mimarisi.....	34
3.7. Kullanılan Veri Seti Özellikleri	36
4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....	38
4.1. DenseNet201 Temelli Analizler	39

4.2. InceptionResNetV2 Temelli Analizler	40
4.3. InceptionV3 Temelli Analizler	41
4.4. ResNet50 Temelli Analizler	42
4.5. ResNet101 Temelli Analizler	43
4.6. SqueezeNet Temelli Analizler	44
4.7. VGG19 Temelli Analizler	45
4.8. Xception Temelli Analizler	46
4.9. Detaylı Analiz ve Tartışma	47
5. SONUÇLAR VE ÖNERİLER	52
KAYNAKLAR	54
EKLER	61



SİMGELER VE KISALTMALAR

Simgeler

$f(x,y)$: Görüntüyü temsil eden fonksiyon
$\delta(j,i)$: Kronecker Delta fonksiyonu
$h(i)$: i .nci gri seviyeye dair tekrarlanma (histogram) değeri
$p(i)$: i .nci gri seviyeye dair OYF değeri
μ	: Ortalama değer
σ	: Standart sapma değeri
μ_3	: Eğrilik değeri
μ_4	: Basıklık değeri

Kısaltmalar

ADD	: Ayrık Dalgacık Dönüşümü
AUC	: İşlem Karakteristik Eğrisi Altında Kalan Alan
BDİ	: Birincil Derece İstatistikler
BDT	: Bilgisayar Destekli Teşhis
BT	: Bilgisayarlı Tomografi
ÇKP	: Çok Katmanlı Perceptron
DDD	: Durağan Dalgacık Dönüşümü
DDG	: Düşük Dereceli Gliyom
DSA	: Derin Sinir Ağı
DSÖ	: Dünya Sağlık Örgütü
DVM	: Destek Vektör Makinesi
ESA	: Evrişimli Sinir Ağları
GDEM	: Gri Düzey Eş Oluşum Matrisi
GDDUM	: Gri Düzey Dizi Uzunluğu Matrisi
GİB	: Grafik İşlem Birimi
GH-KPSO	: Gauss Haritası Tabanlı Kaotik Parçacık Sürü Optimizasyonu
LRDF	: Öğrenme Oranı Düşüş Faktörü
MR	: Manyetik Rezonans
OYF	: Olasılık Yoğunluk Fonksiyonu
ÖÖE	: Özyinelemeli Öznitelik Eleme
RO	: Rastgele Orman
ROI:	: Region of Interest
TS	: Topluluk Sınıflandırıcı
VOI	: Volume of Interest
YDG	: Yüksek Dereceli Gliyom
YGH	: Yönlendirilmiş Gradyanların Histogramı
YİÖ	: Yerel İkili Örüntüler
2B	: İki Boyutlu
2B-ID	: 2B Kimlik Görüntü
3B	: Üç Boyutlu
3t2FTS	: Üç Boyuttan İki Boyuta Özellik Dönüşüm Stratejisi

1. GİRİŞ

Gliyom, Dünya Sağlık Örgütü (DSÖ) derecelendirme sistemine göre en sık görülen ve en hızlı büyüyen beyin tümör türlerinden biridir. Bu değerlendirmeye göre gliyomlar; Derece-I (pilositik astrositom), Derece-II (düşük dereceli gliyom), Derece-III (kötü huylu gliyom) ve Derece-IV (glioblastoma multiform) olmak üzere dört gruba ayrılmaktadır. Burada tümörlerin Yüksek Dereceli Gliyom (YDG) ve Düşük Dereceli Gliyom (DDG) olarak ayrımı, tedavi prosedürünü düzenlemek ve hastaların sağ kalım sürelerini tahmin etmek için önemli bir yer tutmaktadır (Latif ve ark., 2018; Kumar ve ark., 2020). YDG tipi tümörlere sahip hastaların yaşam süresi yaklaşık iki yıldır ve bu tip gliyomlar hızlı müdahale gerektirir. DDG tipi tümörlerin büyüme oranı, YDG tipi tümörlerin aksine düşük seviyelerde kalır ve DDG tip tümöre sahip hastaların hayatta kalma süreleri mümkün olduğunca uzun tutulabilmektedir (Saba ve ark., 2020).

Manyetik Rezonans Görüntüleme (MRG), doku incelemelerinde farklı anormallikleri ortaya koyması açısından tümörler belirlemede sıklıkla tercih edilmektedir (Gupta ve ark., 2019). Diğer bir deyişle, MRG diğer görüntüleme modalitelerine göre küçük doku değişikliklerini bile tanımlayabilmesi nedeniyle ön plana çıkmaktadır (Sharif ve ark., 2020). Ancak hem gerekli hem de gereksiz bilgileri içeren MRG kesitlerinin araştırılması tespit sürecini zorlaştırabilmektedir. Bununla ilgili olarak bir Bilgisayar Destekli Teşhis (BDT) sistemi, tıbbi uzmanların özellikle de radyologların terapötik girişimleri ayarlamasına yardımcı olabilmektedir (Gupta ve ark., 2019).

Derin öğrenme yöntemleri segmentasyon için nitelikli görevleri yerine getirmektedir ve Evrişimli Sinir Ağı (ESA) temelli farklı tasarımlar görüntü işleme konusunda sıklıkla ele alınmaktadır. Derin öğrenme tabanlı yöntemler; doku özneliklerinin Yapay Sinir Ağları (YSA), Destek Vektör Makinesi (DVM), vb. gibi sınıflayıcılarla işleme alındığı istatistiksel yöntemlere üstün olarak, hiyerarşi biçimindeki özellikleri öğrenmektedir. Bu nedenle derin ESA mimarileri; özellik çıkarma, segmentasyon ve sınıflandırma dâhil olmak üzere tıbbi görüntülerin analizinde başarıyla uygulanmaktadır (Fang ve ark., 2022; Li ve ark., 2022; Kronberg ve ark., 2022; Ghaffari ve ark., 2022). Ancak bahsi geçen çalışmalarda genel olarak İki Boyutlu (2B) imgeler üzerinden analizlerin sağlandığı görülmektedir.

Literatür kapsamında yapılan arařtırmalarda, beyin tümörü derecelendirmesini Üç Boyutlu (3B) doku analizleri ve etkin derin öğrenme mimarileri üzerinden sađlayan detaylı bir çalıřma mevcut deđildir.

1.1. Tez Çalıřmasının Önemi

Görüntü, 2B temelli analizde genişlik ve yükseklik kullanılarak ifade edilirken, 3B temelli analizlerde bu kavramlara ek olarak görüntü dilim sayısı tanımlanmaktadır. 2B görüntü temelli birçok sınıflama yapısı olmasına karşın, 3B analiz temelli görece az sınıflama yapısı mevcuttur. Ancak farklı tip tümörlerin sınıflandırılması veya derecelendirilmesinde kullanılacak olan veriler genel olarak 3B yapıda sunulur. Burada tek bir görüntü dilimi üzerinden 2B analiz yapıldığı durumda yüksek sonuçlar kaydedilse dahi, önerilebilecek sistem yarı-otomatik bir yapı olarak kalacaktır. Çünkü kullanıcı veya tıp uzmanı tarafından gerekli görüntü diliminin seçimi ile 2B temelli sınıflama işlemleri sağlanabilmektedir. Bu noktada hacimsel bilgi kaybı, tam otomatik bir sınıflama sisteminin tasarımında dezavantaj olarak ortaya çıkmaktadır ve 3B temelli bir BDT için hacimsel analizin gerçekleştirilmesi gerekmektedir.

YDG ve DDG ayrımının 3B analizler üzerinden gerçekleştirilmesi, uygulanması kolay olan 2B görüntü sınıflamasından daha zordur. Birçok literatür çalışmasında özellik analizleri ve sınıflandırma için, yalnızca bir dilim üzerinde inceleme ve analiz yapıldığı görülmüştür. Bu yaklaşımlar, elde edilen dilimin bitiřindeki dilimlere dair anlamlı bilgileri göz ardı etmektedir. Söz konusu eksikliđi gidermek için bir 3B veri deđerlendirme modelinin gereksinimi ortaya çıkmaktadır.

Beyin tümörlerinin ikili sınıflandırılması, tümör taramalarındaki görüntülere dair dezavantajlardan ötürü zorlu bir konudur. Bu dezavantajlardan birisi aynı tip tümörler için şekil, boyut ve yoğunluk özelliklerinin birbirinden çok farklı olabilesidir. Diđer dezavantaj ise farklı tümör tiplerinin bu özellikler üzerine benzer karakteristiklere sahip olabilesidir.

Söz konusu dezavantajlara göre sistem tasarımları iki olgu için optimize edilmelidir: 1) Özellik analizleri, 2) Sınıflandırma. Dezavantajların zorluđuna bađlı olarak, 3B beyin tümörlerinin ikili sınıflandırmasını ele alan sınırlı bilgi ve çalıřma mevcuttur. Tez çalışmasında, YDG ve DDG ayrımı için 3B MR verilerine dayalı etkin bir modelin tasarımı gerçekleştirilmiştir.

1.2. Tez Çalışmasının Amacı

Bilgisayarlı Tomografi (BT) ve Manyetik Rezonans (MR) görüntüleme modaliteleri, tümörlerin lokasyon ve tip tespiti için sıklıkla başvurulan tarama yöntemleridir. MR görüntüleme; beyindeki küçük bir bozulmayı dahi tespit edilebildiğinden, beyin dokusunda yer alan anormalliklerini vurgulamada diğer modalitelere göre daha çok ön plandadır. Ayrıca MR görüntüleme, gliyomların karakterizasyonunda ümit vadeden ve invazif olmayan bir yöntemdir.

Gliyom karakterizasyonu konusunda literatür çalışmaları incelendiğinde, önerilen sistemlerin neredeyse hepsinin yarı-otomatik sistemler olduğu görülmektedir. Bu sistemler, bir tıp uzmanı tarafından dışarıdan müdahale gerektirmekte ve iki boyutlu görüntülerin sınıflandırılmasını temel almaktadır. Bu nedenle söz konusu sistemler yarı-otomatik sistem olarak atfedilmektedir.

Yapılan incelemeler ışığında;

- 3B analiz, tam otomatik bir teşhis sistemi için gereklidir.
- Tam otomatik bir BDT sisteminin sınıflama birimi, voksel bazlı özellik çıkarma kullanacak şekilde tasarlanmalıdır. Diğer bir deyişle, tam otomatik bir BDT sisteminde sınıflama biriminin 3B tümöre adaptif olması gerekmektedir.
- Derin öğrenme mimarilerinin sınıflama alanında etkin oldukları bilinmektedir. Ancak bu mimarilerin, beyin tümörü derecelendirmede 3B veri temelli kullanımları kapsamlı ele alınmamıştır.
- 3B temelli görüntü voksel bilgisinin 2B temelli görüntüye evrilmesi ile özellik dönüşümü sağlayan, ayrıca bu veri ile geleneksel transfer öğrenme mimarilerinin tümör analizinde kullanımını gerçekleştiren bir çalışma mevcut değildir.

Söz konusu maddeler, tez çalışmasında tasarlanan sisteme dair görevleri ve temel özellikleri teşkil etmektedir. Çalışma kapsamında etkin bir boyut – özellik dönüşüm yaklaşımı önerilmektedir. Bu yaklaşım ile elde edilen veriler, geleneksel transfer öğrenme mimarileri üzerinden sınıflandırılmaktadır. Bu sayede beyin tümörlerini 3B analiz üzerinden ele almak ve sınıflandırmak için bir tam otomatik sınıflama modeli sunulmaktadır. Yine önerilen model, tümör hacmini 3B sunan herhangi bir segmentasyon çıktısına uyarlanabilir durumdadır.

2. KAYNAK ARAŞTIRMASI

Literatür çalışmalarında beyin tümör segmentasyonu birçok kez incelenmiş ve ele alınmıştır. Ancak beyin tümörlerinin sınıflandırılması, segmentasyon konusuna göre daha az değerlendirilmiştir. MR görüntüleri temel alındığı durumda ise sınıflama çalışmalarının genel olarak tümör / tümör olmayan, hastalıklı / sağlıklı ve YDG / DDG ayrımları üzerinden işlendiği görülmektedir.

2.1. Literatür Çalışmaları

Latif ve ark. (2018) Ayırık Dalgacık Dönüşümü (ADD), birincil & ikincil derece istatistikler ve Çok Katmanlı Perceptron (ÇKP) yaklaşımlarını içeren bir sistem önermişlerdir. Model kapsamında tüm sekans bilgileri (T1, T2, T1c, FLAIR) işlenerek sınıflama sağlanmıştır. 2B MR görüntülerinde tümör / tümör olmayan görüntülerin ayrımın için BraTS 2015 kütüphane verileri kullanılmıştır. Analizlerde eğitim için 39 YDG / 26 DDG örneği işletilirken, test için 110 YDG / DDG örneği dikkate alınmıştır. Diğer bir deyişle, sistem analizlerinde eğitim-test ayrımı üzerinden değerlendirme yapılmıştır. Sonuç olarak önerilen sistem, tümör mevcudiyetinin 2B temelli sınıflandırılmasında %96,73 doğruluk ve %99,3 AUC başarısı elde etmiştir. Ancak BraTS 2015 kapsamında tüm eğitim örnekleri dikkate alınmamış, ayrıca önerilen yöntemin yalnızca tümör mevcudiyetini ele alan yarı-otomatik bir sınıflama yapısı olduğu gözlemlenmiştir.

Gupta ve ark. (2019) çalışmalarında hem YDG / DDG hem de tümör / tümör olmayan ayrımını ele almışlardır. YDG / DDG kategorizasyonu için T2+FLAIR sekans kombinasyonları, morfolojik işlemler, doğal özellikler ve çoğunluk oylamasına dayalı Topluluk Sınıflandırıcısı (TS) işleten üç seviyeli bir sınıflama sistemi önerilmiştir. Tümör / tümör olmayan ayrımı ise T2+FLAIR sekans kombinasyonları, Gri Düzey Eş-oluşum Matrisi (GDEM), Gri Düzey Dizi Uzunluğu Matrisi (GDDUM), Yerel İkili Örüntüler (YİÖ) ve TS yaklaşımı içeren model ile gerçekleştirilmiştir. Deneylerde BraTS 2012 kütüphanesinden temin edilen iki veri seti kullanılmış ve değerlendirme için 10-kat çapraz doğrulama yöntemi işletilmiştir. Sonuç olarak, yarı-otomatik bir yapı teşkil eden sistem ile 2B MR görüntülerinin YDG / DDG ayrımı ortalama %96,75 doğruluk ile gerçekleştirilmiştir.

Bodapati ve ark. (2020) çalışmalarında Derin Sinir Ağı (DSA) algoritmalarına dayalı, iki kanallı bir sınıflandırma modeli sunmuştur. Model kapsamında InceptionResNetV2 ve Xception mimarileri kullanılmıştır. Sistem performans değerlendirmesi için 5-kat çapraz doğrulama tercih edilmiş ve BraTS 2018 kütüphanesinden temin edilen iki veri seti üzerinde deneyler gerçekleştirilmiştir. Sonuçlara göre önerilen iki kanallı DSA yapısının, BraTS 2018 veri setinde %93,69 doğruluk elde ettiği ve diğer derin öğrenme yaklaşımlarından daha iyi performans sergilediği tespit edilmiştir. Modelin giriş verilerinin 2B MR görüntüleri olması sebebiyle, modele dair çalışma koşulunun yarı-otomatik temelde olduğu gözlemlenmiştir.

Koyuncu ve ark. (2020) 3B MR görüntülerinde YDG / DDG örneklerinin ayrımını ele almışlardır. Bu amaçla T1+T2+FLAIR sekans kombinasyonları, Birincil Derece İstatistikler (BDİ) ve Wilcoxon özellik seçme yöntemi ile sınıflayıcı yapıya sunulacak veri eldesini sağlanmıştır. Modelin sınıflayıcı bölümünde Gauss Haritası Tabanlı Kaotik Parçacık Sürü Optimizasyonu (GH-KPSO) ve YSA yapısını hibritleştirilen bir optimize sınıflayıcı (GH-KPSO-YSA) işletilmiştir. Modelin testi için BraTS 2017/2018 veri seti üzerine deneyler gerçekleştirilmiş, performans değerlendirmesinde ise 2-kat çapraz doğrulama yöntemi tercih edilmiştir. Sonuç olarak önerilen model, beyin tümörlerinin 3B temelli sınıflandırılması üzerine %90.18 doğruluk ve %85.62 AUC başarısı sağlamıştır.

Kumar ve ark. (2020) 2B MR görüntülerinde YDG ve DDG örneklerini sınıflandırmak için Durağan Dalgacık Dönüşümü (DDD), Özyinelemeli Öznitelik Eleme (ÖÖE), BDİ ve Rastgele Orman (RO) içeren bir model önermişlerdir. Model kapsamında dokusal öznitelikler analiz edilerek, BraTS 2017/2018 veri seti üzerine testler gerçekleştirilmiştir. Deneylerde tüm sekans bilgileri kullanılmış ve performansı değerlendirmek için 5-kat çapraz doğrulama metodu işletilmiştir. Sonuç olarak önerilen model, beyin tümörlerinin 2B temelli sınıflandırılmasında %97,54 doğruluk ve %97,48 AUC başarısı sağlamıştır. Modelde tümör alanı da dâhil olmak üzere tüm beyin ve ilgili alan (*Region of Interest: ROI*) olarak belirtilen tümör alanı, özellik çıkarma işlemi için birlikte değerlendirilmiştir. Bu nedenle önerilen modelin, hem dilim hem de ROI seçimi gerektiren yarı-otomatik bir sınıflama yapısı ihtiva ettiği görülmüştür.

Saba ve ark. (2020) çalışmalarında Yönlendirilmiş Gradyanların Histogramı (YGH), YİÖ, derin özellikler ve bir sınıflandırıcı içeren model ile YDG / DDG etiketli 2B görüntülerin sınıflamasını ele almışlardır. Yapılan deneylerde değerlendirme

yöntemi olarak %50-%50 eğitim-test ayrımı tercih edilmiştir. TS metodu, üç veri kümesi için ortalama sınıflama performansı üzerine en iyi yaklaşım olmuştur. Önerilen model BraTS 2015, BraTS 2016 ve BraTS 2017 veri setlerinde sırasıyla %91.30, %91.47 ve %98.39 doğruluk kaydetmiştir. Ancak önerilen modelin, dilim seçimi gerektiren yarı-otomatik bir sınıflama yapısı olduğu gözlemlenmiştir.

Şerif ve ark. (2020) 2B MR görüntülerinde sağlıklı / hastalıklı örneklerin kategorizasyonu için T2+FLAIR sekans kombinasyonları, YGH, YİÖ, geometrik özellikler ve DVM tekniğini içeren bir model önermişlerdir. Sistem performans analizleri, BraTS 2013 ve BraTS 2015 kütüphane verilerini de içeren üç farklı veri seti üzerinden ele alınmıştır. Değerlendirmeler %50-%50 eğitim-test ayrımı üzerinden gerçekleştirilmiştir. Önerilen model ile BraTS 2013 ve BraTS 2015 veri setleri üzerine sırasıyla %98 ve %100 doğruluk oranları gözlemlenmiştir. Sonuç olarak, sağlıklı / hastalıklı görüntülerin 2B temelli sınıflandırılmasının, yarı-otomatik bir model ile gerçekleştirildiği görülmüştür.

2.2. Kaynak Araştırması Özeti

Literatür çalışmaları incelendiğinde; 3B temelli sınıflandırmanın, 2B tabanlı sınıflandırmaya göre başarı puanları ve işlenecek bilgi miktarı nedeniyle daha karmaşık bir problem olduğu görülmektedir. Ayrıca 3B tabanlı bilgiler, sınıflama görevini doğru bir şekilde gerçekleştirmek için kapsamlı analizler gerektirir. Diğer taraftan, bir uzman yardımı olmaksızın beyin tümörlerinin derecelerini bağımsız olarak sınıflandırabilecek bir tam-otomatik sistem, yalnızca 3B uzayda analiz edilen tümör görüntüleri kullanılarak gerçekleştirilebilir. Burada 3B MR görüntüleri üzerinden YDG / DDG ayrımı işleyen bir modelin tasarımı, tez çalışmasının temel motivasyonunu teşkil etmektedir. Literatür analizlerine göre aşağıda belirtilen maddeler hem tez çalışmasının içeriğini hem de literatürde konu ile ilgili gereksinimleri sunmaktadır:

- 3B MR görüntülerinde farklı beyin tümörlerini sınıflamak için yenilikçi bir özellik dönüşüm stratejisi
- Tümör alanının incelenmesinde 3B bilginin 2B veriye nasıl evrilebileceğinin yorumlandığı kapsamlı bir çalışma
- Tümör sınıflandırmada sekiz nitelikli transfer öğrenme mimarisinin karşılaştırmasını ele alan detaylı bir çalışma

- 3B MR temelli farklı sınıflama görevlerinde kılavuz olabilecek etkili bir model tasarımı

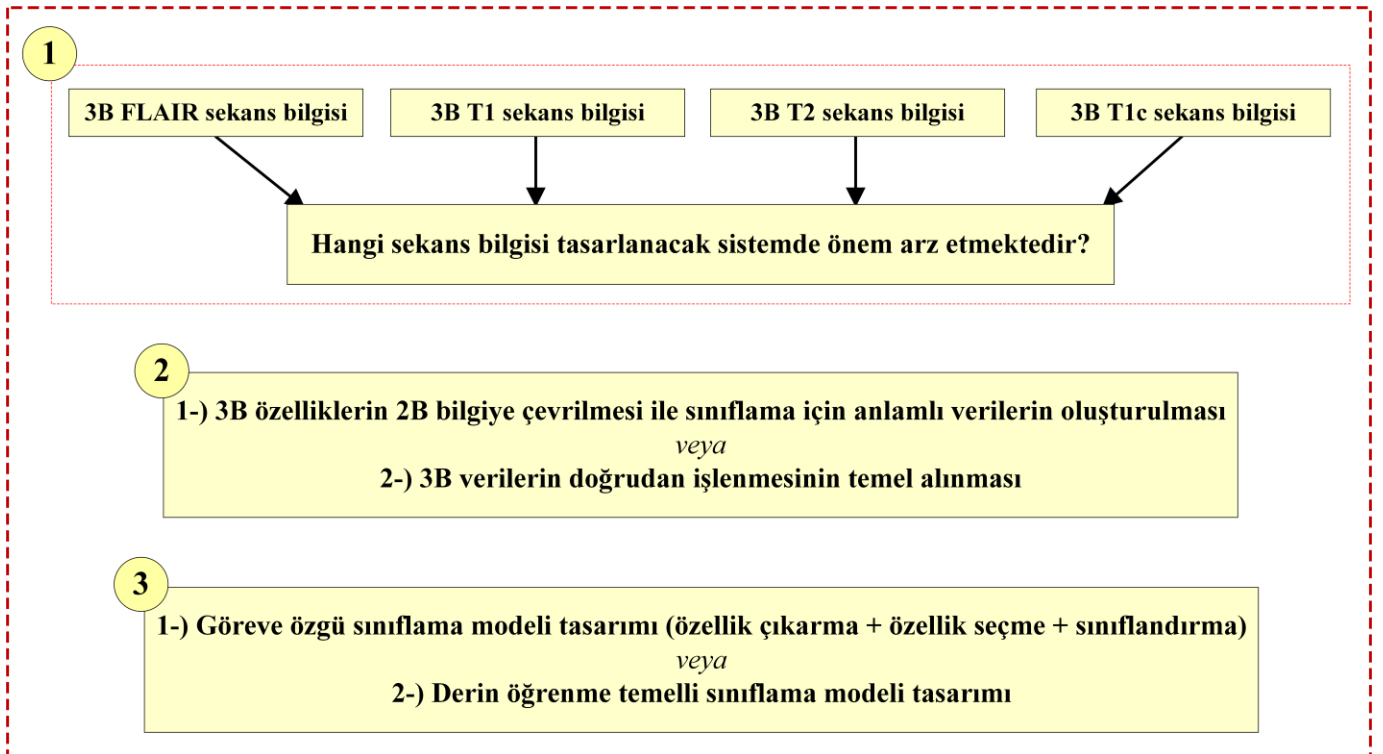


3. MATERYAL VE YÖNTEM

"Kaynak Araştırması" bölümündeki analizlere göre; çalışmalar sekans temelli incelendiğinde, bazı çalışmaların sekans bilgilerine dair iki kombinasyonu ele aldığı görülmektedir (Gupta ve ark., 2019; Şerif ve ark., 2020). Burada farklı sekansların incelendiği ve performans karşılaştırmalarının yapıldığı çalışmalar da mevcuttur (Koyuncu ve ark., 2020). Bazı çalışmalarda ise bütün sekans kombinasyonlarını içeren bilgilerin sistem girdisi olarak atfedildiği bilinmektedir (Latif ve ark., 2018; Kumar ve ark., 2020). Genel olarak çeşitli modeller için, her bir sekans bilgisi farklı kombinasyonlarda incelenerek en yüksek performans eldesi aranmakta veya bütün sekanslara dair bilgi kullanılmaktadır.

Literatür çalışmalarının çoğu 2B imgeler üzerinde işlem yapmaktadır. Burada önerilen modeller, genel olarak yarı-otomatik bir yapı içermektedir. 3B özellikler üzerinden işlem yapabilen çok kısıtlı çalışma mevcuttur (Koyuncu ve ark., 2020).

Literatürde özellik çıkarma, özellik seçme ve sınıflama görevlerinin yer aldığı göreve özgü sınıflayıcı modellerin tasarımı çoğunluktadır. Bu noktada derin öğrenme yöntemleri ile YDG / DDG ayrımı sağlayan kapsamlı bir çalışma mevcut değildir. Şekil 3.1'de tez çalışmasında değerlendirilen tasarı şeması sunulmaktadır.



Şekil 3.1. Tez çalışmasında değerlendirilen tasarı şeması

Şekil 3.1’de görüldüğü üzere; sekans bilgisi bir sınıflama modeline sunulabilecek olan girdiyi temsil etmektedir ve göreve özgü modellerde incelenmesi uygundur. Ancak tasarlanabilecek bir uzay dönüşüm işleminde (3B→2B) bütün sekanslardan gelebilecek bilginin kullanımı daha uygun olacaktır. Çünkü her bir MR sekansı farklı bilgileri barındırabilmektedir. Bu nedenle, tez çalışması kapsamında sunulan özellik dönüşüm stratejisi bütün sekanslardan gelen bilgileri işlemektedir. Özellik dönüşüm stratejisi kapsamında (3B bilginin 2B veriye evrilmesi ile) genel derin öğrenme mimarilerinin, YDG / DDG ayrımında işlevselliğe kavuşacağı ve 3B MR analizine uyarlanabilecekleri görülmektedir. Bu bilgiler ışığında; literatürde gereklilikler göz önüne alınarak, tez çalışmasının tasarımı şeması üç adımda elde edilmektedir. İlk adımda tüm 3B sekans bilgilerinin incelenmesi, ikinci adımda bu bilgilerin dönüşüm stratejisi ile 2B uzaya aktarılması, bu sayede (üçüncü adımda) etkin transfer öğrenme yaklaşımları ile tümör derecelendirmesi sağlanmaktadır.

3.1. Birincil Derece İstatistikler

Birincil Derece İstatistikler (BDİ), bir görüntünün histogram tabanlı yoğunluk analizleri kullanılarak elde edilmektedir. Histogram değerlendirmeleri ile ilgili olarak altı olgu (ortalama, standart sapma, eğrilik, basıklık, enerji ve entropi), literatürde en çok tercih edilen BDİ özelliklerini oluşturmaktadır (Materka ve Strzelecki, 1998; Koyuncu ve ark., 2020; Koyuncu and Barstuğan, 2021; Sakalli and Koyuncu, 2022).

2B görüntüyü simgeleyen fonksiyon $f(x,y)$ ile tanımlı olsun. Bu noktada ‘ x ’ ve ‘ y ’ değişkenleri görüntünün yatay ve dikey düzlemlerdeki koordinatları olarak tanımlanır. Bu değişkenler ise sırasıyla ($x=0,1,\dots,X-1$) ve ($y=0,1,\dots,Y-1$) olarak belirtilir. Burada ‘ G ’ görüntünün toplam yoğunluk sayısı olarak atansın. Bu aşamada ayırık bir yoğunluk değeri şeklinde görülecek olan ‘ i ’ ($f(x,y)$ fonksiyonunun çıktısı), yoğunluk seviyelerinin değerlerini gösteren $[0,G-1]$ aralığında değerlere sahip olabilecektir. Bununla ilgili olarak histogram, görüntü boyunca yoğunluk seviyelerine dair tekrar sayısının istatistiksel bir değerlendirmesi olarak ortaya çıkar (Materka ve Strzelecki, 1998; Koyuncu ve ark., 2020).

2B bir görüntü, genişlik ve yükseklik parametreleri temel alınarak tanımlanmaktadır. Burada 3B alanın tanımlanması için dilim sayısı da dikkate alınır. ‘ M ’, ‘ N ’ ve ‘ L ’ değişkenleri sırasıyla dilim genişliğini, dilim uzunluğunu ve dilim

sayısını gösterebilir. Burada ilgili hacim (*Volume of Interest: VOI*) ‘ M ’ genişliği, ‘ N ’ uzunluğu ve ‘ L ’ dilim sayısı çarpılarak elde edilir (Materka ve Strzelecki, 1998; Koyuncu ve ark., 2020).

Histogram ve Kronecker Delta fonksiyonu sırasıyla ‘ $h(i)$ ’ ve ‘ $\delta(j,i)$ ’ ile temsil edildiği durumda, bu değişkenler sırasıyla Eşitlik (3.1) ve Eşitlik (3.2)’de görüldüğü gibi hesaplanmaktadır (Materka ve Strzelecki, 1998; Koyuncu ve ark., 2020).

$$h(i) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \delta(f(x, y), i) \quad (3.1)$$

$$\delta(j, i) = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases} \quad (3.2)$$

‘ i ’ yoğunluğunun Olasılık Yoğunluk Fonksiyonu (OYF)’nu elde etmek için, ‘ $h(i)$ ’ değeri Eşitlik (3.3)’te görüldüğü gibi VOI’nin toplam voksel sayısına bölünmelidir (Materka ve Strzelecki, 1998; Koyuncu ve ark., 2020).

$$p(i) = \frac{h(i)}{1 * L * M * N}, \quad i = 0, 1, 2, 3, \dots, G-1 \quad (3.3)$$

VOI’ye dair BDİ özellikleri, OYF değerleri kullanılarak Eşitlik (3.4) – Eşitlik (3.9) arası sırasıyla ortalama (μ), standart sapma (σ), eğrilik (μ_3), basıklık (μ_4), enerji ve entropi için nicel şekilde hesaplanabilmektedir (Materka ve Strzelecki, 1998; Koyuncu ve ark., 2020).

$$\mu = \sum_{i=0}^{G-1} ip(i) \quad (3.4)$$

$$\sigma = \sqrt{\sum_{i=0}^{G-1} (i - \mu)^2 p(i)} \quad (3.5)$$

$$\mu_3 = \sigma^{-3} \sum_{i=0}^{G-1} (i - \mu)^3 p(i) \quad (3.6)$$

$$\mu_4 = \sigma^{-4} \sum_{i=0}^{G-1} (i - \mu)^4 p(i) - 3 \quad (3.7)$$

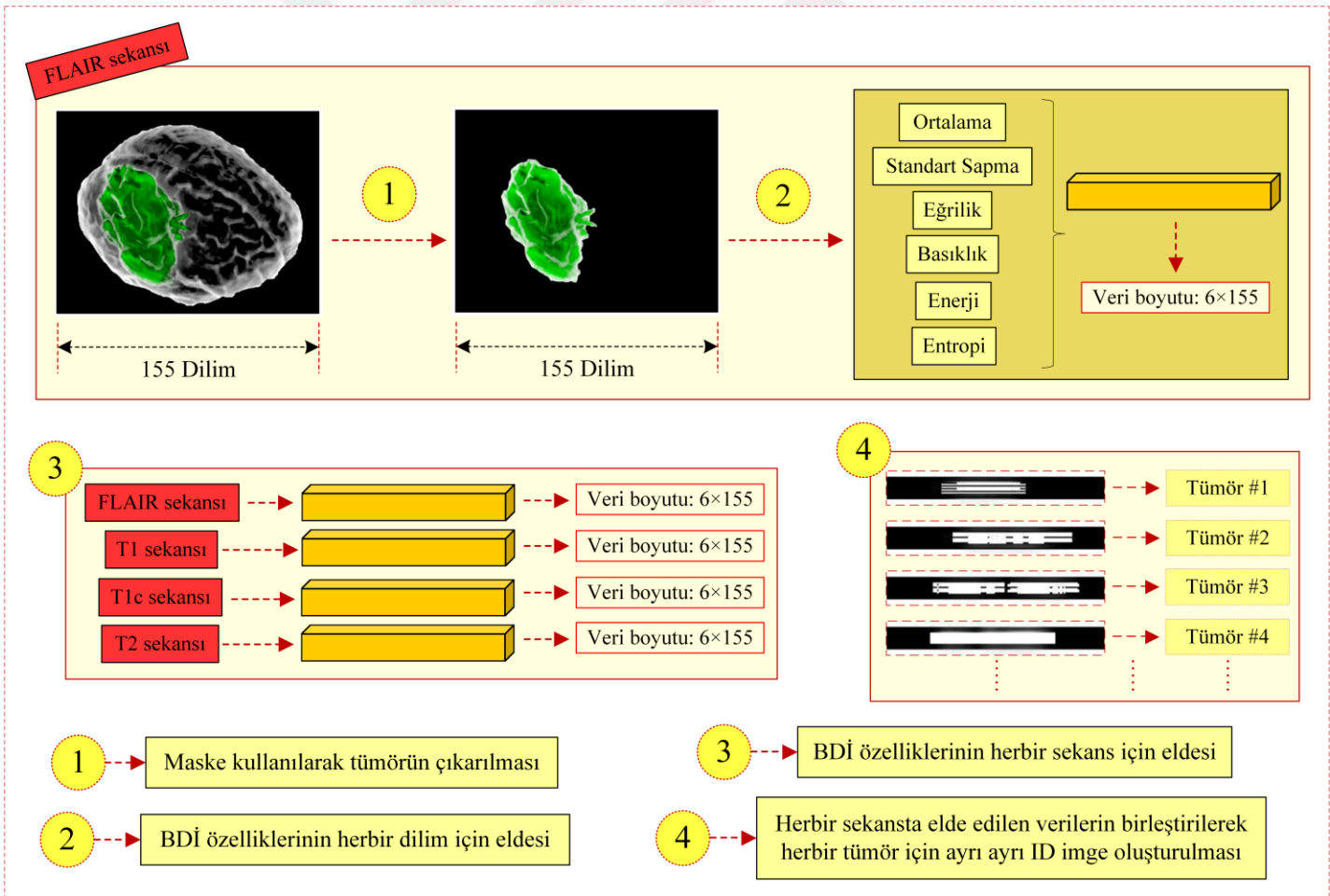
$$\text{Enerji} = \sum_{i=0}^{G-1} [p(i)]^2 \quad (3.8)$$

$$\text{Entropi} = - \sum_{i=0}^{G-1} p(i) \log_2 [p(i)] \quad (3.9)$$

3.2. Üç Boyuttan İki Boyuta Özellik Dönüşüm Stratejisi (3t2FSTS)

MRG analizlerinde tümör dokusu 3B uzay içinde sunulmaktadır ve farklı sekanslar (T1, T2, T1c, FLAIR) üzerinden doku değerlendirmeleri yapılmaktadır. Bu durum ise dört çeşit sekans kullanımı ile dört farklı 3B bilginin analizini gerektirir.

Veri işlem aşamasında tasarlanabilecek bir dönüşüm sistemi ile 3B bilgilerin 2B uzayda ifade edilmesi, hem verilerin daha rahat işlenebilmesini hem de her bir tümörlere dair 2B Kimlik Görüntüsü (2B-ID)'nün eldesini sağlayabilecektir. Bu amaçla, tez çalışması kapsamında her bir tümöre dair 2B-ID görüntüsü üretebilecek ve dört sekans bilgili 3B MR görüntülerinde rahatlıkla kullanılabilir etkin bir metot önerilmektedir. Üç Boyuttan İki Boyuta Özellik Dönüşüm Stratejisi (3t2FSTS) olarak isimlendirilen yaklaşımın kısaltması İngilizce karşılığı kullanılarak belirtilmiştir. Şekil 3.2'de 3t2FSTS yaklaşımının tasarımı şeması sunulmaktadır.



Şekil 3.2. 3t2FSTS metodu tasarımı şeması

Şekil 3.2’de görüldüğü üzere, 3t2FTS yöntemi kapsamında dört temel adım işletilmektedir. Bu noktada;

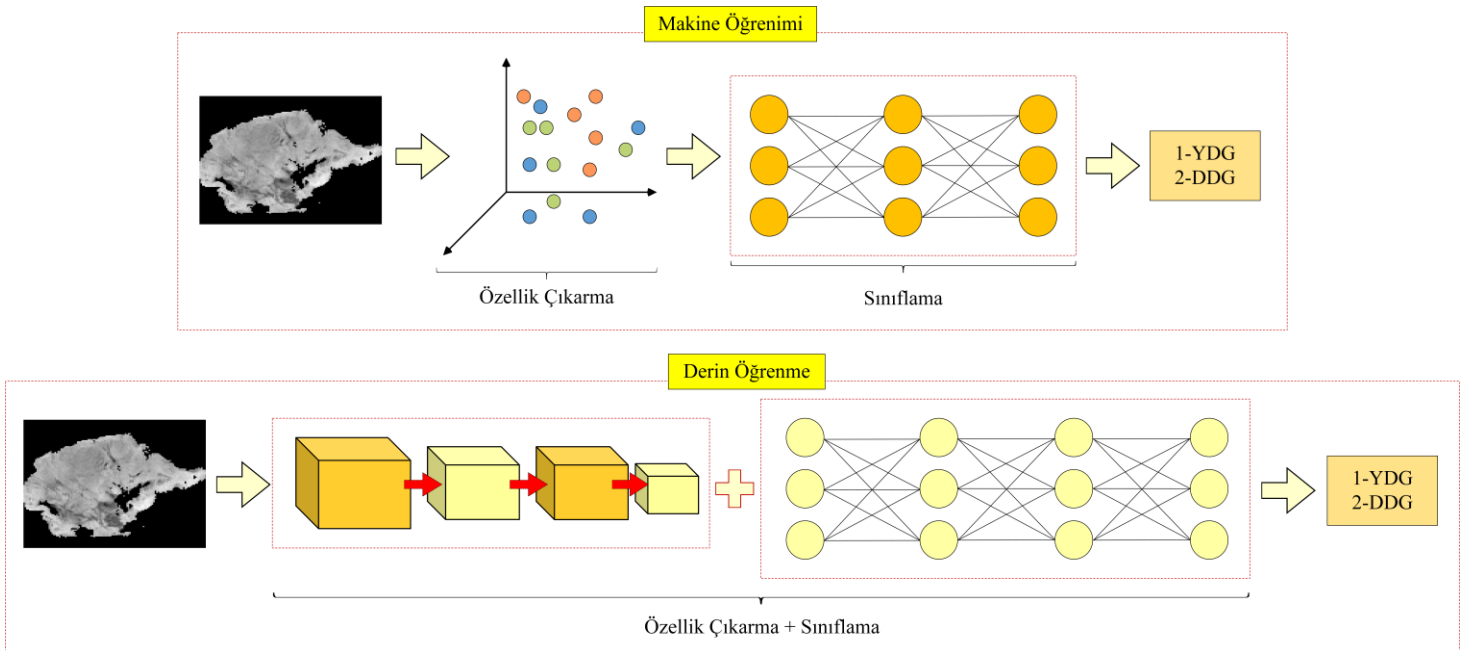
- Adım 1 kapsamında tümör hacmini barındıran 3B maske ile orijinal görüntü çarpılarak, yalnızca tümörü içeren hacmin eldesi gerçekleştirilir. Burada tam otomatik bir BDT sisteminin sınıflama modeline sunulacak girdi sağlanmaktadır. Ayrıca 3t2FTS yöntemi, BDT sisteminin segmentasyon bölümü olması halinde bu bölüme applike edilebilecek şekilde önerilmektedir. Çalışma kapsamında kullanılan maske BraTS 2017/2018 kütüphanesinde mevcuttur.
- Adım 2 kapsamında tekil tip 3B sekans verisi ele alınmakta ve 6 farklı BDİ özelliği (ortalama, standart sapma, eğrilik, basıklık, enerji, entropi) her bir 2B görüntü diliminden elde edilmektedir. Bu işlem neticesinde 6×155 boyutlu bir matris üretilecektir. Burada gerekli istatistikler her bir görüntünün histogramı üzerinden sağlanmaktadır.
- Adım 3 kapsamında önceki adımda uygulanan işlem her bir sekans için tekrar edilmektedir. Böylelikle T1, T2, T1c ve FLAIR sekansları üzerinden 4 farklı 6×155 boyutlu matrislerin eldesi sağlanır.
- Adım 4 kapsamında önceki adım çıktısı 4 matris sıralı olarak birleştirilmekte ve tümörler için kimlik değeri taşıyacak olan 2B-ID görüntülerin üretilmesi sağlanmaktadır. Sonuç olarak imge boyutu, BraTS 2017/2018 kütüphanesinde her bir görüntü 155 dilim içerdiğinden 24×155 olarak elde edilmektedir.

Şekil 3.2 kapsamında adım 4’de belirtilen 2B-ID görüntüler, her bir tümör için bir barkod niteliği taşır. Burada tümöre dair lokasyon, şekil, boyut ve yoğunluk bilgileri istatistiksel analiz ile 3B uzaydan 2B uzaya dönüştürülmektedir. Elde edilen 2B-ID görüntüler; yalnızca bahsi geçen tümör tipleri için değil, aynı zamanda 3B MR görüntülerindeki farklı tip beyin tümörlerinin değerlendirilmesinde de kullanılabilir durumdadır. Ayrıca 3t2FTS yönteminin, bütün bir beyin dokusuna applike edilmesi ile farklı hastalıkların sınıflandırılması da gerçekleştirilebilir. Bu nedenle 3t2FTS yaklaşımı, 3B MR görüntüleri üzerinden farklı hastalık ve tümör türlerine uyarlanabilecek şekilde önerilmektedir. Burada metodun, farklı BDT sınıflama tasarımlarının uygulamasına açık bir yapı teşkil ettiği görülmektedir.

3.3. Derin Öğrenme

Derin öğrenme, insan beyninin merkezinden ilham alan *makine öğreniminin* bir alt kümesidir. Derin öğrenmede insanlar tarafından önceden belirlenmiş kurallara ihtiyaç yoktur. Derin öğrenme yöntemleri daha spesifik muhakeme kuralları yerine, etiketlenmiş veriler arasındaki ilişki için büyük miktarda veri gerektirir. Derin öğrenme, her biri bir önceki katmanın çıktısından derlenen ve daha sonra bu anlatıyı elde etmek için farklı şekilde yorumlanan çok sayıda katmanı işletmektedir (Lu ve ark., 2015; Gupta ve ark., 2022).

Derin öğrenmede sınıflandırma için önemli olan özellikler derin sinir ağları sayesinde otomatik öğrenilirken, makine öğrenmede bu özelliklerin manuel olarak tanımlanması (spesifik yöntem işletilerek eldesi) gerekir. Derin öğrenme algoritmaları, özellik hiyerarşisi adı verilen bir yöntemle, birden çok görüntü özellik katmanını keşfeder. Her katmanın öznelikleri bir önceki katmanın temsilinden hesaplanır ve öznelikleri düşükten yükseğe doğru yavaş yavaş öğrenilir. Makine öğrenme algoritmaları kararlar veya tahminler yapmak için verilerden bilgi çıkarırken, derin öğrenme algoritmaları, öğrenmeyi artımlı ve kümülatif hale getirmek için veri katmanlarıyla etkileşime girer. Şekil 3.3 kapsamında makine öğrenme ve derin öğrenme için, tümör görüntü sınıflaması üzerine uygulanan genel adımlar sunulmaktadır (Lu ve ark., 2015; Gupta ve ark., 2022).



Şekil 3.3. Makine öğrenmesi ve derin öğrenme yöntemlerinde görüntü sınıflama örneği

Derin öğrenme algoritmaları genel olarak iki amaç için kullanılmaktadır. Bu amaçlar; özellik çıkarma ve transfer öğrenmedir. Özellik çıkarımı aşamasında önceden eğitilmiş yapı yeniden eğitilir ve probleme göre sınıflama sağlanır. Transfer öğrenmede ise önceden eğitilen ağın son üç katmanı kaldırılmakta / revize edilmekte ve soruna uygun bir çözüm bulmak için yeni katmanlar eklenebilmektedir (Lu ve ark., 2015; Gupta ve ark., 2022).

3.4. Transfer Öğrenme

Transfer öğrenme, sınıflama ve özellik çıkarımı için bir alanda daha önce öğrenilen bilgilerin başka bir alana aktarılması olarak tanımlanır. Hedef veri kümesi sınırlı miktarda etiketli veriye sahip olduğunda, belleğin üstesinden gelmek için etkili bir araç vazifesi görürler. Transfer öğrenme temelli yaklaşımlar, makine öğrenimi tabanlı yöntemlerden farklı olarak bir görev için, başka bir görevle iş gerçekleştirmiş olan yapı bilgisini kullanarak öğrenme ve başarıyı artırma amacı taşırlar. Diğer bir deyişle, daha önce büyük bir veri kümesinde eğitilmiş derin bir ESA mimarisi kullanılarak yeni görevin gerçekleştirilmesi, transfer öğrenmenin temel mantığını teşkil eder. Daha az eğitim görüntüsü içeren yeni bir veri kümesinde, önceden eğitilmiş bir ESA kullanılır ise yeni veri, önceden eğitilmiş veri kümelerinden daha fazla eğitilecektir. Bu işleme ince ayar (*fine tuning*) denilmektedir (Cireşan ve ark., 2012; Yosinski ve ark., 2014; Orenstein ve Beijbom, 2017; Krizhevsky ve ark., 2017).

Transfer öğrenme algoritmaları son zamanlarda birçok görev çözümünde ve derin öğrenme uygulamalarında kullanılmaktadır. Bunun sebebi ise önceden eğitilmiş bir ESA mimarisine dair ince ayar yapılmasının, rastgele ağırlıklarla başlatılmış bir ESA mimarisinin eğitilmesinden çok daha hızlı ve kolay olmasıdır. Bu noktada derin ESA yapılarının sıfırdan eğitilmesi; yüksek hesaplama gücü, geniş bellek kaynakları ve fazlaca işlem zamanı gerektirmektedir. Bir transfer öğrenme mimarisinin performans değerlendirmeleri, genel olarak önceden eğitilmiş ağırlıklı ağların hata oranları ile rastgele başlatılmış ağların hata oranları karşılaştırılarak sağlanır (Cireşan ve ark., 2012; Yosinski ve ark., 2014; Orenstein ve Beijbom, 2017; Krizhevsky ve ark., 2017).

ESA mimarileri kapsamında ilk katmanlar; kenarlar, eğriler, köşeler ve renk bölgeleri gibi özelliklerin öğrenilmesini sağlarken, son katmanlar daha soyut ve belirli özelliklerin teminini gerçekleştirir. Transfer öğrenmede önceden eğitilmiş ESA

mimarisine dair son üç katman (tam bağlı katman, *softmax* katmanı, sınıflandırma katmanı) kaldırılmakta ve kalan katmanlar yeni sınıflama görevinde aktif olarak kullanılmaktadır (Cireşan ve ark., 2012; Yosinski ve ark., 2014; Orenstein ve Beijbom, 2017; Krizhevsky ve ark., 2017).

Şekil 3.4 kapsamında; transfer öğrenmenin medikal alandaki kullanımı ile ilgili olarak genel transfer öğrenme yöntemleri, yöntemlere dair kullanım amaçları, problemlerde kullanılan medikal görüntüleme teknikleri ve transfer öğrenmenin uygulandığı medikal problem konuları (üzerinde çalışılan organlar) bir şema üzerinden sunulmaktadır (Kora ve ark., 2022).

Transfer Öğrenme	Evrışimli Sinir Ağları	Sınıflandırma	X-ray	Beyin
			Bilgisayarlı Tomografi	Retinal
			Manyetik Rezonans Görüntüleme	Toraks
	Yinelemeli Sinir Ağları	Segmentasyon	Manyetik Rezonans Görüntüleme	Kalp
			Pozitron Emisyon Tomografisi	Göğüs
			Ultrasonografi	Akciğer
Çekişmeli Üretici Ağlar	Segmentasyon	Pozitron Emisyon Tomografisi	Abdomen	
		Ultrasonografi	Prostat	
			Ultrasonografi	Deri

Şekil 3.4. Transfer öğrenmenin medikal alandaki kullanımı (Kora ve ark., 2022)

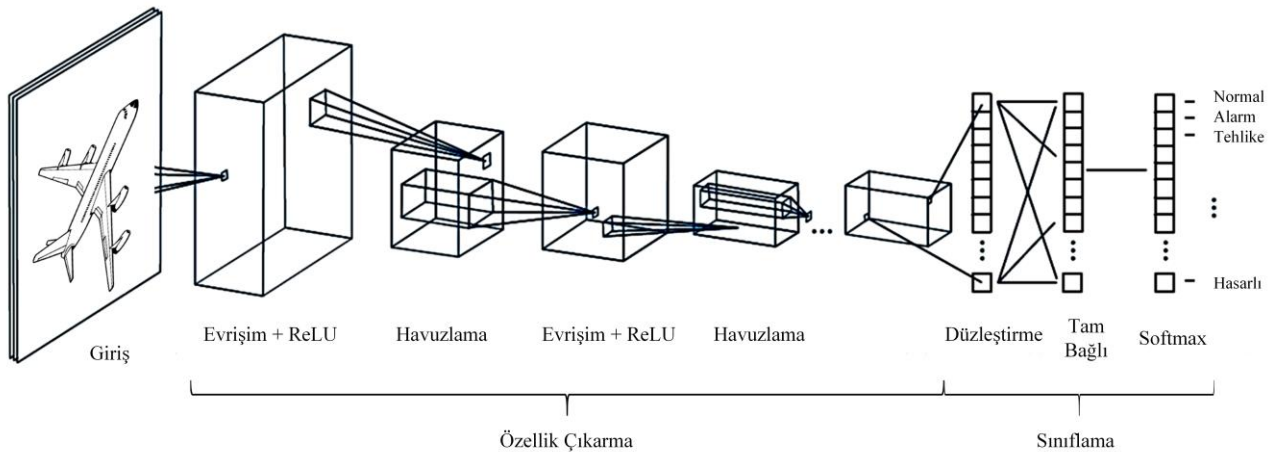
3.5. Evrişimsel Sinir Ağları

Evrışimsel Sinir Ağları (ESA), kendi kendini optimize etmeyi öğrenen bir sinir ağına benzemektedir. ESA mimarisi, klasik bir sinir ağında katman sayısının artışı ile meydana gelebilecek hesaplama probleminin üstesinden gelebilmektedir. Ancak ESA'ların problem çözümü için, yüksek işlem hızlı ve yüksek kapasiteli Grafik İşlem Birimi (GİB)'ne ihtiyaç duyulur. ESA mimarilerinin temeli ise giriş verisinde mevcut olan nitelikli bilgilerin konvensiyonel filtreler vasıtasıyla tespitine dayanır. ESA mimarisi kendi içinde YSA yapısı barındırmasına karşın YSA'dan farklı olduğu analiz, doğrudan görüntüdeki özelliklerden gerekli örüntü bilgisini çekebilmesidir. Oysaki YSA yapısına anlamlı örüntülerin sunumu gerekir. Bu noktada ESA mimarisi, filtrelerin kullanımı ile doğrudan görüntü üzerinde çalıştırılacak işleme sahiptir (O'Shea ve Nash, 2015).

ESA mimarisi, 3B düzeninde evrişim katmanlarına sahiptir. Burada evrişim katmanları yükseklik, genişlik ve derinlik temelli tanımlanır. Bu katmanların kullanımı ile ESA'nın görüntü işleme uygulamalarına kolaylıkla uygulanabilir olduğu bilinir. ESA kapsamında evrişim katmanları işletilerek öznitelik eldesi sağlanır ve bu elde işlemi birçok evrişim katmanı kullanıldığı durumda artırılır. Artırılan bu özellikler, hesaplama karmaşıklığının giderilmesi amacıyla sonraki aşamada faydalı azaltma işlemi ile azaltılır. Böylelikle mimarinin sonuç verme süresi azaltılmış ve üreteceği çıktı iyileştirilmiş olur (O'Shea ve Nash, 2015).

Bir ESA mimarisinde yer alan katmanlar genel olarak giriş, evrişim (*convolution*), doğrultulmuş doğrusal birim (*ReLU*), havuzlama (*pooling*), tam bağlı (*fully-connected*), seyreltme (*dropout*) ve sınıflama şeklinde sıralanmaktadır. ESA mimarisi eldesinde evrişim ve havuzlama katmanları peşi sıra eklenmekte, doğrusal birim katmanı ise havuzlama öncesi veya sonrasında eklenebilmektedir. Evrişim ve havuzlama katmanları kullanılarak, üst seviye özniteliklerin ardışık eldesi sağlanmaktadır. Temel sınıflama işlemi ise ESA mimarisindeki tam bağlı katmanda gerçekleştirilir. ESA içindeki filtrelerin değerleri eğitim işlemi boyunca otomatik öğrenilir. Bu kapsamda bir görüntünün ilk verildiği katmanda kenar varlığı / yokluğu analiz edilirken, ikinci katmanda basit şekillerin değerlendirilmesi ve tespiti, ilerleyen katmanlarda ise üst seviye (görev odaklı – spesifik) şekillerin tespiti ele alınmaktadır. Bu bağlamda mimariye dair son katman, söz konusu özelliklerin sınıflandırılması için kullanılır (Gu ve ark., 2018).

Şekil 3.5'te basit bir ESA yapısının katmanları gösterilerek, bir uçağa dair durumların ele alındığı örnek bir uyarılma sunulmaktadır (Tabian ve ark., 2019).



Şekil 3.5. Basit ESA yapısı ve uçak durum örneği (Tabian ve ark., 2019)

3.5.1. Giriş katmanı

Giriş katmanı ESA mimarisinin ilk katmanı olup, imgeler bu bölümde yapıya sunulmaktadır. ESA temelli transfer öğrenme algoritmalarının imge giriş boyutları sabit olduğundan, gerekli boyut düzenlemeleri yapılarak giriş katmanına veriler aktarılır. Geleneksel bir ESA yapısı ile tasarım yapılacağına; giriş görüntü boyutu büyük olduğunda, sistem içinde incelenecek özellik sayısı da boyutla orantılı olarak artacaktır. Bu durum, mimarinin başarısının artmasını sağlayabilmektedir. Diğer taraftan, özellik sayısındaki bu artış ağın cevap verme süresini de artırır. Geleneksel ESA mimarisine imgelerin boyutu küçültülerek de veri aktarımı sağlanabilmektedir. Yine bu durum özellik sayısında azalma meydana getireceğinden, ağın çalışma zamanı azalsa dahi başarısı da azalabilmektedir. Giriş katmanına sunulan veri, ESA mimarisinde katmanlar arası aktarılan bilgi ve mimarinin veri üzerindeki etkinliğine göre tanımlı kategorilerden birisine atanır (Tabian ve ark., 2019).

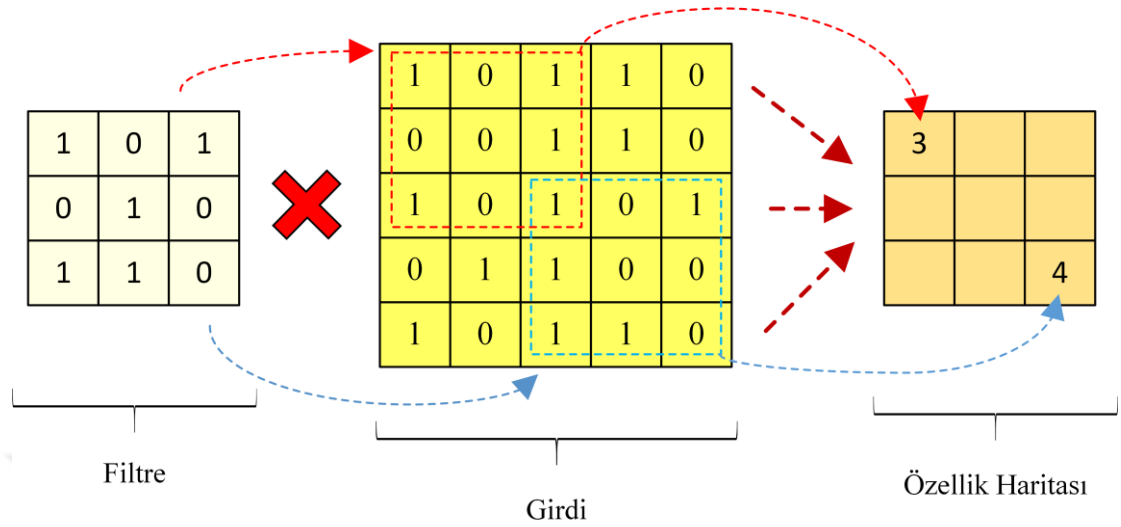
3.5.2. Evrişim katmanı

ESA yapısına adını veren ve mimarinin temelini teşkil eden katman evrişim katmanıdır. Evrişim katmanı kapsamında girişteki matris bilgisi alınmakta ve bir dizi filtre üzerinden işlenmektedir. Katmandaki her bir filtre, imge içi bazı özelliklerin etkinleşmesini sağlar. İlk evrişim katmanında kenar, köşe ve çizgi gibi alt özellikler seçilerek çıkarılırken, sonraki evrişim katmanları daha önemli özelliklerin analizini ve eldesini sağlar. Bu aşamada filtre, giriş matrisi üzerinde yatay ve dikey olarak kaydırılmak suretiyle girdi matrisi içindeki bilgiler filtre ile çarpılır. Çarpım neticesinde elde edilen değerler toplanır ve çıktı matrisine aktarılır. Burada gerçekleştirilen işlem *evrişim* olarak isimlendirilir (Tabian ve ark., 2019; Ofir ve Ben-Artzi, 2022).

Filtre boyutları mimari içinde değişkenlik gösterebilmektedir. Evrişim işlemi ile özellik haritası adı verilen bilgilerin temini sağlanır. Örneğin, ' $M \times M \times D$ ' boyutlu bir girdi için ' $j \times j \times D$ ' boyutunda bir filtre uygulandığında, elde edilecek özellik haritasının yatay ve dikey boyutları ' $(M-j+1) \times (M-j+1)$ ' olarak tanımlanacaktır (Tabian ve ark., 2019; Ofir ve Ben-Artzi, 2022).

Şekil 3.6 kapsamında tipik bir evrişim örneği sunulmaktadır. Burada girdi matrisi 5×5 ve uygulanan filtre 3×3 boyutlu iken, elde edilen özellik haritasının

yukarıda belirtilen kurala göre 3×3 boyutunda elde edildiği görülür (Tabian ve ark., 2019; Ofir ve Ben-Artzi, 2022).



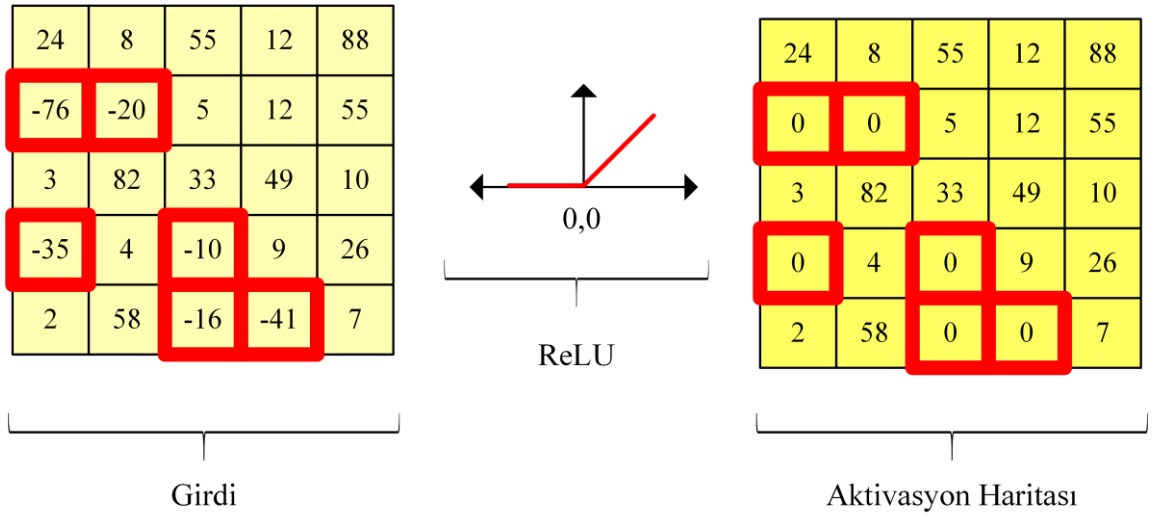
Şekil 3.6. Tipik evrişim örneği

3.5.3. Doğrultulmuş doğrusal birim katmanı

Doğrultulmuş doğrusal birim katmanı, genelde evrişim katmanından sonra işletilmekte ve bir aktivasyon fonksiyonu kullanımını sağladığından aktivasyon katmanı olarak da bilinmektedir. İngilizce kısaltması olan *ReLU* adıyla bilinen bu katman, girdi matrisine basit bir dönüşüm uygulayarak doğrultulmuş aktivasyon haritasını sağlar. Buradaki amaç, Eşitlik 3.10'da görüldüğü üzere girdi matrisindeki negatif değerlerin yerine sıfır atamalarının yapılmasıdır (Hijazi ve ark., 2015).

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (3.10)$$

ReLU haricinde de kullanılan birçok aktivasyon katmanı bulunmaktadır. Ancak *ReLU*, ileri besleme ve geriye yayılım işlemlerini hızlı bir şekilde gerçekleştirebildiğinden çok sık tercih edilir. Şekil 3.7 kapsamında tipik bir *ReLU* uygulama örneği görülmektedir. Burada girdi bilgisi içinde negatif değerlerin elimine edildiği ve pozitif sayıların değiştirilmeden aktarıldığı görülmektedir (Hijazi ve ark., 2015).



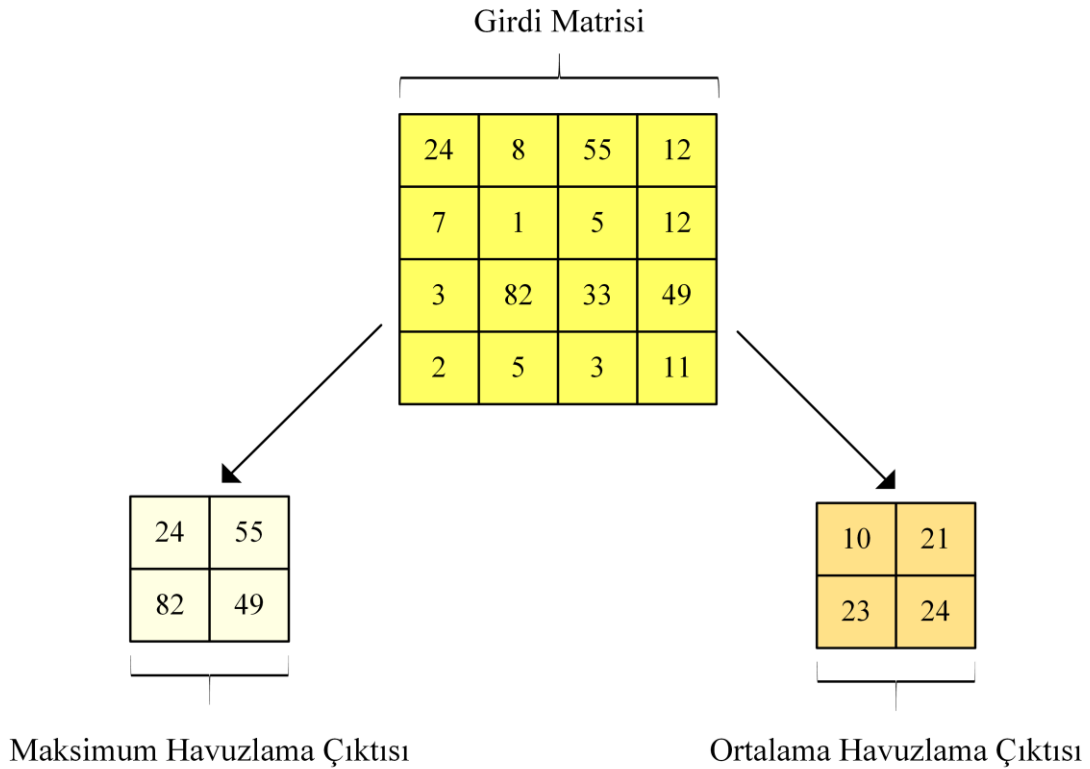
Şekil 3.7. ReLU fonksiyonunun matris üzerindeki etkisi

3.5.4. Havuzlama katmanı

Havuzlama katmanı, genelde doğrultulmuş doğrusal birim katmanı veya aktivasyon katmanı sonrasında kullanılmaktadır. Havuzlamanın en temel işlevi ise bir görüntüde, farklı boyutlara veya özelliklere sahip alt görüntüleri tanımdır (Hijazi ve ark., 2015).

Alt (aşağı) örnekleme adı verilen havuzlama işlemi kapsamında; belirli boyuttaki bir filtre, girdi matrisi üzerinde gezdirilerek girdi matris boyutundan daha küçük boyutta bir bilgi matrisi üretilir. Burada gerçekleştirilen boyut azaltımı faydalı bir azaltmadır ve bu işlem girdi bilgisinin ESA içinde daha hızlı analizine olanak sağlar. Bu aşamada en sık kullanılan filtre tipleri ise maksimum ve havuzlama olarak sıralanır. Bu filtreler sırasıyla girdi matrisi içerisinde filtrenin eşleştiği bölümdeki en yüksek değeri veya filtre-girdi matrisi eşleşmesinin ortalamasını almaktadır (Hijazi ve ark., 2015).

Şekil 3.8 kapsamında bir maksimum ve ortalama havuzlama örneği sunulmaktadır. Burada 4×4 boyutlu bir girdi matrisine, 2×2 'lik maksimum ve ortalama havuzlama uygulanmaktadır. Söz konusu havuzlama yöntemleri dışında toplam havuzlama gibi türler de mevcuttur. Bu aşamada ESA içi en sık kullanılan havuzlama tipi ise maksimum havuzlama olarak bilinir (Hijazi ve ark., 2015).



Şekil 3.8. Maksimum ve ortalama havuzlama örneği

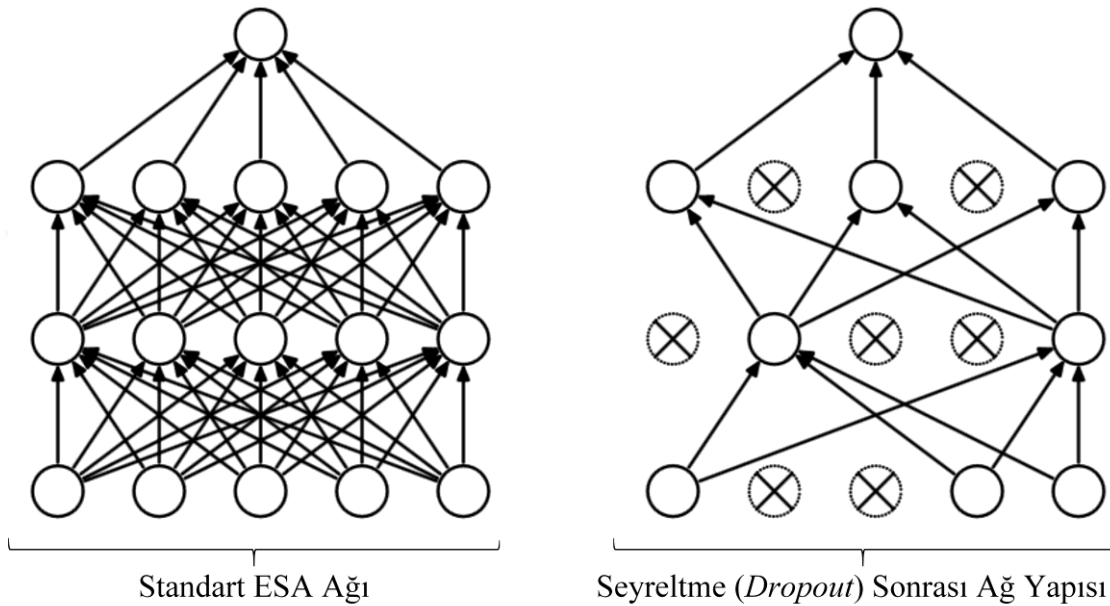
3.5.5. Tam bağlantılı katman

ESA mimarisinde sınıflama biriminin temeli tam bağlı katman olarak bilinir. Bu katman öncesi ESA yapısı özellik çıkarma işlemini gerçekleştirirken, sınıflamada kullanılacak bilgi bu katmanın girişine sunulur. Tam bağlı katman, havuzlama katmanı sonrasında gelir. Katman içinde birbirine bağlı nöronlar YSA tekniği sağlanacak biçimde dizilidir. Ancak bu katmanda, gizli katman sayısı basit bir YSA'dan daha fazla sayıda seçilebilmektedir. Evrişim ve havuzlama katmanlarından sonra elde edilen üst seviye özellikler, bu katman ile eğitim veri seti üzerinden gerekli çıkışın eldesinde kullanılır (Hijazi ve ark., 2015; Özkan ve Ülker, 2017).

3.5.6. Seyreltme katmanı

ESA mimarileri geleneksel sinir ağlarından daha fazla katmana sahiptir. Bu katmanlarda işlenen veri sayısı da katman sayısı ile birlikte artmaktadır. Bu durum ise mimariye ek bir hesaplama yükü oluşturur. Diğer taraftan, büyük miktarda veri işlendiği durumda da mimarinin veriyi ezberlemesi gibi dezavantajlar meydana gelebilmektedir.

Burada ağ içi bazı düğümlerin egale edilmesi ile hem hesaplama yükünün azaltılması sağlanabilmekte hem de ağın veriyi ezberlemesinin önüne geçilebilmektedir. Nöron eliminasyonu ise seyreltme işlemi olarak nitelenir. Burada eliminasyon rasgele bir şekilde gerçekleştirilir. Şekil 3.9 kapsamında klasik ESA ağı ve rasgele seyreltme sonrası ağ yapıları bir örnek üzerinden sunulmaktadır. (Özkan ve Ülker, 2017; Apaydin ve ark., 2020).



Şekil 3.9. Evrişimsel sinir ağına seyreltme uygulaması (Apaydin ve ark., 2020)

3.5.7. Sınıflandırma katmanı

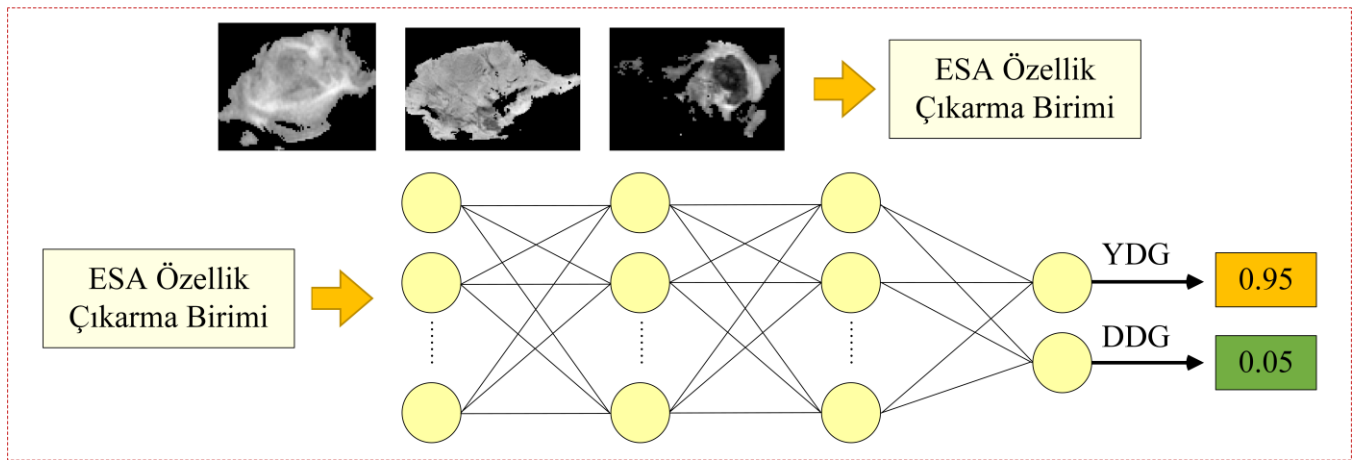
ESA kapsamında sonuç verme, son katman olan sınıflandırma katmanında sağlanır. Bu katmanda, farklı fonksiyonlar üzerinden bütün mimarinin cevabı belirlenebilmektedir. Önceki katmanlarda çıkarılan bilgiler, tam bağlı ve seyreltme katmanlarından sonra bu katman içerisinde kategorize edilir. Bu amaçla en sık kullanılan fonksiyon ise Eşitlik 3.11’de ifade edilen *Softmax* fonksiyonu olarak bilinir (Es-Sabery ve ark., 2021).

$$f(x) = \frac{e_i^x}{\sum_{k=1}^K e_i^x} \quad (3.11)$$

Eşitlik (3.11)’de; x , e_i^x ve K sırasıyla giriş verisini, veriye dair standart exponansiyel çıkışı ve veri setindeki sınıf sayısını simgelemektedir. *Softmax*

fonksiyonu, normalize exponansiyel fonksiyonu olarak da adlandırılır (Es-Sabery ve ark., 2021).

Softmax katmanında sınıf sayısı kadar çıkış mevcuttur ve bu çıkışlardan en yüksek olasılığa sahip sınıf, giriş verisi için tahmin edilen sınıf olarak atanır (Şekil 3.10). *Softmax* katmanında ağ yapısından gelen nümerik girdi işlenerek, kategorizasyon işleminin sağlanması için olasılığa çevrilir. Her bir sınıfa dair olasılık değerleri $[0,1]$ aralığında atanmakta ve bütün sınıflar için olasılıkların toplamı '1' olmaktadır (Hijazi ve ark., 2015; Bhandare ve Kaur, 2021; Es-Sabery ve ark., 2021).



Şekil 3.10. Gliyom görüntüleri ve ESA üzerinden Softmax fonksiyonu çalışma örneği

3.5.8. Optimizasyon teknikleri

Derin öğrenme yaklaşımlarının temel amaçlarından biri; mimarinin çalışması neticesinde tahmin edilen çıktı ile olması gereken (gerçek) çıktı arasındaki farkı, yani hatayı olabildiğince minimize ederek kategorizasyon sağlamaktır. ESA mimarisinde optimizasyon teknikleri işletilerek, YSA ağırlıkları için uygun değer tespiti sağlanır. Böylelikle hata (kayıp fonksiyonu) olabildiğince düşük tutulmaya çalışılır. Gradyan iniş yaklaşımı, hatanın azaltılmasında en sık kullanılan yöntemdir ve genelde üç başlıkta incelenir: 1-Toplu gradyan inişi, 2-Stokastik gradyan inişi, 3-Mini toplu gradyan inişi. Burada *sgdm*, *rmsprop* ve *adam* algoritmaları, sık kullanılan gradyan inişi temelli optimizasyon tekniklerine örnek olarak verilebilir (Jiang ve ark., 2020).

Sgdm: Tüm parametreler için tek bir öğrenme oranı kullanan algoritmadır. Diğer optimizasyon algoritmaları ise parametrelere göre farklı öğrenme hızlarının işletilmesini

sağlayarak, ağ eğitimini iyileştirmeyi amaçlar ve optimize edilmiş kayıp fonksiyonuna otomatik uyum sağlarlar (Jiang ve ark., 2020).

Rmsprop: *Adagrad* algoritması kapsamında öğrenme katsayısı zamanla azalır. Bu dezavantaja çözüm olarak *rmsprop* algoritması önerilmiştir. *Rmsprop* yaklaşımı; *sgd* algoritmasının uzantısı, *momentum* ve *adam* metotlarının temeli olması nedeniyle son yıllarda etkinlik kazanan uyarlamalı öğrenme yöntemleri arasında yer alır (Jiang ve ark., 2020).

Adam: Bu algoritma *rmsprop* yaklaşımına benzer bir şekilde, bir momentum terimi üzerinden parametre güncellemelerini sağlamaktadır. *Adam* yöntemi, *adagrad* ve *rmsprop* yöntemlerinin etkin özellikleri değerlendirilerek tasarlanmıştır (Jiang ve ark., 2020).

3.6. Transfer Öğrenme Yöntemleri

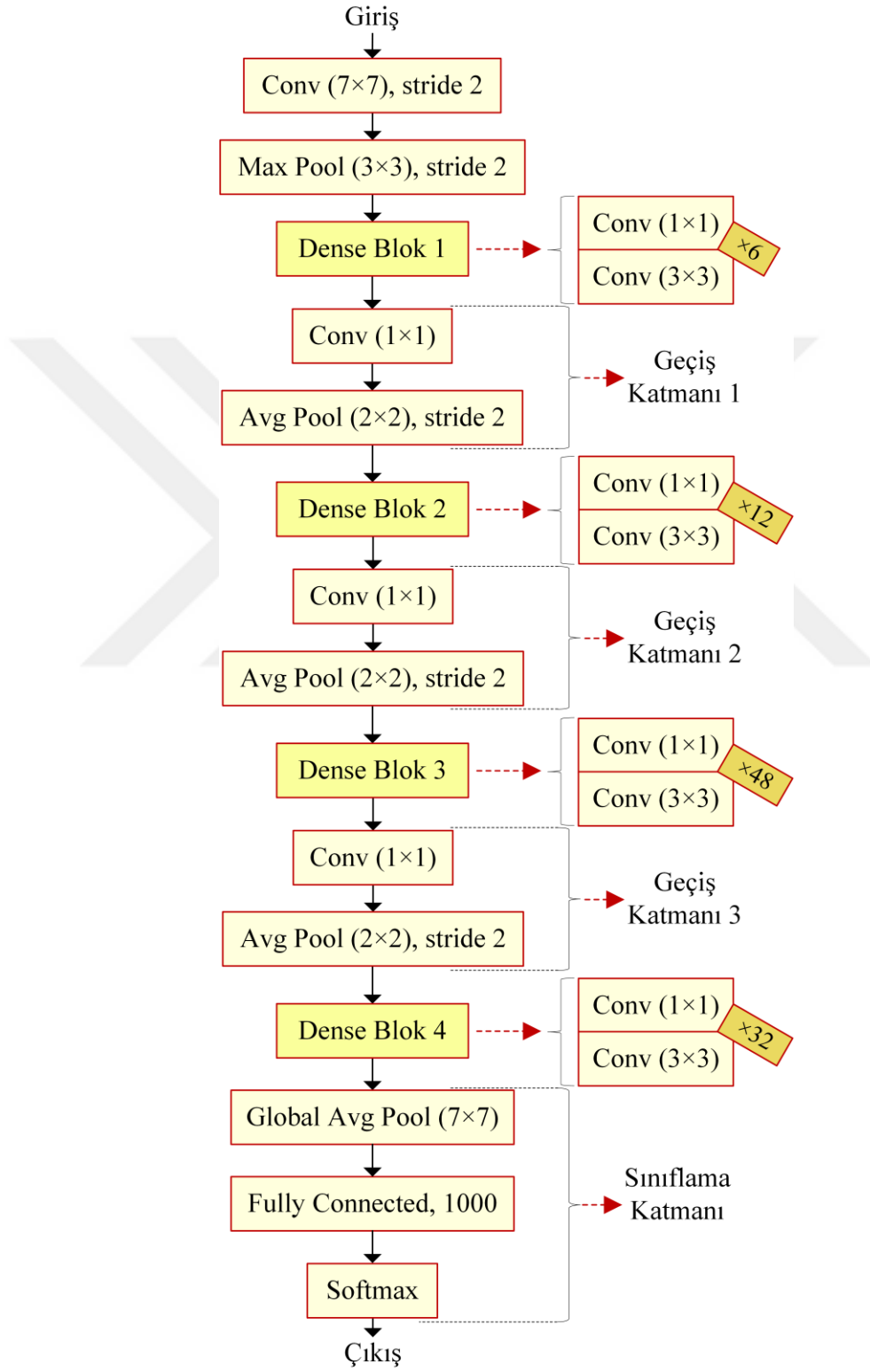
Tez çalışması kapsamında kullanılan 8 adet transfer öğrenme mimarisi bulunmaktadır. Bu bölüm kapsamında mimarilere dair görsel temsiller ayrıntılı bir biçimde ele alınmaktadır.

Mimarilerin gösterimlerinde ESA içi sınıflama aşaması için tam bağlı (*fully-connected*), seyreltme (*dropout*) veya bu iki işlemin de yer aldığı bloklar kullanılmaktadır. Bu üç durum da YSA tekniğine dair kullanımı temsil etmekte, bazı mimarilerde ise bu gösterimler yukarıda belirtilen üç durumdan biri ile ifade edilmektedir.

3.6.1. DenseNet201 mimarisi

DenseNet201 mimarisi, ileri beslemeli çalışma konsepti temelinde oluşturulan Yoğun Bağlantılı Evrişimli Ağlar (*DenseNet*)'in verimli bir versiyonudur. DenseNet, kaybolan gradyan problemini kolaylaştırır ve özellik yayılımını geliştirerek girdi verilerinin sonraki katmanlarda yeniden kullanımını sağlar. DenseNet; girdi verilerinin sistem tanınmasını yükseltmek ve önceki katmanlarda oluşturulan özellik haritalarının sonraki katmanlara bir girdi olarak beslenmesi için, yoğun (*dense*) blokların kullanımını temel almaktadır. Özetle DenseNet tabanlı mimariler, üç iyileştirme stratejisi

(özelliklerin yeniden kullanımı, kısa bağlantıların kullanımı, daha derin mimari kullanımı) üzerinden başarılı sonuçların eldesini amaçlar. DenseNet201 mimarisi Şekil 3.11’de görüldüğü gibidir (Huang ve ark., 2017; Jaiswal ve ark., 2021).



Şekil 3.11. DenseNet201 mimarisi

DenseNet201 mimarisi giriş görüntüsünü değerlendirmek için evrişim katmanlarını, çoklu bağlantılı evrişim katmanlarını içeren yoğun (*dense*) blokları, ortalama (*average*) ve maksimum (*max*) havuzlama yaklaşımlarını, tam bağlı katmanı ve *softmax* fonksiyonunu kullanmaktadır. DenseNet201 mimarisinde geçiş (*transition*) katmanı; başka bir yoğun (*dense*) bloğa beslenen ve gerekli azaltılmış bilgiye sahip özellik haritasını tanımlamak için, evrişim katmanı ile ortalama havuzlama katmanını içerir (Huang ve ark., 2017; Jaiswal ve ark., 2021).

3.6.2. InceptionResNetV2 mimarisi

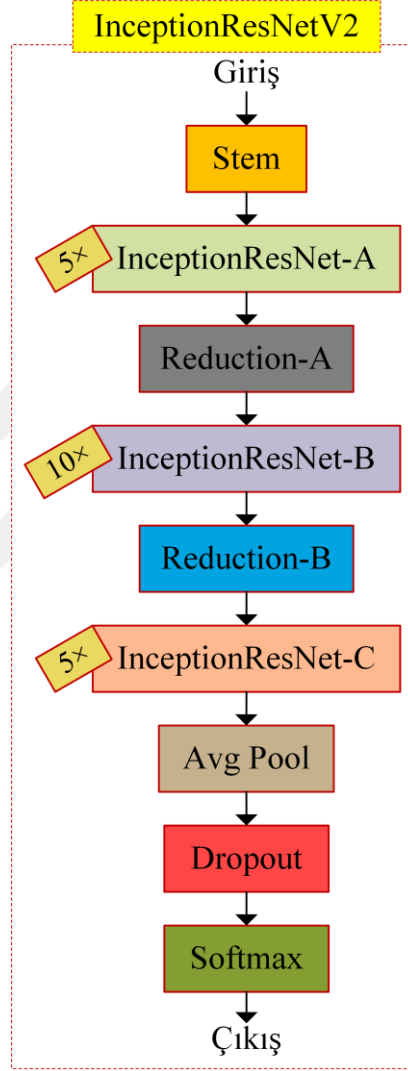
InceptionResNetV2, artık (*residual*) bağlantılar ve başlangıç (*inception*) fenomenlerinden oluşan hibrit bir mimari olarak bilinir. *Inception* modüllerini kullanan *inception* ağları; geleneksel ESA mimarilerinde yapı kaynaklı aşırı uyum (*overfitting*), düşük performans vb. sorunların üstesinden gelmek için önerilmiştir (Szegedy ve ark., 2017; Hassan ve ark., 2021).

Mimari kapsamında bahsi geçen artık bağlantılar, ağ yapısını kolaylaştırmak ve karar sürecini hızlandırmak için çok derin ağların tasarımında etkin bir şekilde kullanılmaktadır. Bu iki amaç çerçevesinde *InceptionResNet* temelli mimariler üretilmiştir (Szegedy ve ark., 2017; Hassan ve ark., 2021).

InceptionResNetV2 mimarisi; kök (*stem*) modülü, *InceptionResNet* tabanlı üç modülü (*InceptionResnet-A*, *InceptionResnet-B*, *InceptionResnet-C*), azaltma modüllerini (*Reduction-A*, *Reduction-B*), ortalama (*average*) havuzlama katmanını, seyreltme (*dropout*) katmanını ve *softmax* fonksiyonunu içermektedir. Şekil 3.12 ve Şekil 3.13 sırasıyla InceptionResNetV2 mimarisini ve mimari içi modülleri göstermektedir (Szegedy ve ark., 2017; Hassan ve ark., 2021).

InceptionResNetV2, çeşitli boyutlara sahip farklı verileri istiflemek için filtre birleştirmeyi (*filter-concat* veya *concat*) kullanır. Başka bir deyişle, üretilen çeşitli bilgileri birleştirmek için kullanılan filtre birleştirme ile katmanlar arasında hiçbir bilgi kaçırılmaz. Kök (*stem*) modülü, iş akışını hızlandırmak için giriş verilerinde alt örnekleme gerçekleştirir. Bu durum ise tüm mimarinin bellek kullanımını ve hesaplama maliyetini azaltmaktadır. Üç *InceptionResNet* tabanlı modül, hesaplama karmaşıklığını azaltmak için çoklu-boyutlu çekirdekler (*kernel*) işletmektedir. Ayrıca artık (*residual*) konseptinin kullanımı ile giriş bilgisi dönüştürülerek modül çıkışına aktarılır. Böylelikle

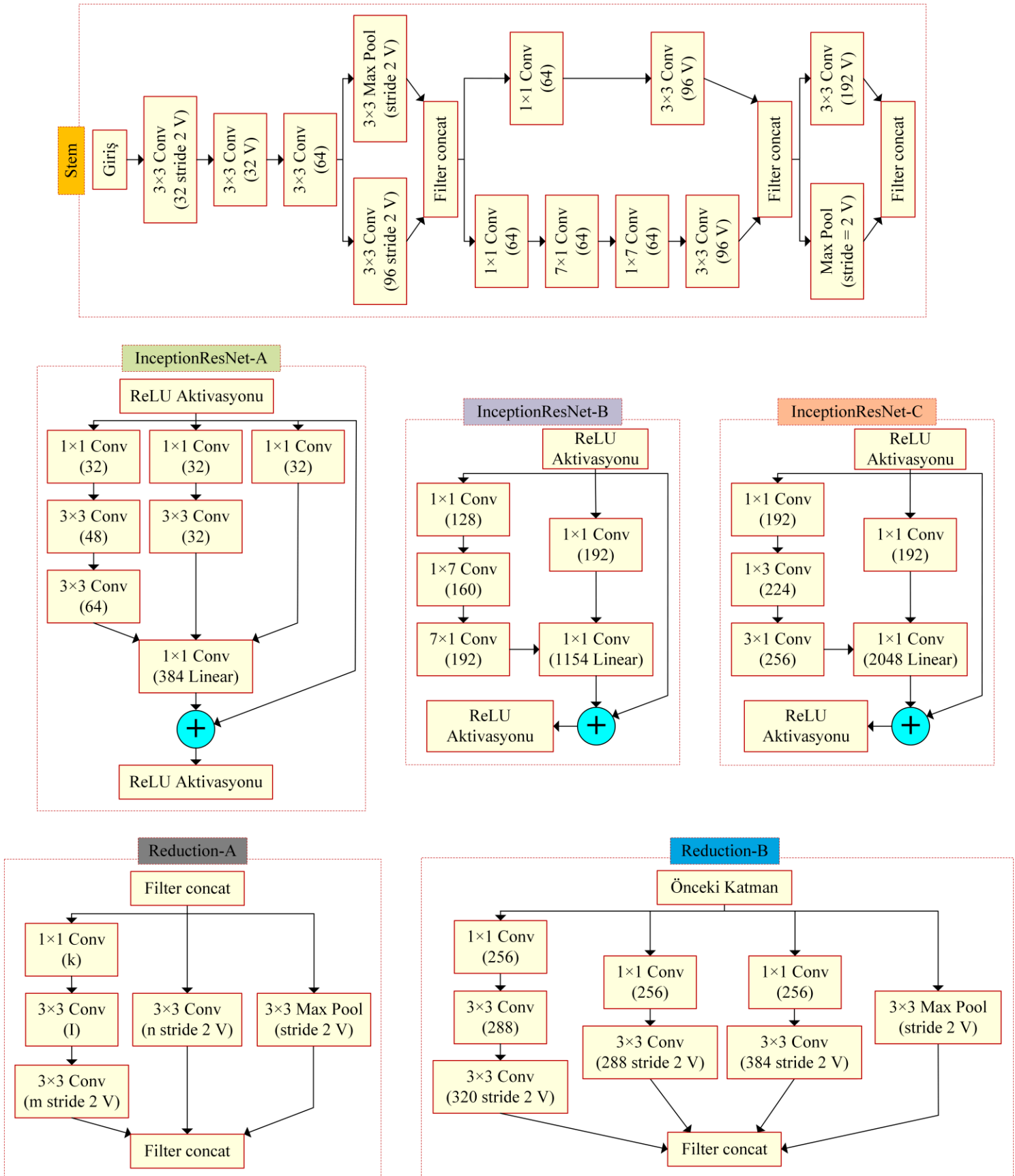
mimari içinde daha fazla bilginin modül çıkışına aktarımı sağlanır. Diğer taraftan artık (*residual*) akış, modül çıktısını iyileştirmenin yanı sıra eğitim sonrası gerekli bilgilerin iletilmediği durumlar için bir by-pass görevi görür. Azaltma (*reduction*) modülleri ise bilgi kaybını önleyerek verinin boyutsal indirgemelerini sağlamaktadır (Szegedy ve ark., 2017; Hassan ve ark., 2021).



Şekil 3.12. InceptionResNetV2 mimarisi

3.6.3. InceptionV3 mimarisi

InceptionV3, kendisinden esinlenerek oluşturulan InceptionResNetV2'nin az gelişmiş bir versiyonudur. InceptionV3 mimarisi içinde tekrarlanan artık (*residual*) bloklar, InceptionResNetV2'deki artık bloklara göre sıkıştırılmıştır durumdadır (Szegedy ve ark., 2016; Mahdianpari ve ark., 2018; Al Husaini ve ark., 2022).



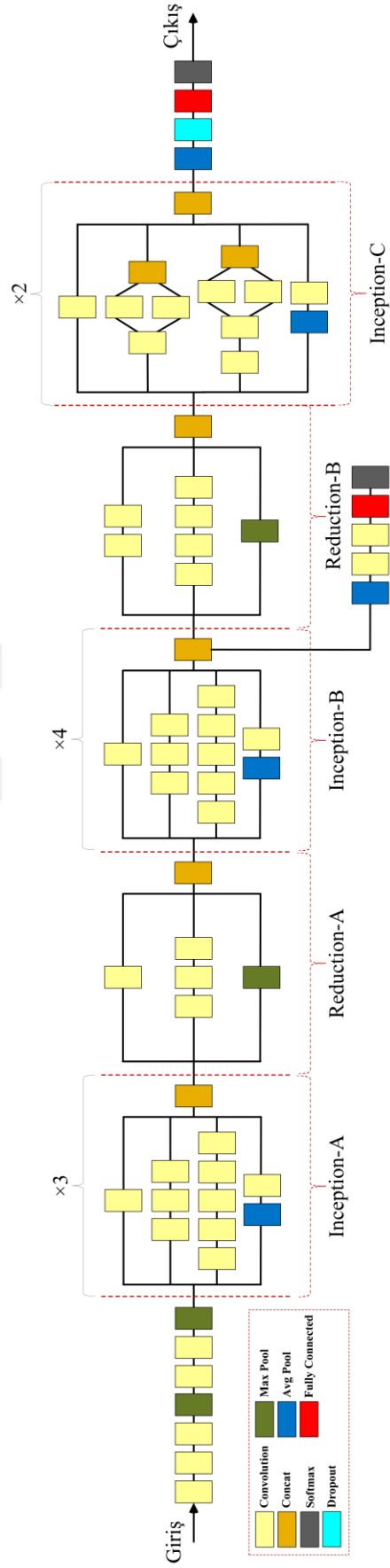
Şekil 3.13. InceptionResNetV2 modülleri

InceptionV3 mimarisi kapsamında üç başlangıç (*inception*) modülü (*Inception-A*, *Inception-B*, *Inception-C*), iki azaltma (*reduction*) modülü (*Reduction-A*, *Reduction-B*), ortalama (*average*) ve maksimum (*max*) havuzlama katmanları, seyreltme katmanı, tam bağlı katman ve *softmax* fonksiyonu kullanılmaktadır (Szegedy ve ark., 2016; Mahdianpari ve ark., 2018; Al Husaini ve ark., 2022).

Detaylı bir inceleme yapıldığında InceptionV3'ün, InceptionResNetV2 mimarisinden hem kullanılan blokların içyapısı hem de modüllerin kullanım sayıları temelli farklılık gösterdiği görülür. Şekil 3.14'de InceptionV3 mimarisinin şematik yapısı sunulmaktadır (Szegedy ve ark., 2016; Mahdianpari ve ark., 2018; Al Husaini ve ark., 2022).

InceptionV3 mimarisi, InceptionV1 ve InceptionV2 mimarilerinin gelişmiş bir sürümü olarak önerilmektedir. InceptionV3 mimarisi, eski versiyonlara göre hesaplama maliyetlerini azaltmak için modüllerdeki evrişim çekirdek boyutlarını değiştirmekte, asimetrik evrişimler uygulamakta ve aksiller sınıflandırıcılar kullanılmaktadır. Verileri büyük bir çekirdek ile çoğaltmak yerine çekirdek boyutlarının küçültülmesi ve asimetrik evrişimlerin uygulanması, mimariden kaynaklı hesaplama maliyetlerini azaltmak için nitelikli bir düzenleme olarak ortaya çıkmaktadır. 1×1 çekirdekli darboğaz (*bottleneck*) katmanı; ana konvolüsyonlardan hemen önce kullanılarak, mimari kapsamında işletilen parametre sayısının azaltılması sağlanır. Bu nedenlerle InceptionV3, zaman karmaşıklığını optimize etmeyi temel amaç edinmeden InceptionV1 ve InceptionV2 yapılarından daha derin formda önerilmiştir (Szegedy ve ark., 2016; Mahdianpari ve ark., 2018; Al Husaini ve ark., 2022).

Inception-B katmanı sonunda filtre birleştirmenin çıktısı olan aksiller sınıflandırıcı, tüm mimarinin sonuna doğru daha iyi bilgi aktarımı ve yakınsama sağlamaktadır. Diğer bir deyişle, kaybolan gradyan problemini ortadan kaldırılmaya çalışılırken, mimari sonuna doğru gerekli gradyanların kullanımı (sistem performansının artırılması) aksiller sınıflandırıcı kullanılarak gerçekleştirilir. Bununla ilgili olarak aksiller sınıflandırıcılar, özellikle başlangıç (*inception*) temelli ESA yapılarında sıklıkla işletilmektedir (Szegedy ve ark., 2016; Mahdianpari ve ark., 2018; Al Husaini ve ark., 2022).



Şekil 3.14. InceptionV3 mimarisinin şematik görünümü

3.6.4. ResNet50 ve ResNet101 mimarileri

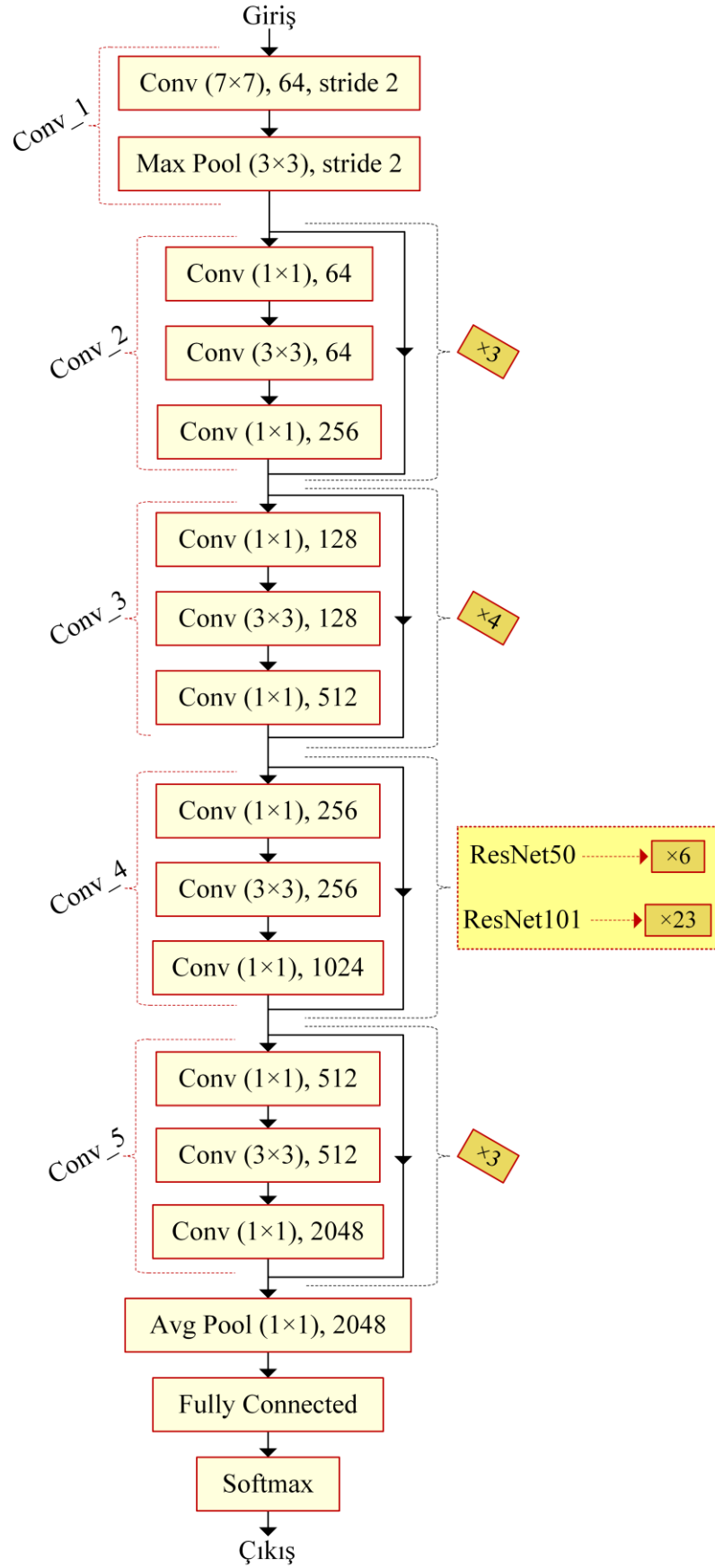
Katman sayısının artırılması genel olarak mimarinin doğruluğunu yükseltebilmektedir. Diğer taraftan ağlar ne kadar derin hale gelirse, mimarinin doğruluğu eğitim sürecinde daha da azalabilmektedir. Bununla ilgili olarak, daha derin ağlarda kaybolan gradyan problemini ortadan kaldırmak için ResNet50 ve ResNet101'i de içeren *ResNet* temelli mimariler önerilmiştir. Şekil 3.15 kapsamında ResNet50 ve ResNet101 mimarilerinin tasarımı aynı şema üzerinde gösterilmektedir (He ve ark., 2016; Kim ve ark., 2021; Rao ve ark., 2021).

Her iki mimari kapsamında da aynı çekirdek boyutuna sahip beş ana evrişim bloğu (*Conv_1, ..., Conv_5*), ortalama (*average*) ve maksimum (*max*) havuzlama katmanları, tam bağlı katman ve *softmax* fonksiyonu kullanılmaktadır. Burada mimariler arasındaki temel fark, her iki mimari için *Conv_4* bloğu kullanım sayısıdır (He ve ark., 2016; Kim ve ark., 2021; Rao ve ark., 2021).

Artık ağlar (*ResNet*) veya bu ağları içeren diğer mimariler (InceptionResNetV2 vb.), performansı yüksek seviyelerde tutmak ve derin katmanlarda bilgi kaybını önlemek için artık (*residual*) bağlantıları kullanmaktadır. Bu bağlantılar; *ResNet* tabanlı mimarilerde bazı katmanları atlayarak, bir katmanın çıktısını sonraki katmanların çıkışına beslemek için değerlendirilir. Ayrıca atlama bağlantılarının kullanımı ile mimariye ekstra bir parametre eklenmez. Burada artık bloklar, katmanlar arasındaki atlama bağlantıları aracılığıyla bypass işlemini gerçekleştirir. Bu durum ise mimarinin daha fazla eğitim hatası almasını önleyebilmektedir. Böylelikle katmanlar arası bilgi kaybı önlenerek, kaybolan gradyan probleminin minimum düzeyde kalması sağlanır (He ve ark., 2016; Kim ve ark., 2021; Rao ve ark., 2021).

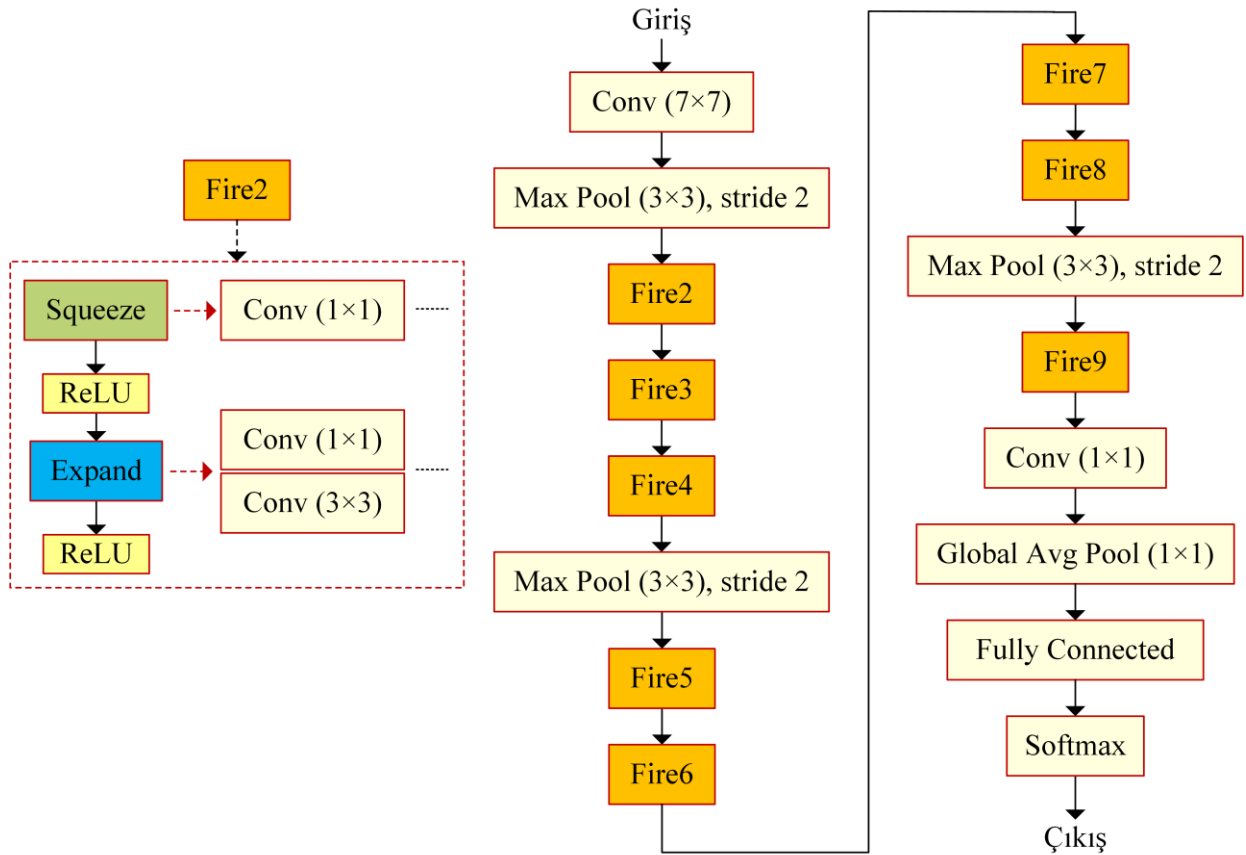
3.6.5. SqueezeNet mimarisi

SqueezeNet, AlexNet mimarisine rakip olarak önerilen etkin bir mimaridir. SqueezeNet mimarisinin üretimi esnasında yüksek doğruluklu başarı korunarak, AlexNet'ten daha az hesaplama karmaşıklığı ve daha az mimari parametresinin eldesi amaçlanmıştır (Iandola ve ark., 2016; Li ve ark., 2021).



Şekil 3.15. ResNet50 ve ResNet101 mimarileri

Şekil 3.16 kapsamında SqueezeNet mimarisi sunulmaktadır (Iandola ve ark., 2016; Li ve ark., 2021).

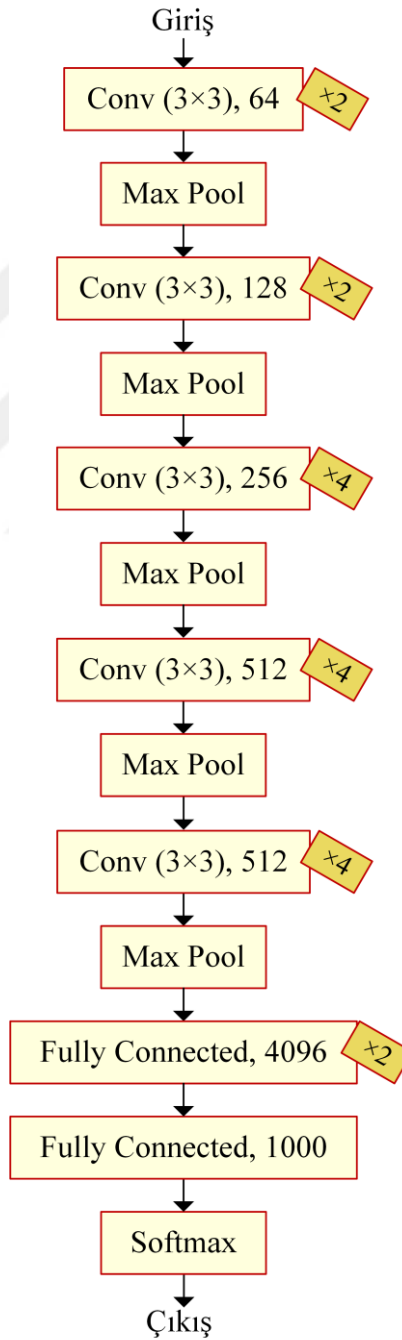


Şekil 3.16. SqueezeNet mimarisi

SqueezeNet mimarisi kapsamında evrişim katmanları, sıkıştırma (*squeeze*) ve genişletme (*expand*) bloklarını içeren ateş (*fire*) modülleri, ortalama (*average*) ve maksimum (*max*) havuzlama katmanları, tam bağlı katman ve *softmax* fonksiyonu kullanılmaktadır. SqueezeNet’de işletilen ateş modülleri, AlexNet’e kıyasla mimari parametrelerini azaltmak üzere ortaya çıkmıştır. Ateş modülündeki sıkıştırma (*squeeze*) ve genişletme (*expand*) blokları, sistem parametrelerini azaltmak için sırasıyla 1×1 ve 1×1 & 3×3 boyutlu evrişim filtrelerini kullanır. Ateş modüllerinin yapıda işletilmesi ile tüm mimarinin verimli bir şekilde istiflenmesi sağlanır. SqueezeNet mimarisi ateş modüllerine ek olarak, giriş kanallarının 3×3 boyuta düşürülmesi işlemi ile mimarinin hesaplama karmaşıklığını azaltmaktadır. Ayrıca mimarinin doğruluğunu en üst düzeye çıkarmak için, havuzlama sonrası alt örnekleme işlemi gerçekleştirilir (Iandola ve ark., 2016; Li ve ark., 2021).

3.6.6. VGG19 mimarisi

VGG19, yüksek doğruluğu koruyarak hesaplama karmaşıklığını ortadan kaldırmak için küçük evrişim çekirdeklerini kullanan derin bir ESA yapısıdır. VGG19, ağ derinliği ve bu derinliğin sistem çıkışı üzerine etkisinin incelenmesi ile önerilmiştir. Şekil 3.17’de VGG19 mimarisi sunulmaktadır (Simonyan ve Zisserman, 2014; Bansal ver ark., 2021).



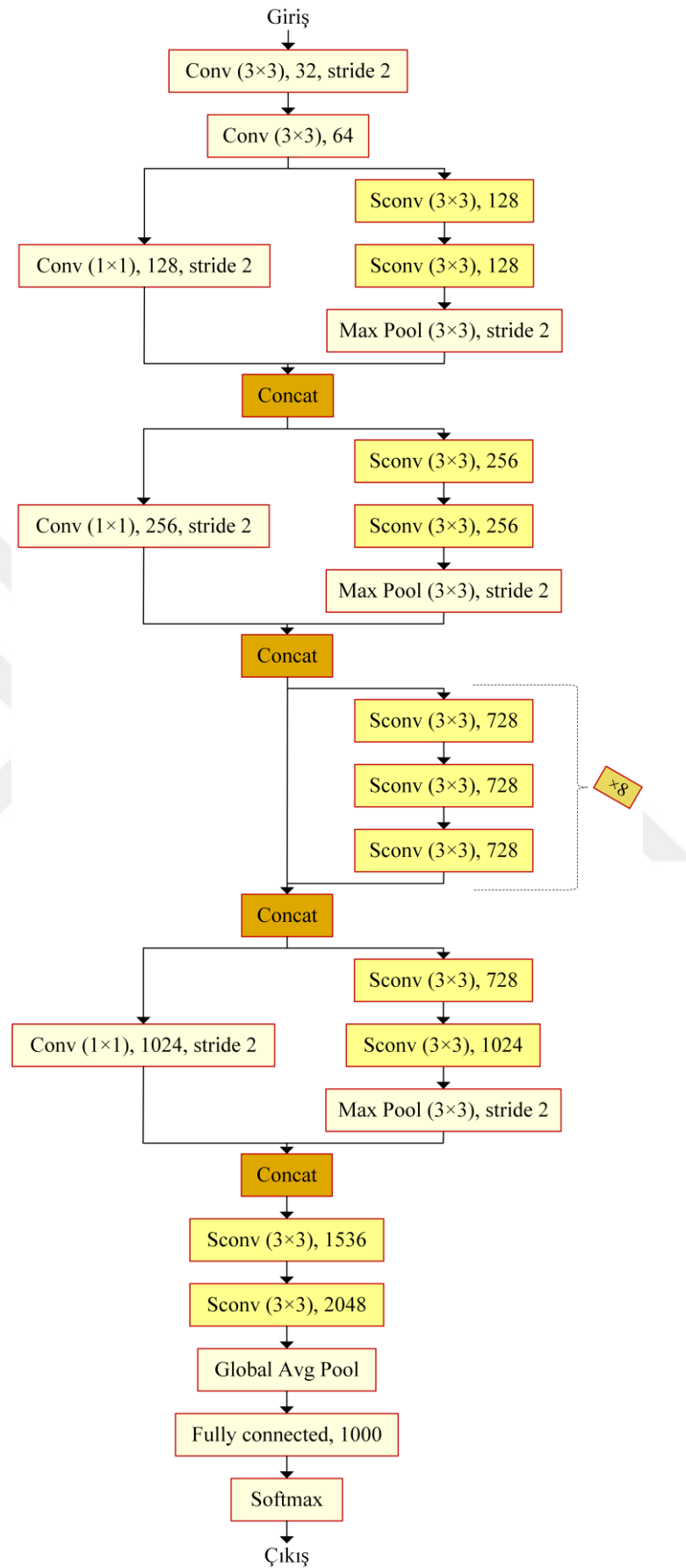
Şekil 3.17. VGG19 mimarisi

VGG19 mimarisi kapsamında evrişim katmanları, maksimum (*max*) havuzlama katmanları, tam bağlı katman ve *softmax* fonksiyonu kullanılmaktadır. Çoklu 3×3 evrişim çekirdeklerinin işletilmesi ile mimaride gerekli özellik haritalarının temini sağlanmaktadır. Mimarinin sınıflama biriminde, 3 gizli katmanlı – tam bağlı katman üzerinden verilerin yüksek doğrulukla sınıflandırılması amaçlanır. VGG19 mimarisinin karmaşıklığı ve derinliği göz önüne alındığında, 5 adet maksimum havuzlama katmanı kullanılarak mimaride daha az parametre kullanımı ve daha az hesaplama karmaşıklığı sağlanmaktadır (Simonyan ve Zisserman, 2014; Bansal ve ark., 2021).

3.6.7. Xception mimarisi

Aşırı başlangıç (*Extreme Inception: Xception*) mimarisi kapsamında evrişim katmanları, ayrılabilir evrişim (*Sconv*) blokları, atlama bağlantıları ve tüm mimarinin tutarlılık analizlerinin yanı sıra *inception* mimarisi ele alınarak tasarımı sağlanmıştır. Şekil 3.18’de Xception mimarisi sunulmaktadır (Chollet, 2017; Leonardo ve ark., 2018).

Xception mimarisi evrişim katmanları, iyileştirilmiş – derinlemesine ayrılabilir evrişim blokları (*Sconv*), filtre birleştirme (*filter-concat* veya *concat*), ortalama (*average*) ve maksimum (*max*) havuzlama katmanları, tam bağlı katman ve *softmax* fonksiyonunu içermektedir. Xception mimarisi başlangıç (*inception*) modülleri kullanmak yerine, derinlemesine ayrılabilir (*depth-wise separable*) evrişimler kullanılarak tasarlanmıştır. Bu sayede mimari, başlangıç (*inception*) temelli ağlarda olduğu gibi uzamsal ve kanallar arası korelasyonları ayrıştırmayı amaçlar. Derinlemesine ayrılabilir evrişim, uzamsal korelasyonu her kanal için ayrı ayrı haritalayan ve 1×1 derinlemesine (*depth-wise*) evrişim yoluyla kanallar arası korelasyonu yakalayan özel bir dönüşümdür. Burada Xception mimarisinde yer alan derinlemesine (*depth-wise*) evrişim, sistemde kullanılan iyileştirilmiş – derinlemesine ayrılabilir evrişimin ikinci bölümünü teşkil eder. İlk bölüm ise noktasal (*pointwise*) evrişim olarak bilinir. Şekil 3.18 kapsamında *Sconv* blokları, iyileştirilmiş – derinlemesine ayrılabilir evrişimi temsil etmektedir. Xception mimarisinde eğitim hatasını azaltmak için, *ResNet* temelli mimarilerde olduğu gibi atlama bağlantıları kullanılmaktadır (Chollet, 2017; Leonardo ve ark., 2018).



Şekil 3.18. Xception mimarisi

3.7. Kullanılan Veri Seti Özellikleri

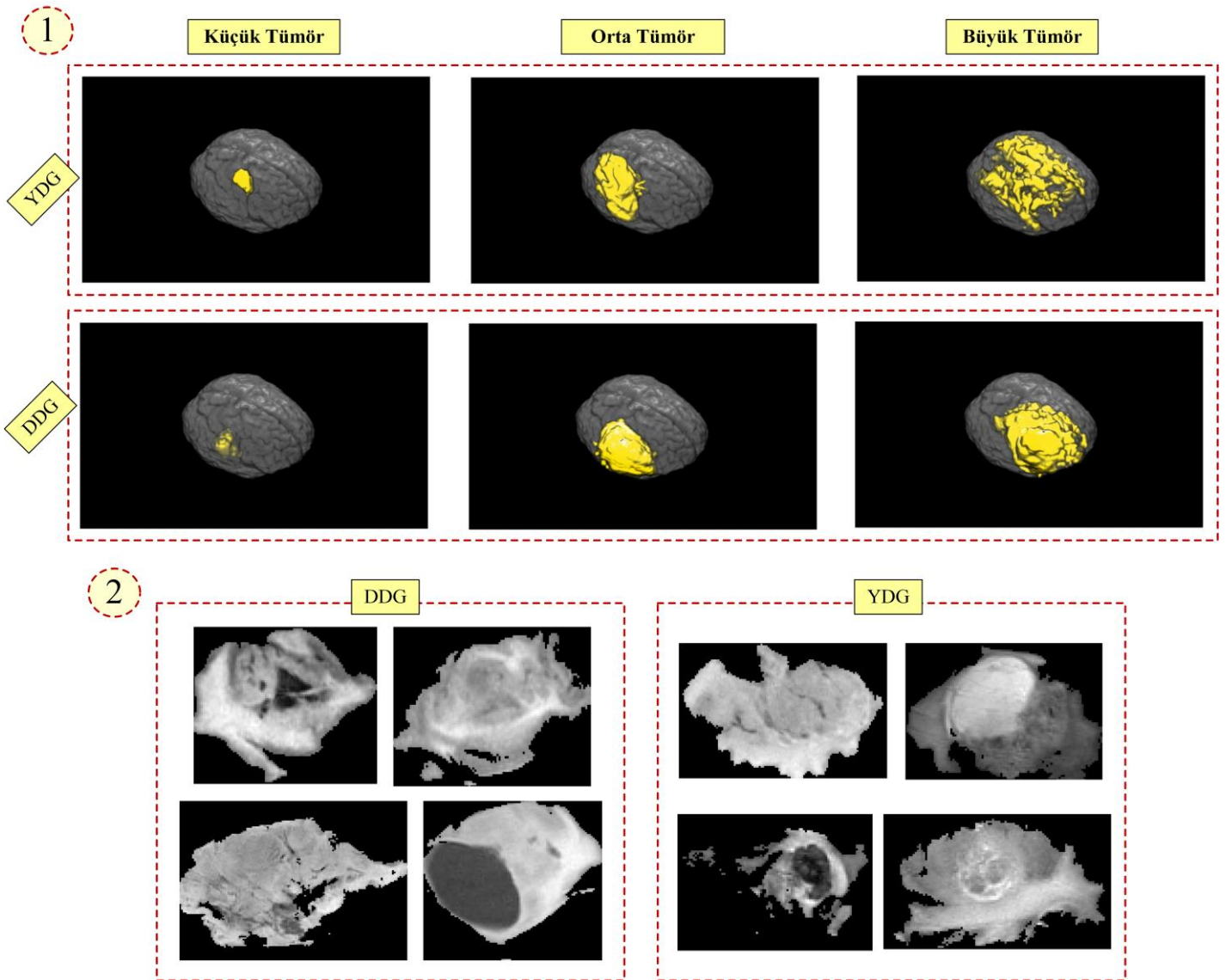
BraTS 2017/2018 kütüphanesi; eğitim veri seti kapsamında 210 YDG ve 75 DDG örneğine sahip 3B MR görüntülerinden oluşmaktadır. Her 3B görüntü 155 adet dilim içermekte olup, her dilim RGB alanı ile ilgili olarak yine 3B formatta tanımlıdır. Veri setinde her bir deneğe ait 3B görüntüler, dört farklı görüntüleme modalitesi / sekansı (T1, T2, T1c, FLAIR) kullanılarak elde edilmiştir. Modaliteler ve dilim sayısı ile ilgili olarak, bir hastanın tümör analizi için 620 dilim ele alınmaktadır. Bir dilimin görüntü boyutu ve kalınlığı sırasıyla 240×240 ve 1mm'dir (Menze ve ark., 2014; Bakas ve ark., 2017; Bakas ve ark., 2018).

Eğitim veri seti kapsamında *şablon* (4), *peritümöral* (2), *ilerlemeyen - ödem* (1) ve *arka plan* (0) olmak üzere 4 etiket üzerinden bir maske tanımlı olup, bu maske ile tümör alt bölgeleri normal beyin dokusundan, birbirinden ve arka plandan ayırtılabilmektedir. Tez çalışması kapsamında 1-2-4 etiketleri '1' ve 0 etiketi '0' olarak atanmış, bu sayede yalnızca tümörün yer aldığı 3B bilgi maske ile elde edilmiştir. Bu noktada önerilen sınıflama modelinin, tümör alt bölgeleri veya tüm tümörü çıkaran bir segmentasyon yöntemine adaptif olması gerektiğinden, yalnızca tüm tümör bölgesi ele alınmaktadır. Şekil 3.19 veri setine dair dezavantajları sunmaktadır (Koyuncu ve ark., 2020).

Şekil 3.19 kapsamında ilk görsel, tümörlerin şekil ve boyut analizleri ile ilgilidir. İlk görselde yatay perspektiften detaylı inceleme yapılırsa; DDG veya YDG tipi tümörlerin, aynı kategoride dahi çok farklı şekil ve boyut özelliklerine sahip olabileceği gözlemlenir. Buna karşılık görsel inceleme dikey bir perspektiften yapıldığı durumda da YDG ve DDG tipi tümörlerin birbirine benzer şekil ve boyut özelliklerine sahip olabileceği görülür. Diğer bir deyişle, YDG ve DDG tip tümörlerin sınıflandırılması için şekil ve boyut hususunda doğrudan ayırt edici bir bilgi bulunmamaktadır (Koyuncu ve ark., 2020).

Şekil 3.19 kapsamında ikinci görsel, YDG ve DDG tip tümörlerin yoğunluk değerlendirmesi ile ilgilidir. Burada bir tümörün yoğunluk seviyesinin aynı kategoride çok fazla değişebileceği veya farklı kategoriler için benzer olabileceği görülmektedir. Diğer bir deyişle, YDG ve DDG tip tümörleri yoğunluk temelli ayırt etmek için belirgin bir özellik mevcut değildir (Koyuncu ve ark., 2020).

Dezavantajlarla ilgili olarak; öznitelik çıkarma ve sınıflandırıcı seçiminin kapsamlı analizleri, sınıf etiketlerinin etkin bir şekilde ayrıştırılabilmesinde önemli ve etken konuları teşkil etmektedir. Ayrıca önerilecek bir 3B modelin, 2B bir tümör modelinden daha zorlu bir konu teşkil edeceği de açık bir gerçektir. Çünkü Şekil 3.19 kapsamında belirtilen şekil, boyut ve yoğunluk dezavantajları, 3B ele alındığı durumda daha detaylı analiz ve bilgi gerektirecektir. Bu amaçla önerilecek bir 3B sınıflama modelinde, etkin bilgi üretilmesinin yanı sıra kararlı sınıflayıcı seçimi ön plana çıkmalıdır (Koyuncu ve ark., 2020).



Şekil 3.19. Veri seti dezavantajları

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Tez çalışması kapsamında sekiz transfer öğrenme mimarisi (DenseNet201, InceptionResNetV2, InceptionV3, ResNet50, ResNet101, SqueezeNet, VGG19, Xception), hiperparametreleri ayarlamaları ile detaylı bir şekilde incelenmektedir. Bu bağlamda mimariler, önerilecek model için en uygun olanı belirlemek amacıyla kapsamlı bir şekilde karşılaştırılmaktadır. 3t2FTS yaklaşımı sonucunda elde edilen veriler, doğrudan derin öğrenme mimarilerinin girişine beslenmiştir. Böylelikle YDG / DDG sınıflandırmasını ele alan etkin bir modelin önerilmesi sağlanmaktadır. Mimariler, kapsamlı bir şekilde analiz sağlanması amacıyla 2-kat çapraz doğrulama test yöntemi kullanılarak değerlendirilmiştir.

Çizelge 4.1, kullanılan mimarilerin hiperparametre ayarlamalarını sunmaktadır. Bu bağlamda, etkin veri setleri ile önceden eğitilmiş mimarilerin bilgi kaybını önlemek için sadece önemli parametreler dikkate alınmaktadır. YDG / DDG ayrımında transfer öğrenme mimarilerinin en yüksek performansını gözlemlemek amacıyla dört etkin parametre incelenmektedir. Bu parametreler; mini-yığın boyutu, öğrenme oranı, öğrenme oranı düşüş faktörü (*LearnRateDropFactor: LRDF*) ve optimize edici şeklinde sıralanabilir. Sistemlerin parametre ayarlamalarında hem yeterli sayıda iterasyon olması, hem de hafıza hatası almamak için epok değeri '100' olarak sabitlenmiştir.

Çizelge 4.1. Transfer öğrenme mimarileri için hiperparametre ayarlamaları

Parametre	Değer / Aralık
Epok sayısı	100
Mini-yığın boyutu	16, 32
Öğrenme oranı	0.01, 0.001, 0.0001
LRDF	0.2, 0.4, 0.6, 0.8
Optimizör	Sgdm, Rmsprop, Adam

Çizelge 4.2. – Çizelge 4.9 kapsamında sırasıyla DenseNet201, InceptionResNetV2, InceptionV3, ResNet50, ResNet101, SqueezeNet, VGG19 ve Xception mimarileri için hiperparametre ayarlaması – sistem başarısı ile ilgili analizler sunulmaktadır.

4.1. DenseNet201 Temelli Analizler

Çizelge 4.2'deki sonuçlara göre; 24 farklı opsiyonda incelenen en iyileyiciler / optimize ediciler arasında *sgdm* ortalama doğruluğu %75,61 iken, *adam* ve *rmsprop* için bu oranlar sırasıyla %74,94 ve %72,67 olarak gözlemlenmiştir. LRDF değeri '0,2' seçildiğinde 18 opsiyonda ortalama %75,53 doğruluk kaydedilmiştir ve bu seçim diğer tercihlerden daha iyi performans sağlamıştır. Öğrenme oranı '0,0001' seçildiği durumda ise 24 denemede ortalama %75,24 doğruluk elde edilmiştir ve söz konusu öğrenme oranı başarısı diğer opsiyonlarınkinden daha yüksektir. Mini-yığın boyutu '32' seçildiğinde, '16' olarak seçildiği durumdan %1,45 daha yüksek ortalama doğruluk kaydedilmiştir. Burada '32' opsiyonu ile 36 deneme için ortalama %75,20 doğruluk gözlemlenmiştir. DenseNet201, doğruluk temelli deneylerde 72 denemede %74,47 ortalama başarı sağlamıştır. DenseNet201 mimarisi, elde edilen en yüksek doğruluk (%79,30) temelli incelendiğinde *sgdm* en iyileyici, '0,8' LRDF değeri, '0,001' öğrenme oranı ve '32' mini-yığın boyutu kullanılmaktadır.

Çizelge 4.2. DenseNet201 mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk
16	0,01	0,2	Adam	76,49	0,001	0,2	Adam	74,74	0,0001	0,2	Adam	75,40
			Sgdm	74,74			Sgdm	75,44			Sgdm	76,84
			Rmsprop	49,12			Rmsprop	77,90			Rmsprop	74,04
		0,4	Adam	71,93		0,4	Adam	71,93		0,4	Adam	76,49
			Sgdm	65,26			Sgdm	78,95			Sgdm	74,74
			Rmsprop	69,47			Rmsprop	74,04			Rmsprop	75,44
		0,6	Adam	76,49		0,6	Adam	76,14		0,6	Adam	71,58
			Sgdm	74,04			Sgdm	78,25			Sgdm	76,14
		0,8	Rmsprop	73,68		0,8	Rmsprop	73,68		0,8	Rmsprop	75,10
			Adam	76,49			Adam	75,44			Adam	74,39
		0,8	Sgdm	74,74		0,8	Sgdm	76,84		0,8	Sgdm	76,49
			Rmsprop	49,12			Rmsprop	66,32			Rmsprop	74,39
32	0,01	0,2	Adam	78,25	0,001	0,2	Adam	74,39	0,0001	0,2	Adam	75,09
			Sgdm	69,12			Sgdm	77,54			Sgdm	75,79
			Rmsprop	71,93			Rmsprop	75,09			Rmsprop	76,84
		0,4	Adam	78,25		0,4	Adam	76,49		0,4	Adam	75,01
			Sgdm	77,54			Sgdm	74,74			Sgdm	75,09
			Rmsprop	71,93			Rmsprop	77,90			Rmsprop	76,84
		0,6	Adam	71,23		0,6	Adam	70,18		0,6	Adam	73,68
			Sgdm	77,52			Sgdm	75,44			Sgdm	75,79
		0,8	Rmsprop	73,68		0,8	Rmsprop	65,61		0,8	Rmsprop	73,33
			Adam	75,44			Adam	77,54			Adam	75,44
		0,8	Sgdm	76,84		0,8	Sgdm	79,30		0,8	Sgdm	75,79
			Rmsprop	74,39			Rmsprop	74,04			Rmsprop	76,14

4.2. InceptionResNetV2 Temelli Analizler

Çizelge 4.3'teki sonuçlara göre; 24 farklı opsiyonda incelenen en iyileycilerden *sgdm*'ye dair ortalama doğruluk %75,37 olarak kaydedilmiştir. Ayrıca bu başarı, *adam* (%72,38) ve *rmsprop* (%72,09) optimize edici sonuçlarından daha yüksektir. LRDF değeri '0,4' seçildiğinde, 18 denemede ortalama %73,65 doğruluk elde edilmiştir ve bu oran diğer tercihlerden daha yüksek performans sağlamıştır. Öğrenme oranı '0,0001' seçildiği durumda ise 24 denemede ortalama %74,39 doğruluk gözlemlenmiştir ve bu öğrenme oranı diğer opsiyonlardan daha yüksek başarı kaydetmiştir. Mini-yığın boyutu '32' seçildiği durumda %0,04 daha iyi ortalama başarı gözlenmiştir ve bu seçenek ile 36 denemede %73,30 ortalama doğruluk sağlanmaktadır. InceptionResNetV2 mimarisi ortalama doğruluk temelli değerlendirmelerde 72 deneme üzerine %73,28 başarı sunmuştur. InceptionResNetV2, gözlemlenen en yüksek doğruluk (%77,90) açısından genelde *sgdm* en iyileyici, '0,0001' öğrenme oranı ve '16' mini-yığın boyutu kullanmaktadır.

Çizelge 4.3. InceptionResNetV2 mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk				
16	0,01	0,2	Adam	74,04	0,001	0,2	Adam	70,88	0,0001	0,2	Adam	73,68				
			Sgdm	77,90			Sgdm	72,98			Sgdm	77,90				
			Rmsprop	64,21			Rmsprop	73,33			Rmsprop	73,68				
		0,4	Adam	73,68		Adam	73,33	Adam		74,39						
			Sgdm	75,09		Sgdm	76,84	Sgdm		77,90						
			Rmsprop	73,68		Rmsprop	70,53	Rmsprop		73,68						
		0,6	Adam	72,28		Adam	74,74	Adam		74,39						
			Sgdm	74,39		Sgdm	72,98	Sgdm		77,90						
		0,8	Rmsprop	73,68		Rmsprop	68,42	Rmsprop		74,39						
			Adam	50,18		Adam	72,63	Adam		74,39						
		32	0,01	0,2		Sgdm	76,49	0,001		0,2	Sgdm	72,63	0,0001	0,2	Sgdm	77,90
						Rmsprop	73,33				Rmsprop	74,39			Rmsprop	74,39
Adam	70,88				Adam	74,04	Adam		71,58							
0,4	Sgdm			72,98	Sgdm	75,79	Sgdm		75,44							
	Rmsprop			73,33	Rmsprop	75,09	Rmsprop		70,18							
	Adam			73,33	Adam	71,58	Adam		72,98							
0,6	Sgdm			76,84	Sgdm	72,98	Sgdm		76,14							
	Rmsprop			70,53	Rmsprop	70,18	Rmsprop		71,93							
0,8	Adam			74,74	Adam	73,68	Adam		74,74							
	Sgdm			72,98	Sgdm	75,44	Sgdm		75,44							
0,8	Rmsprop			68,42	Rmsprop	71,93	Rmsprop		72,63							
	Adam			72,63	Adam	74,74	Adam		73,68							
0,8	Sgdm	72,63	Sgdm	75,79	Sgdm	75,44										
	Rmsprop	74,39	Rmsprop	73,33	Rmsprop	70,53										

4.3. InceptionV3 Temelli Analizler

Çizelge 4.4'deki sonuçlara göre; 24 denemede *sgdm* ortalama doğruluğu %75,16 iken, bu oran *adam* (%72,65) ve *rmsprop* (%71,64) optimize ediciler için daha düşük kalmıştır. LRDF değeri '0,2' seçildiğinde 18 denemede ortalama %74,11 doğruluk kaydedilmiştir ve bu başarının diğer opsiyonlarınkinden daha yüksek olduğu görülmüştür. Öğrenme oranı '0,001' olduğunda 24 denemede ortalama %73,96 doğruluk elde edilerek, diğer seçeneklere üstünlük sağlanmıştır. Mini-yığın boyutu '32' seçildiğinde 36 denemede ortalama %73,19 doğruluk gözlenmiştir ve bu oran diğer opsiyon başarısından %0,08 daha yüksektir. InceptionV3, ortalama doğruluğa dayalı değerlendirmede 72 deneme üzerine %73,15'lik bir başarı sunmaktadır. Gözlemlenen en yüksek doğruluk (%78,60) açısından InceptionV3'ün *sgdm* en iyileyici, '0,6' LRDF değeri, '0,001' öğrenme oranı ve '32' mini-yığın boyutu kullandığı tespit edilmiştir.

Çizelge 4.4. InceptionV3 mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk
16	0,01	0,2	Adam	74,74	0,001	0,2	Adam	75,09	0,0001	0,2	Adam	74,39
			Sgdm	75,80			Sgdm	76,84			Sgdm	73,68
			Rmsprop	73,68			Rmsprop	73,68			Rmsprop	73,68
		0,4	Adam	74,74		0,4	Adam	72,28		0,4	Adam	74,74
			Sgdm	72,98			Sgdm	75,44			Sgdm	75,79
			Rmsprop	49,83			Rmsprop	73,68			Rmsprop	75,09
		0,6	Adam	73,68		0,6	Adam	74,74		0,6	Adam	74,39
			Sgdm	77,19			Sgdm	77,19			Sgdm	76,84
		0,8	Rmsprop	56,14		0,8	Rmsprop	72,63		0,8	Rmsprop	73,68
			Adam	72,98			Adam	70,53			Adam	74,74
			Sgdm	75,79			Sgdm	75,79			Sgdm	72,63
		32	0,01	0,2		Adam	72,98	0,001		0,2	Adam	72,98
Sgdm	70,18				Sgdm	75,44	Sgdm		78,25			
Rmsprop	73,68				Rmsprop	75,09	Rmsprop		73,68			
0,4	Adam			73,68	0,4	Adam	73,68		0,4	Adam	68,07	
	Sgdm			73,68		Sgdm	72,28			Sgdm	78,25	
	Rmsprop			73,68		Rmsprop	76,49			Rmsprop	74,04	
0,6	Adam			62,46	0,6	Adam	71,23		0,6	Adam	68,07	
	Sgdm			73,33		Sgdm	78,60			Sgdm	76,49	
0,8	Rmsprop			72,98	0,8	Rmsprop	65,26		0,8	Rmsprop	72,98	
	Adam			75,09		Adam	75,09			Adam	72,98	
	Sgdm			71,93		Sgdm	74,39			Sgdm	75,09	
					Rmsprop	73,68				Rmsprop	73,68	

4.4. ResNet50 Temelli Analizler

Çizelge 4.5'teki sonuçlara göre; *sgdm* en iyileycisininin 24 denemede ortalama doğruluğu %74,88 olarak kaydedilirken, bu oran *adam* (%74,02) ve *rmsprop* (%71,52) optimize edicileri için daha düşük kalmıştır. LRDF değeri '0,4' iken 18 denemede ortalama %74,68 başarı sağlanarak, diğer opsiyonlardan daha yüksek performans elde edilmiştir. Öğrenme oranı '0,0001' seçildiğinde 24 denemede ortalama %75,64 başarı gözlenerek, diğer seçeneklere üstünlük sağlanmıştır. Mini-yığın boyutu '32' seçildiği durumda, '16' opsiyonundan %0,09 daha iyi ortalama doğruluk gözlemlenmiştir ve '32' seçeneğine dair ortalama başarının %73,52 olduğu tespit edilmiştir. ResNet50 mimarisi, ortalama doğruluğa dayalı değerlendirmede 72 deneme üzerine %73,47'lik bir başarı sunmuştur. Gözlemlenen en yüksek doğruluk (%80) açısından ResNet50'nin *sgdm* en iyileyci, '0,8' veya '0,4' LRDF değeri, '0,0001' öğrenme oranı ve '32' mini-yığın boyutu kullandığı görülmüştür.

Çizelge 4.5. ResNet50 mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk				
16	0,01	0,2	Adam	75,09	0,001	0,2	Adam	71,22	0,0001	0,2	Adam	75,79				
			Sgdm	74,74			Sgdm	74,74			Sgdm	76,84				
			Rmsprop	73,68			Rmsprop	73,33			Rmsprop	74,04				
		0,4	Adam	74,04		Adam	76,14	Adam		75,09						
			Sgdm	75,79		Sgdm	69,82	Sgdm		76,84						
			Rmsprop	73,68		Rmsprop	73,68	Rmsprop		76,14						
		0,6	Adam	73,68		Adam	74,35	Adam		75,79						
			Sgdm	74,39		Sgdm	74,73	Sgdm		76,84						
		0,8	Rmsprop	68,07		Rmsprop	71,58	Rmsprop		74,04						
			Adam	75,44		Adam	71,93	Adam		75,09						
		32	0,01	0,2		Sgdm	71,93	0,001		0,2	Sgdm	76,14	0,0001	0,2	Sgdm	72,63
						Rmsprop	45,96				Rmsprop	75,79			Rmsprop	74,39
Adam	74,04				Adam	71,58	Adam		72,98							
0,4	Sgdm			75,44	Sgdm	71,23	Sgdm		78,60							
	Rmsprop			73,33	Rmsprop	67,37	Rmsprop		74,74							
	Adam			75,79	Adam	73,68	Adam		75,79							
0,6	Sgdm			71,93	Sgdm	74,39	Sgdm		80,00							
	Rmsprop			71,58	Rmsprop	74,39	Rmsprop		75,44							
0,8	Adam			66,67	Adam	73,68	Adam		77,19							
	Sgdm			72,63	Sgdm	76,49	Sgdm		73,68							
0,8	Rmsprop			70,53	Rmsprop	74,39	Rmsprop		72,28							
	Adam			74,04	Adam	71,58	Adam		75,79							
0,8	Sgdm	73,68	Sgdm	73,68	Sgdm	80,00										
	Rmsprop	58,95	Rmsprop	73,68	Rmsprop	75,44										

4.5. ResNet101 Temelli Analizler

Çizelge 4.6'daki sonuçlara göre; *sgdm* en iyileycisinin 24 deneme üzerine ortalama doğruluğu (%75,39) olup, bu oran *adam* (%73,46) ve *rmsprop* (%71,34) optimize edicilerinin skorlarına kıyasla daha yüksektir. LRDF değeri '0,4' seçildiği durumda 18 deneme üzerine %73,87 ortalama doğruluk elde ederek, diğer tercihlerden daha yüksek performans gözlenmiştir. Öğrenme oranı '0,0001' seçildiğinde 24 denemede ortalama %74,39 doğruluk kaydedilerek, diğer öğrenme oranlarına üstünlük sağlanmıştır. Mini-yığın boyutu '16' seçildiği durumda, 36 deneme üzerine %73,52 ortalama doğruluk ve diğer opsiyona göre %0,31 daha yüksek performans gözlemlenmiştir. ResNet101, ortalama doğruluk temelli değerlendirmelerde 72 deneme üzerine %73,37 başarı sunmuştur. Gözlemlenen en yüksek doğruluk (%77,89) açısından, ResNet101'in '0,0001' öğrenme oranı, '32' mini-yığın boyutu, *sgdm* en iyileyici ile '0,4' LRDF değeri veya *rmsprop* optimize edici ile '0,6' LRDF değeri kullandığı görülmüştür.

Çizelge 4.6. ResNet101 mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk				
16	0,01	0,2	Adam	74,03	0,001	0,2	Adam	69,82	0,0001	0,2	Adam	77,54				
			Sgdm	76,14			Sgdm	76,14			Sgdm	76,49				
			Rmsprop	73,68			Rmsprop	50,17			Rmsprop	75,38				
		0,4	Adam	73,68		0,4	Adam	77,19		0,4	Adam	75,08				
			Sgdm	77,54			Sgdm	72,98			Sgdm	75,78				
			Rmsprop	73,68			Rmsprop	68,72			Rmsprop	72,98				
		0,6	Adam	72,98		0,6	Adam	76,14		0,6	Adam	70,17				
			Sgdm	75,78			Sgdm	73,33			Sgdm	73,68				
		0,8	Rmsprop	73,68		0,8	Rmsprop	73,68		0,8	Rmsprop	70,87				
			Adam	75,08			Adam	72,28			Adam	75,08				
		32	0,01	0,2		Adam	75,43	0,001		0,2	Adam	74,43	0,0001	0,2	Adam	74,03
						Sgdm	74,43				Sgdm	76,49			Sgdm	75,38
Rmsprop	67,01				Rmsprop	73,68	Rmsprop		77,19							
0,4	Adam			75,43	0,4	Adam	68,72		0,4	Adam	70,17					
	Sgdm			74,43		Sgdm	74,68			Sgdm	77,89					
	Rmsprop			73,68		Rmsprop	74,68			Rmsprop	72,33					
0,6	Adam			76,49	0,6	Adam	72,33		0,6	Adam	70,17					
	Sgdm			74,38		Sgdm	76,14			Sgdm	76,84					
0,8	Rmsprop			65,96	0,8	Rmsprop	73,68		0,8	Rmsprop	77,89					
	Adam			72,98		Adam	74,43			Adam	69,47					
					Sgdm	74,73				Sgdm	71,22				Sgdm	74,38
					Rmsprop	73,68				Rmsprop	57,19				Rmsprop	76,49

4.6. SqueezeNet Temelli Analizler

Çizelge 4.7’deki sonuçlara göre; 24 denemede *sgdm*, *adam* ve *rmsprop* en iyileyicilerinin ortalama doğrulukları sırasıyla %73,94, %70,99 ve %72,03 olarak gözlenmiştir. LRDF değeri ‘0,6’ olduğu seçenek için 18 deneme üzerine %73,78 ortalama doğruluk elde ederek, diğer tercihlerden daha iyi performans sağlanmıştır. Öğrenme oranı ‘0,001’ olduğu durumda 24 denemede ortalama %73,90 doğruluk kaydedilerek, söz konusu opsiyon diğer öğrenme oranlarına üstün gelmiştir. Mini-yığın boyutu ‘16’ seçildiğinde 36 deneme üzerine %73,12 ortalama doğruluk ve diğer opsiyona %1,59 üstünlük sağlanmıştır. SqueezeNet, ortalama doğruluk temelli değerlendirmeler neticesinde 72 deneme üzerine %72,32 başarı sunmuştur. SqueezeNet’in gözlemlenen en yüksek doğruluk (%75,44) açısından *sgdm* optimize edici, ‘0,8’ LRDF değeri, ‘0,001’ öğrenme oranı ve ‘32’ mini-yığın boyutu kullandığı, *sgdm* optimize edici, ‘0,2’ LRDF değeri, ‘0,001’ öğrenme oranı ve ‘16’ mini-yığın boyutu işlettiği veya *adam* optimize edici, ‘0,4’ LRDF değeri, ‘0,0001’ öğrenme oranı ve ‘32’ mini-yığın boyutu üzerinden yetkilendirildiği tespit edilmiştir.

Çizelge 4.7.SqueezeNet mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk
16	0,01	0,2	Adam	73,68	0,001	0,2	Adam	73,68	0,0001	0,2	Adam	71,23
			Sgdm	73,41			Sgdm	75,44			Sgdm	73,91
			Rmsprop	73,68			Rmsprop	73,68			Rmsprop	57,54
		0,4	Adam	73,68		Adam	73,68	Adam		68,42		
			Sgdm	74,41		Sgdm	73,91	Sgdm		73,68		
			Rmsprop	73,68		Rmsprop	73,68	Rmsprop		72,98		
		0,6	Adam	73,68		Adam	73,68	Adam		72,98		
			Sgdm	74,41		Sgdm	73,41	Sgdm		73,68		
			Rmsprop	73,68		Rmsprop	73,68	Rmsprop		74,74		
		0,8	Adam	73,68		Adam	73,68	Adam		71,93		
			Sgdm	73,68		Sgdm	75,08	Sgdm		73,68		
			Rmsprop	73,68		Rmsprop	73,68	Rmsprop		74,74		
32	0,01	0,2	Adam	49,83	0,001	0,2	Adam	73,68	0,0001	0,2	Adam	72,98
			Sgdm	73,41			Sgdm	73,95			Sgdm	73,68
			Rmsprop	73,68			Rmsprop	73,68			Rmsprop	61,40
		0,4	Adam	73,68		Adam	73,68	Adam		75,44		
			Sgdm	73,68		Sgdm	73,91	Sgdm		73,68		
			Rmsprop	73,68		Rmsprop	73,68	Rmsprop		74,04		
		0,6	Adam	73,68		Adam	73,68	Adam		73,68		
			Sgdm	73,41		Sgdm	73,41	Sgdm		73,68		
			Rmsprop	73,68		Rmsprop	73,68	Rmsprop		75,09		
		0,8	Adam	50,18		Adam	73,68	Adam		65,61		
			Sgdm	73,95		Sgdm	75,44	Sgdm		73,68		
			Rmsprop	73,68		Rmsprop	73,68	Rmsprop		59,30		

4.7. VGG19 Temelli Analizler

Çizelge 4.8'deki sonuçlara göre; *sgdm* ortalama doğruluğunun 24 deneme üzerine %74,85 olduğu, bu oranın *adam* (%73,67) ve *rmsprop* (%72,62) optimize edicilerinin puanlarına kıyasla daha üstün olduğu gözlemlenmiştir. LRDF değeri '0,2' olduğunda 18 deneme üzerine %73,88 ortalama doğruluk elde edilerek, diğer opsiyonlara üstünlük sağlanmıştır. Öğrenme oranı '0,0001' seçildiğinde, 24 deneme üzerine diğer seçeneklerden daha yüksek ortalama doğruluk (%74,21) kaydedilmiştir. Mini-yığın boyutu '32' olduğu durumda ise 36 deneme üzerine ortalama %73,75 doğruluk ve diğer tercihten %0,08 daha iyi başarı gözlenmiştir. VGG19, ortalama doğruluk temelli değerlendirmelerde 72 denemede %73,71 başarı oranı sunmuştur. Gözlemlenen en yüksek doğruluk (%78,25) hususunda VGG19'un *sgdm* en iyileyici, '0,2' LRDF değeri, '0,0001' öğrenme oranı ve '32' mini-yığın boyutu kullandığı görülmüştür.

Çizelge 4.8. VGG19 mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk
16	0,01	0,2	Adam	72,98	0,001	0,2	Adam	71,23	0,0001	0,2	Adam	73,68
			Sgdm	75,44			Sgdm	75,09			Sgdm	73,33
			Rmsprop	70,88			Rmsprop	77,54			Rmsprop	72,63
		0,4	Adam	73,68		0,4	Adam	75,09		0,4	Adam	71,23
			Sgdm	74,39			Sgdm	71,93			Sgdm	75,09
			Rmsprop	73,68			Rmsprop	75,09			Rmsprop	71,93
		0,6	Adam	74,74		0,6	Adam	72,28		0,6	Adam	70,88
			Sgdm	76,49			Sgdm	77,90			Sgdm	77,54
			Rmsprop	72,28			Rmsprop	74,04			Rmsprop	73,68
		0,8	Adam	75,09		0,8	Adam	76,84		0,8	Adam	74,39
			Sgdm	76,49			Sgdm	73,33			Sgdm	76,49
			Rmsprop	54,74			Rmsprop	77,54			Rmsprop	75,44
32	0,01	0,2	Adam	75,09	0,001	0,2	Adam	73,68	0,0001	0,2	Adam	71,23
			Sgdm	74,04			Sgdm	75,79			Sgdm	78,25
			Rmsprop	70,53			Rmsprop	74,04			Rmsprop	74,39
		0,4	Adam	75,79		0,4	Adam	70,88		0,4	Adam	75,44
			Sgdm	75,79			Sgdm	70,53			Sgdm	76,14
			Rmsprop	73,68			Rmsprop	75,44			Rmsprop	72,98
		0,6	Adam	73,68		0,6	Adam	71,58		0,6	Adam	76,14
			Sgdm	72,98			Sgdm	72,28			Sgdm	72,98
			Rmsprop	73,68			Rmsprop	64,21			Rmsprop	75,44
		0,8	Adam	74,74		0,8	Adam	73,68		0,8	Adam	74,04
			Sgdm	75,09			Sgdm	73,69			Sgdm	75,44
			Rmsprop	74,04			Rmsprop	72,63			Rmsprop	72,28

4.8. Xception Temelli Analizler

Çizelge 4.9'daki sonuçlara göre; 24 denemede *adam*, *sgdm* ve *rmsprop* en iyileycilerinin ortalama başarıları sırasıyla %73,95, %73,45 ve %71,20 olarak gözlemlenmiştir. LRDF değeri '0,2' seçimi ile 18 deneme üzerine %73,68 ortalama doğruluk elde edilerek, diğer seçeneklere üstünlük sağlanmıştır. Öğrenme oranı '0,001' seçildiğinde 24 deneme üzerine ortalama %73,35 doğruluk elde edilerek, diğer opsiyonlardan daha yüksek performans gözlenmiştir. Mini-yığın boyutu '16' opsiyonu ile 36 deneme üzerine ortalama %73,67 doğruluk ve diğer tercihe %1,61 üstünlük kaydedilmiştir. Ortalama doğruluk temelli değerlendirmelerde Xception mimarisi 72 deneme üzerine %72,87 başarı sunmuştur. Xception mimarisinin gözlenen en yüksek doğruluk (%76,49) düşünüldüğünde *sgdm* en iyileyici, '16' mini-yığın boyutu, '0,2' LRDF değeri ve '0,01' öğrenme oranı işlettiği, *sgdm* optimize edici, '0,4' LRDF değeri, '0,001' öğrenme oranı ve '16' mini-yığın boyutu kullandığı veya *rmsprop* en iyileyici, '0,8' LRDF değeri, '0,0001' öğrenme oranı ve '16' mini-yığın boyutu üzerinden sonuç verdiği tespit edilmiştir.

Çizelge 4.9. Xception mimarisinin sonuçları

Mini-yığın Boyutu	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk	Öğrenme Oranı	LRDF	Optimize Edici	Doğruluk				
16	0,01	0,2	Adam	73,68	0,001	0,2	Adam	74,39	0,0001	0,2	Adam	72,28				
			Sgdm	76,49			Sgdm	71,93			Sgdm	71,23				
			Rmsprop	73,68			Rmsprop	74,39			Rmsprop	75,44				
		0,4	Adam	75,09		0,4	Adam	75,79		0,4	Adam	72,63				
			Sgdm	71,93			Sgdm	76,49			Sgdm	71,23				
			Rmsprop	73,68			Rmsprop	68,07			Rmsprop	72,63				
		0,6	Adam	73,68		0,6	Adam	75,09		0,6	Adam	71,23				
			Sgdm	74,04			Sgdm	75,44			Sgdm	70,18				
		0,8	Rmsprop	73,68		0,8	Rmsprop	72,98		0,8	Rmsprop	72,63				
			Adam	75,44			Adam	76,14			Adam	74,39				
		32	0,01	0,2		Sgdm	76,14	0,001		0,2	Sgdm	73,68	0,0001	0,2	Sgdm	72,63
						Rmsprop	70,53				Rmsprop	74,74			Rmsprop	71,93
Adam	74,04				Adam	72,98	Adam		73,68							
0,4	Sgdm			72,63	0,4	Sgdm	72,98		0,4	Sgdm	75,44					
	Rmsprop			71,23		Rmsprop	75,09			Rmsprop	73,68					
	Adam			74,04		Adam	74,39			Adam	71,93					
0,6	Sgdm			73,33	0,6	Sgdm	74,74		0,6	Sgdm	70,53					
	Rmsprop			62,11		Rmsprop	57,90			Rmsprop	70,18					
0,8	Adam			75,79	0,8	Adam	72,28		0,8	Adam	72,63					
	Sgdm			75,44		Sgdm	70,88			Sgdm	75,09					
				Rmsprop	53,33		Rmsprop		74,74		Rmsprop	70,18				

Tez çalışmadaki derin öğrenme temelli deneyler; 2.50 GHz CPU, 8 GB RAM ve Intel(R) Core(TM) i5-7200U ekran kartına sahip bir kişisel bilgisayarda, MATLAB yazılımında yer alan Deep Network Designer araç kutusunda gerçekleştirilmiştir.

4.9. Detaylı Analiz ve Tartışma

Bölüm 4 kapsamındaki değerlendirmelere göre tüm derin öğrenme mimarilerinin gözlenen performanslarına dair derinlemesine analiz gerekliliğini ortaya koyduğu (çok yakın sonuçlar ürettikleri) görülmektedir. Deneyler boyunca test yöntemi olarak 2-kat çapraz doğrulama kullanılmakta olup, diğer test yöntemleri (%50-50 eğitim-test ayrımı, %70-30 eğitim-test ayrımı vb.) değerlendirmeye alınmamaktadır. Buradaki amaç, mimarileri zorlayarak performans karşılaştırmalarının sağlanmasıdır. Derin öğrenme yöntemlerinin kullanıldığı literatür araştırmalarında, tez çalışmasında ele alınmayan veri çoğaltma (*augmentation*) sıklıkla tercih edilmektedir. Diğer bir deyişle amacımız, 3t2FTS yaklaşımının işletildiği durum için veri çoğaltma olmadan (ancak yeterli eğitim verisiyle) transfer öğrenme mimarilerinin karşılaştırılmasıdır. Ayrıca çalışma kapsamında önerilen 2B-ID giriş görüntüleri, veri çoğaltma için uygun bir yapıya sahip değildir.

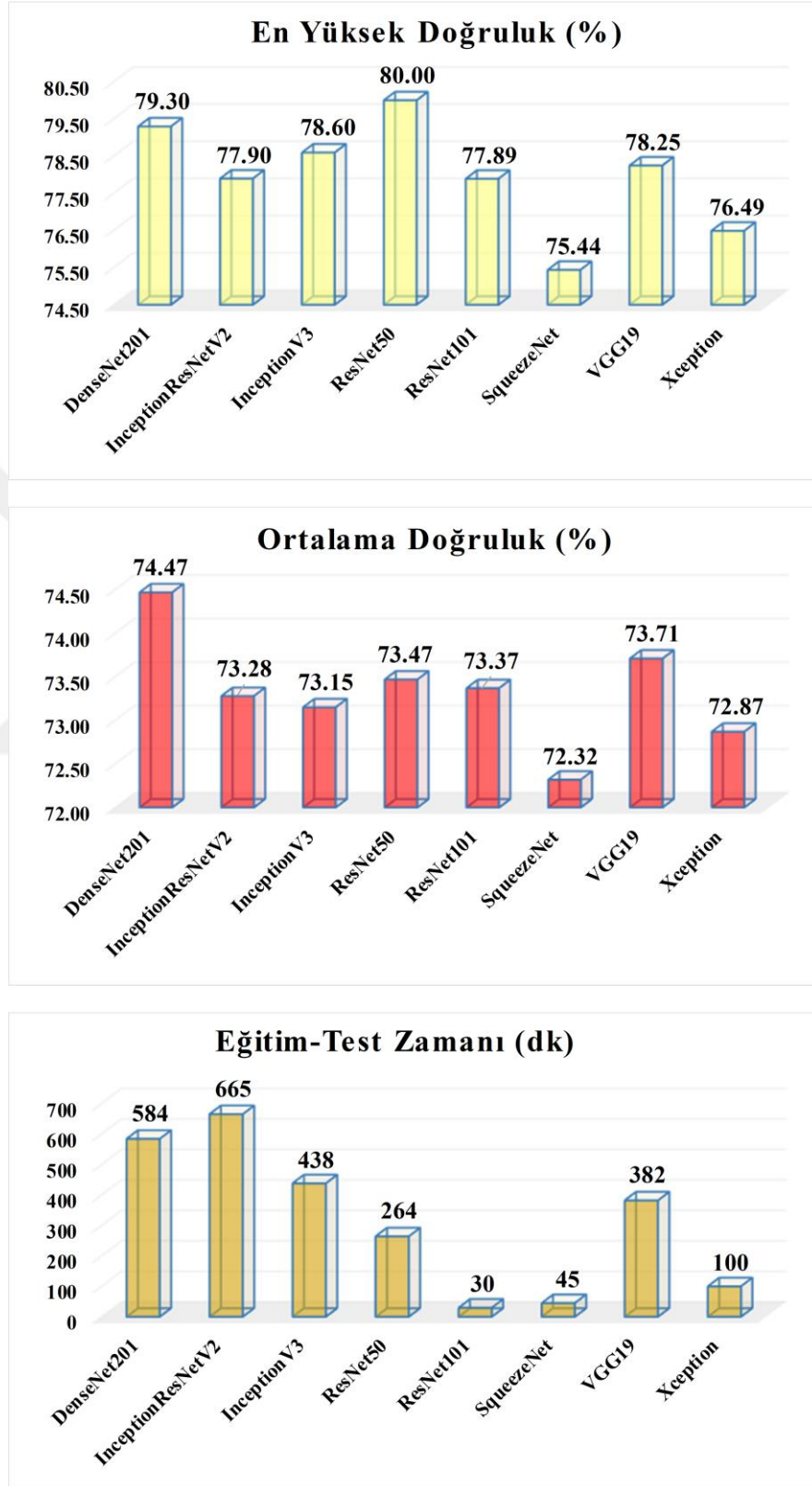
Şekil 4.1 kapsamında, 3t2FTS ve derin öğrenme mimarilerini içeren modellerin derinlemesine analizini sağlamak için üç aşamalı bir analiz sunulmaktadır. Bu bağlamda 8 transfer öğrenme yapısına dair en yüksek başarı, ortalama doğruluk ve hesaplama süresi temelli karşılaştırmalar değerlendirilmektedir. Bölüm 4 ve Şekil 4.1 kapsamındaki değerlendirmeler ışığında;

- Transfer öğrenme mimarilerinin genel olarak daha düşük öğrenme oranları ve *sgdm* en iyileyci ile daha yüksek performans üretme eğiliminde oldukları görülmektedir. LRDF ve mini-yığın boyutu parametreleri ile ilgili olarak tanımlanabilecek – kesin bir ayarlama mevcut değildir ve bu parametrelere dair optimum değerlerin mimariden mimariye farklılık gösterdiği bilinmektedir.

- En yüksek doğruluk oranı (%80) ResNet50 mimarisi ile elde edilirken, DenseNet201 ve InceptionV3 sırasıyla ikinci ve üçüncü en iyi doğruluk oranlarını elde etmektedir.
- En yüksek ortalama doğruluk DenseNet201 mimarisi tarafından kaydedilmektedir. Bu noktada VGG19 ve ResNet50 mimarilerinin sırasıyla ikinci ve üçüncü en yüksek ortalama doğrulukları sağladığı görülür. Diğer bir deyişle bu üç mimari, 72 deneme boyunca yapılan analizler neticesinde en kararlı mimariler olarak ön plana çıkmaktadır.
- En düşük hesaplama süresi ResNet101 tarafından kaydedilirken, SqueezeNet ve Xception mimarilerine dair sırasıyla ikinci ve üçüncü en iyi sonuçlar gözlenmektedir. Bu noktada ResNet50 mimarisi, 264 dakika içinde sonuç vererek dördüncü en iyi performansa sahip mimari olmaktadır.
- DenseNet201 mimarisinin, en yüksek ortalama doğruluk oranı ve ikinci en yüksek doğruluk değerini sağladığı görülmektedir. Ancak hesaplama süresi temelli karşılaştırmada, mimarinin 584 dakikada sonuçlandırma sağladığı ve bu performansın ikinci en kötü performans olduğu gözlemlenmektedir.
- Üç kararlı mimariden biri olan ResNet50 mimarisi; en yüksek sınıflandırma doğruluğunu elde ederek, özellikle DenseNet201, InceptionV3 ve VGG19 mimarilerine göre daha kısa sürede çıktı verebilmektedir. Bu noktada ResNet50 mimarisi; kararlı (en yüksek ortalama doğruluk veren) üç mimari arasında yer alan, bu üç mimari arasında en düşük hesaplama süresine sahip ve bütün mimariler arasında en yüksek doğruluğu kaydeden yapı olarak ön plana çıkmaktadır.

ResNet50 mimarisi; YDG ve DDG kategorizasyonunu gerçekleştirmek için kararlı mimarilere (DenseNet201 ve VGG19) kıyasla en yüksek doğruluğu, kararlı bir

ortalama performansı ve daha az çalışma süresini kaydederek, 3t2FTS yaklaşımı ile kullanılmaya en uygun mimari olduğunu ispat etmektedir.



Şekil.4.1 Derin öğrenme mimarilerinin derinlemesine karşılaştırılması

Çizelge 4.10 kapsamında önerilen 3t2FTS yaklaşımı ve ResNet50 mimarisini içeren model ile literatürdeki en son çalışmaların karşılaştırılması sunulmaktadır.

Çizelge 4.10. Literatür karşılaştırması

Çalışma	Önerilen Yaklaşım	Veri Seti	Görev	Doğruluk (%)
Latif ve ark. (2018)	Tüm sekans bilgileri, ADD, birincil ve ikincil derece istatistikler ve ÇKP içeren model	39 YDG / 26 DDG & 110 YDG + DDG (BraTS 2015)	2B-temelli sınıflandırma (tümör / tümör olmayan)	96,73
Gupta ve ark. (2019)	FLAIR+T2 sekans kombinasyonu, morfolojik işlemler, doğal özellikler ve çoğunluk oylamasına dayalı TS içeren model	120 YDG / 80 DDG (BraTS 2012)	2B-temelli sınıflandırma (YDG vs DDG)	96,75
Kumar ve ark. (2020)	Tüm sekans bilgileri, DDD, BDİ, ÖÖE ve RO içeren model	210 YDG / 75 DDG (BraTS 2017/2018)	2B-temelli sınıflandırma (YDG vs DDG)	97,54
Saba ve ark. (2020)	YGH, YİÖ, derin özellikler ve TS içeren model	110 YDG + DDG (BraTS 2015)	2B-temelli sınıflandırma (YDG vs DDG)	91,30
		220 YDG / 54 DDG (BraTS 2016)		91,47
		210 YDG / 75 DDG (BraTS 2017/2018)		98,39
Sharif ve ark. (2020)	T2+FLAIR sekans kombinasyonu, YGH, YİÖ, geometrik özellikler ve DVM içeren model	20 YDG / 10 DDG (BraTS 2013)	2B-temelli sınıflandırma (sağlıklı vs hastalıklı)	98
		220 YDG / 54 DDG & 110 YDG + DDG (BraTS 2015)		100
Bodapati ve ark. (2021)	InceptionResNetV2 ve Xception kullanan iki kanallı DSA modeli	210 YDG / 75 DDG (BraTS 2017/2018)	2B-temelli sınıflandırma (YDG vs DDG)	93,69
Koyuncu ve ark. (2020)	FLAIR+T1+T2 sekans kombinasyonu, BDİ, Wilcoxon sıralaması ve GH-KPSO-YSA içeren model	210 YDG / 75 DDG (BraTS 2017/2018)	3B-temelli sınıflandırma (YDG vs DDG)	90,18
Tez çalışması	3t2FTS ve ResNet50 içeren model	210 YDG / 75 DDG (BraTS 2017/2018)	3B-temelli sınıflandırma (YDG vs DDG)	80

Çizelge 4.10'da görüldüğü üzere literatür çalışmaları, genel olarak MR verilerinin 2B temelli sınıflandırılmasını ele almaktadır. Bununla birlikte 2B temelli bir sınıflandırma, daha önce de belirtildiği üzere bir tıp uzmanı tarafından bir görüntü seçimini gerektiren yarı-otomatik bir süreç anlamına gelir. Burada tam otomatik bir sınıflama modelinin, 3B veri analizine uyumlu üretilmesi ve bir segmentasyon sistemine dair 3B çıktıya (3B tümöre) adaptif olması gerektiği bilinmektedir.

Literatürde Koyuncu ve ark. (2020); YDG ve DDG tipi tümörleri doğrudan 3B temelli ele alarak, sınıflandırmayı etkin, istatistiksel ve deneysel bir model ile sağlamaktadır. Ancak bu çalışmada yüksek başarı sağlamasına rağmen, etkin bir özellik dönüşüm stratejisi veya derin öğrenme temelli bir sınıflama sistemi bulunmamaktadır. Çizelge 4.10'da görüldüğü gibi 3t2FTS yaklaşımı ve ResNet50 mimarisini içeren model ile kayda değer başarı sağlanmakta, elde edilen başarı (doğruluk oranı) geliştirmeye açık bir konu teşkil etmektedir. Diğer bir deyişle, 3t2FTS yeni bir strateji olarak önerilmekte ve uygulama uygulamalarının çeşitlendirilebileceği öngörülmektedir. Bu aşamada 3t2FTS uyumlu derin öğrenme yapılarının önerilmesi de literatüre rehberlik edecek farklı bir konuyu teşkil edecektir. Ayrıca, elde edilen sonuçlarla ilgili olarak, ResNet50 temelli çeşitli mimarilerin de performans incelemelerinin sağlanması ile farklı sonuçların elde edilebileceği öngörülmektedir.

5. SONUÇLAR VE ÖNERİLER

Tez çalışması kapsamında, 3t2FTS isimli yeni bir özellik çıkarma stratejisi ve ResNet50 transfer öğrenme mimarisi kullanılarak, YDG ve DDG tipi tümörlerin sınıflandırılmasını sağlayan ümit verici bir model önerilmektedir. Yapılan incelemeler ve elde edilen bulgulara göre;

- 3t2FTS yaklaşımı; üzerinde işlem yapılan 3B uzayı, BDİ özellikleri kullanarak 2B uzaya dönüştürmektedir. Böylelikle 3B voksel bazlı bilgi incelenerek, 3B MR görüntülerinin basit bir 2B-ID formuna özetlenmesi sağlanmaktadır. Önerilen bu yaklaşımın farklı türdeki tümörlerin veya alt tümör alanlarının sınıflandırılmasında da MR görüntüleri üzerine aplike edilebileceği görülmektedir. Ayrıca çeşitli beyin hastalıklarının; bütün bir 3B beyin görüntüsüne 3t2FTS uygulanması ile üretilecek 2B-ID görüntüleri üzerinden, farklı modellerin kullanımı ile sınıflanabileceği öngörülmektedir. 3t2FTS yaklaşımına adaptif ve kararlı bir derin öğrenme mimarisi üretilmesi de gelişmeye açık farklı bir konuyu ihtiva etmektedir.
- Çalışma kapsamında ResNet50; doğruluk ve hesaplama zamanına dayalı derinlemesine analizler neticesinde, tümör derecelendirme konusu üzerine yedi nitelikli transfer öğrenme mimarisi arasında en optimum seçenek olarak ön plana çıkmaktadır. Burada *sgdm* ve düşük öğrenme oranları, derin öğrenme mimarileri için en çok tekrarlanan optimize edici ve öğrenme oranı ayarlaması olarak dikkat çekmektedir.
- Analizlerle ilintili olarak; ResNet temelli önerilen veya üretilebilecek derin öğrenme mimarilerinin, 3t2FTS yaklaşımı ile MR görüntüleri üzerine farklı sınıflama uygulamaları için yüksek başarı elde edebileceği, bu aşamada ResNet50 mimarisini doğrudan kullanan hibrit bir derin öğrenme mimarisinin de başarı oranını artırabileceği öngörülmektedir.

Tez çalışma neticesinde literatürde 3B MRG temelli derecelendirme görevlerine rehberlik edebilecek verimli bir araştırma sunulmaktadır. Özetle; gliyom derecelendirme veya 3B beyin MR görüntüsü temelli tümör / hastalık tespiti için özellik dönüşüm stratejisi ile ResNet veya ResNet50 temelli mimari kullanımı, 3t2FTS'ye uyumlu yeni bir derin öğrenme mimarisinin üretilmesi veya önerilmesi, 3t2FTS'ye uyumlu önerilen mimaride ResNet / ResNet50 temelli yeni, türev veya hibrit yaklaşım(lar)ın işletilmesi hususları tez çalışmasının önerisi ve konunun evrilebileceği alanları teşkil etmektedir.



KAYNAKLAR

- Al Husaini, M. A. S., Habaebi, M. H., Gunawan, T. S., Islam, M. R., Elsheikh, E. A. and Suliman, F. M., 2022, Thermal-based early breast cancer detection using inception V3, inception V4 and modified inception MV4, *Neural Computing and Applications*, 34 (1), 333-348.
- Apaydin, H., Feizi, H., Sattari, M. T., Colak, M. S., Shamshirband, S. and Chau, K. W., 2020, Comparative analysis of recurrent neural network architectures for reservoir inflow forecasting, *Water*, 12 (5), 1500.
- Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J. S., Freymann, J. B., Farahani, K. and Davatzikos, C., 2017, Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features, *Scientific Data*, 4 (1), 1-13.
- Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., ... Jambawalikar, S. R., 2018, Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge, *arXiv preprint*, arXiv:1811.02629.
- Bansal, M., Kumar, M., Sachdeva, M. and Mittal, A., 2021, Transfer learning for image classification using VGG19: Caltech-101 image data set, *Journal of Ambient Intelligence and Humanized Computing*, 1-12.
- Bhandare, A. and Kaur, D., 2021, Designing convolutional neural network architecture using genetic algorithms, *International Journal of Advanced Network, Monitoring and Controls*, 6 (3), 26-35.
- Bodapati, J. D., Shaik, N. S., Naralasetti, V. and Mundukur, N. B., 2021, Joint training of two-channel deep neural network for brain tumor classification, *Signal, Image and Video Processing*, 15 (4), 753-760.

- Chollet F., 2017, Xception: Deep learning with depthwise separable convolutions, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR-2017, Honolulu, 1251-1258.
- Cireşan, D. C., Meier, U. and Schmidhuber, J., 2012, Transfer learning for Latin and Chinese characters with deep neural networks, *The 2012 International Joint Conference on Neural Networks*, IJCNN-2012, Brisbane, 1-6.
- Es-Sabery, F., Hair, A., Qadir, J., Sainz-De-Abajo, B., García-Zapirain, B. and De La Torre-Díez, I., 2021, Sentence-level classification using parallel fuzzy deep learning classifier, *IEEE Access*, 9, 17943-17985.
- Fang, L., Wang, X., Lian, Z., Yao, Y. and Zhang, Y., 2022, Supervoxel-based brain tumor segmentation with multimodal MRI images, *Signal, Image and Video Processing*, 1-9.
- Ghaffari, M., Samarasinghe, G., Jameson, M., Aly, F., Holloway, L., Chlap, P., Koh, E-S., Sowmya, A. and Oliver, R., 2022, Automated post-operative brain tumour segmentation: A deep learning model based on transfer learning from pre-operative images, *Magnetic Resonance Imaging*, 86, 28-36.
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J. and Chen, T., 2018, Recent advances in convolutional neural networks, *Pattern recognition*, 77, 354-377.
- Gupta, N., Bhatele, P. and Khanna, P., 2019, Glioma detection on brain MRIs using texture and morphological features with ensemble learning, *Biomedical Signal Processing and Control*, 47, 115-125.
- Gupta, J., Pathak, S. and Kumar, G., 2022, Deep learning (CNN) and transfer learning: A review. *Journal of Physics: Conference Series*, 2273 (1), 012029.

- Hassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z. and Jasińska, E., 2021, Identification of plant-leaf diseases using CNN and transfer-learning approach, *Electronics*, 10 (12), 1388.
- He, K., Zhang, X., Ren, S. and Sun, J., 2016, Deep residual learning for image recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR-2016, Las Vegas, 770-778.
- Hijazi, S., Kumar, R. and Rowen, C., 2015, Using convolutional neural networks for image recognition, *Cadence Design Systems Inc.*, San Jose, CA, USA, 9.
- Huang, G, Liu, Z, Van Der Maaten, L. and Weinberger, K. Q., 2017, Densely connected convolutional networks, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR-2017, Honolulu, 4700-4708.
- Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J. and Keutzer, K., 2016, SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size, *arXiv preprint*, arXiv:1602.07360.
- Jaiswal, A., Gianchandani, N., Singh, D., Kumar, V. and Kaur, M., 2021, Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning, *Journal of Biomolecular Structure and Dynamics*, 39 (15), 5682-5689.
- Jiang, X., Hu, B., Chandra Satapathy, S., Wang, S. H. and Zhang, Y. D., 2020, Fingerspelling identification for Chinese sign language via AlexNet-based transfer learning and Adam optimizer, *Scientific Programming*, 2020, 1-13.
- Kim, Y. H., Park, J. B., Chang, M. S., Ryu, J. J., Lim, W. H. and Jung, S. K., 2021, Influence of the depth of the convolutional neural networks on an artificial intelligence model for diagnosis of orthognathic surgery, *Journal of Personalized Medicine*, 11 (5), 356.

- Kora, P., Ooi, C. P., Faust, O., Raghavendra, U., Gudigar, A., Chan, W. Y., Meenakshi, K., Swaraja, K., Plawiak, P. and Acharya, U. R., 2022, Transfer learning techniques for medical image analysis: A review, *Biocybernetics and Biomedical Engineering*, 42, 79-107.
- Koyuncu, H., Barstuğan, M. and Öziç, M. Ü., 2020, A comprehensive study of brain tumour discrimination using phase combinations, feature rankings, and hybridised classifiers, *Medical & Biological Engineering & Computing*, 58 (12), 2971-2987.
- Koyuncu, H. and Barstuğan, M., 2021, COVID-19 discrimination framework for X-ray images by considering radiomics, selective information, feature ranking, and a novel hybrid classifier, *Signal Processing: Image Communication*, 97, 116359.
- Krizhevsky, A., Sutskever, I. and Hinton, G. E., 2017, Imagenet classification with deep convolutional neural networks, *Communications of the ACM*, 60.6, 84-90.
- Kronberg, R. M., Meskelevicius, D., Sabel, M., Kollmann, M., Rubbert, C. and Fischer, I., 2022, Optimal acquisition sequence for AI-assisted brain tumor segmentation under the constraint of largest information gain per additional MRI sequence, *Neuroscience Informatics*, 100053.
- Kumar, R., Gupta, A., Arora, H. S., Pandian, G. N. and Raman, B., 2020, CGHF: A computational decision support system for glioma classification using hybrid radiomics-and stationary wavelet-based features, *IEEE Access*, 8, 79440-79458.
- Latif, G., Iskandar, D. A., Alghazo, J. M. and Mohammad, N., 2018, Enhanced MR image classification using hybrid statistical and wavelets features, *IEEE Access*, 7, 9634-9644.
- Leonardo, M. M., Carvalho, T. J., Rezende, E., Zucchi, R. and Faria, F. A., 2018, Deep feature-based classifiers for fruit fly identification (Diptera: Tephritidae), *31st SIBGRAPI Conference on Graphics, Patterns and Images, SIBGRAPI-2018, Foz do Iguaçu*, 41-47.

- Li, Q., Yang, Y., Guo, Y., Li, W., Liu, Y., Liu, H. and Kang, Y., 2021, Performance evaluation of deep learning classification network for image features, *IEEE Access*, 9, 9318-9333.
- Li, P., Wu, W., Liu, L., Serry, F. M., Wang, J. and Han, H., 2022, Automatic brain tumor segmentation from Multiparametric MRI based on cascaded 3D U-Net and 3D U-Net++, *Biomedical Signal Processing and Control*, 78, 103979.
- Lu, J., Behbood, V., Hao, P., Zuo, H., Xue, S. and Zhang, G., 2015, Transfer learning using computational intelligence: A survey, *Knowledge-Based Systems*, 80, 14-23.
- Mahdianpari, M., Salehi, B., Rezaee, M., Mohammadimanesh, F. and Zhang, Y., 2018, Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery, *Remote Sensing*, 10 (7), 1119.
- Materka, A. and Strzelecki, M., 1998, Texture analysis methods—a review, Technical University of Lodz, Institute of Electronics, COST B11 report, 10 (1.97), 4968.
- Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L., Gerstner, E., Weber, M-A., Arbel, T., Avants, B. B., Ayache, N., Buendia, P., Collins, D. L., Cordier, N., Corso, J. J., Criminisi, A., Das, T., Delingette, H., Demiralp, Ç., Durst, C. R., Dojat, M., Doyle, S., Festa, J., Forbes, F., Geremia, E., Glocker, B., Golland, P., Guo, X., Hamamci, A., Iftekharuddin, K. M., Jena, R., John, N. M., Konukoglu, E., Lashkari, D., Mariz, J. A., Meier, R., Pereira, S., Precup, D., Price, S. J., Raviv, T. R., Reza, S. M. S., Ryan, M., Sarikaya, D., Schwartz, L., Shin, H-C., Shotton, J., Silva, C. A., Sousa, N., Subbanna, N. K., Szekely, G., Taylor, T. J., Thomas, O. M., Tustison, N. J., Unal, G., Vasseur, F., Wintermark, M., Ye, D. H., Zhao, L., Zhao, B., Zikic, D., Prastawa, M., Reyes, M. and Van Leemput, K., 2014, The multimodal brain tumor image segmentation benchmark (BRATS), *IEEE Transactions on Medical Imaging*, 34 (10), 1993-2024.

- Ofir, A. and Ben-Artzi, G., 2022, SMM-Conv: Scalar matrix multiplication with zero packing for accelerated convolution, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (CVPR-2022)*, 3067-3075.
- Orenstein, E. C. and Beijbom, O., 2017, Transfer learning and deep feature extraction for planktonic image data sets, *2017 IEEE Winter Conference on Applications of Computer Vision, WACV-2017, Santa Rosa*, 1082-1088.
- O'Shea, K. and Nash, R., 2015, An introduction to convolutional neural networks, *arXiv preprint*, arXiv:1511.08458.
- Özkan, İ. N. İ. K. and Ülker, E., 2017, Derin öğrenme ve görüntü analizinde kullanılan derin öğrenme modelleri, *Gaziosmanpaşa Bilimsel Araştırma Dergisi*, 6 (3), 85-104.
- Rao, A. S., Nguyen, T., Palaniswami, M. and Ngo, T., 2021, Vision-based automated crack detection using convolutional neural networks for condition assessment of infrastructure, *Structural Health Monitoring*, 20 (4), 2124-2142.
- Saba, T., Mohamed, A. S., El-Affendi, M., Amin, J. and Sharif, M., 2020, Brain tumor detection using fusion of hand crafted and deep learning features, *Cognitive Systems Research*, 59, 221-230.
- Sakalli, G. and Koyuncu, H., Discrimination of electrical motor faults in thermal images by using first-order statistics and classifiers, *2022 IEEE International Congress on Human-Computer Interaction, Optimization and Robotic Applications, HORA-2022, Ankara*, 1-5.
- Sharif, M., Amin, J., Nisar, M. W., Anjum, M. A., Muhammad, N. and Shad, S. A., 2020, A unified patch based method for brain tumor detection using features fusion, *Cognitive Systems Research*, 59, 273-286.
- Simonyan, K. and Zisserman, A., 2014, Very deep convolutional networks for large-scale image recognition, *arXiv preprint*, arXiv:1409.1556.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z., 2016, Rethinking the inception architecture for computer vision, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR-2016, Las Vegas, 2818-2826.

Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A. A., 2017, Inception-v4, inception-resnet and the impact of residual connections on learning, *Thirty-first AAAI Conference on Artificial Intelligence*, AAAI-2017, San Francisco, 1-12.

Tabian, I., Fu, H. and Sharif Khodaei, Z., 2019, A convolutional neural network for impact detection and characterization of complex composite structures, *Sensors*, 19 (22), 4933.

Yosinski, J., Clune, J., Bengio, Y. and Lipson, H., 2014, How transferable are features in deep neural networks?, *Advances in Neural Information Processing Systems*, 27.

EKLER

EK-1 DenseNet201 Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=densenet201;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
    lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
    newsoftmaxLayer = softmaxLayer('Name','new_softmax');
    lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
    newClassLayer = classificationLayer('Name','new_classoutput');
    lgraph =
replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

    % Preprocessing İşlemi
    imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);

```

```

imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

% Training Hiperparametre Ayarlamaları
options = trainingOptions('rmsprop',...
    'MaxEpochs',100,'MiniBatchSize',32,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.01, ...
    'LearnRateDropFactor',0.2, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augmentation ve Batch Ayarlama
auimds = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auimds,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()== 'HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-2 InceptionResNetV2 Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=inceptionresnetv2;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
    lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
    newsoftmaxLayer = softmaxLayer('Name','new_softmax');
    lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
    newClassLayer = classificationLayer('Name','new_classoutput');
    lgraph =
    replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

    % Preprocessing İşlemi
    imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);
    imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

    % Training Hiperparametre Ayarlamaları

```



```

options = trainingOptions('adam',...
    'MaxEpochs',100,'MiniBatchSize',16,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.0001, ...
    'LearnRateDropFactor',0.4, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augumentation ve Batch Ayarlama
auidms = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auidms,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()=='HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-3 InceptionV3 Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=inceptionv3;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
    lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
    newsoftmaxLayer = softmaxLayer('Name','new_softmax');
    lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
    newClassLayer = classificationLayer('Name','new_classoutput');
    lgraph =
    replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

    % Preprocessing İşlemi
    imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);
    imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

    % Training Hiperparametre Ayarlamaları

```

```

options = trainingOptions(sgdm,...
    'MaxEpochs',100,'MiniBatchSize',16,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.0001, ...
    'LearnRateDropFactor',0.6, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augumentation ve Batch Ayarlama
auidms = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auidms,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()=='HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-4 ResNet50 Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=resnet50;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
newsoftmaxLayer = softmaxLayer('Name','new_softmax');
lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
newClassLayer = classificationLayer('Name','new_classoutput');
lgraph =
replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

% Preprocessing İşlemi
imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);
imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

% Training Hiperparametre Ayarlamaları

```

```

options = trainingOptions('rmsprop',...
    'MaxEpochs',100,'MiniBatchSize',32,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.001, ...
    'LearnRateDropFactor',0.8, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augumentation ve Batch Ayarlama
auidms = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auidms,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()=='HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-5 ResNet101 Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=resnet101;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
    lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
    newsoftmaxLayer = softmaxLayer('Name','new_softmax');
    lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
    newClassLayer = classificationLayer('Name','new_classoutput');
    lgraph =
    replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

    % Preprocessing İşlemi
    imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);
    imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

    % Training Hiperparametre Ayarlamaları

```

```

options = trainingOptions('sgdm',...
    'MaxEpochs',100,'MiniBatchSize',32,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.0001, ...
    'LearnRateDropFactor',0.2, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augumentation ve Batch Ayarlama
auimds = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auimds,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()=='HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-6 SqueezeNet Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=squeezenet;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
    lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
    newsoftmaxLayer = softmaxLayer('Name','new_softmax');
    lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
    newClassLayer = classificationLayer('Name','new_classoutput');
    lgraph =
    replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

    % Preprocessing İşlemi
    imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);
    imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

    % Training Hiperparametre Ayarlamaları

```



```

options = trainingOptions('rmsprop',...
    'MaxEpochs',100,'MiniBatchSize',32,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.01, ...
    'LearnRateDropFactor',0.4, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augmentation ve Batch Ayarlama
auidms = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auidms,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()=='HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-7 VGG19 Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=vgg19;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
    lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
    newsoftmaxLayer = softmaxLayer('Name','new_softmax');
    lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
    newClassLayer = classificationLayer('Name','new_classoutput');
    lgraph =
    replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

    % Preprocessing İşlemi
    imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);
    imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

    % Training Hiperparametre Ayarlamaları

```

```

options = trainingOptions('sgdm',...
    'MaxEpochs',100,'MiniBatchSize',16,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.01, ...
    'LearnRateDropFactor',0.2, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augumentation ve Batch Ayarlama
auimds = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auimds,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()=='HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-8 Xception Mimarisi MATLAB İşletim Kodu

```

clc;close all;clear;

%-----Cross Validation Temelli-----%
%% İmgeleri Yükleme ve Fold Belirleme
digitDatasetPath = fullfile('D:\veri yolu.....');
imds = imageDatastore(digitDatasetPath, ...
    'IncludeSubfolders',true,'LabelSource','foldernames');

% Etiketleri ve Herbir Etiket Verisini Tespit Etme
total_split=countEachLabel(imds)

% Toplam Veri Sayısını Tespit Etme
num_images=length(imds.Labels);

% Fold Sayısını Atama
num_folds=2;

%% Herbir Fold İçin Döngü İşlemi
for fold_ind=1:num_folds
    fprintf('Processing %d among %d folds \n',fold_ind,num_folds);

    % Test Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    test_ind=fold_ind:num_folds:num_images;

    % Test Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTest = subset(imds,test_ind);

    % Train Veri İndislerinin Şimdiki Fold'da Belirlenmesi
    train_ind=setdiff(1:length(imds.Files),test_ind);

    % Train Verilerinin Atanması --> Şimdiki Fold İçinde
    imdsTrain = subset(imds,train_ind);

    % ResNet Mimarisi
    net=xception;
    lgraph = layerGraph(net);
    clear net;

    % Kategori Sayısının Belirlenmesi
    numClasses = numel(categories(imdsTrain.Labels));

    % Yeni Öğrenebilen Katman Atama
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);

% Ağa Dair Fully Connected Katmanların Yenileri İle değiştirilmesi
    lgraph = replaceLayer(lgraph,'fc1000',newLearnableLayer);
    newsoftmaxLayer = softmaxLayer('Name','new_softmax');
    lgraph = replaceLayer(lgraph,'fc1000_softmax',newsoftmaxLayer);
    newClassLayer = classificationLayer('Name','new_classoutput');
    lgraph =
    replaceLayer(lgraph,'ClassificationLayer_fc1000',newClassLayer);

    % Preprocessing İşlemi
    imdsTrain.ReadFcn = @(filename)preprocess_islemi(filename);
    imdsTest.ReadFcn = @(filename)preprocess_islemi(filename);

    % Training Hiperparametre Ayarlamaları

```

```

options = trainingOptions('adam',...
    'MaxEpochs',100,'MiniBatchSize',32,...
    'Shuffle','every-epoch', ...
    'InitialLearnRate',0.0001, ...
    'LearnRateDropFactor',0.6, ...
    'Verbose',false, ...
    'Plots','training-progress');

% Data Augumentation ve Batch Ayarlama
auimds = augmentedImageDatastore([224 224],imdsTrain);

% Eğitim
netTransfer = trainNetwork(auimds,lgraph,options);

% Test Verilerinin Boyut Ayarlaması (Batch Müsadesi İçin)
augtestimds = augmentedImageDatastore([224 224],imdsTest);

% Test İmgelerinde Tahminin Sağlanması
[predicted_labels(test_ind),posterior(test_ind,:)] =
classify(netTransfer,augtestimds);

% Gereksiz Değişkenlerin Silinmesi
clearvars -except fold_ind num_folds num_images predicted_labels
posterior imds netTransfer;

end

%% Çıktıların Eldesi
actual_labels=imds.Labels;
figure;plotconfusion(actual_labels,predicted_labels')
title('Confusion Matrix: ResNet');

%% Sonuç Elde Etme Bölümü
ACTUAL=actual_labels;
PREDICTED=predicted_labels';
indis = (ACTUAL()=='HGG');
p = length(ACTUAL(indis));
n = length(ACTUAL(~indis));
N = p+n;
tp = sum(ACTUAL(indis)==PREDICTED(indis));
tn = sum(ACTUAL(~indis)==PREDICTED(~indis));
accuracy = 100*((tp+tn)/N);

```

EK-9 Veri Adaptasyon MATLAB Kodu

```
%% Veri 3B Adapte Değilse Adaptasyonun Sağlanması  
function Iout = preprocess_islemi(filename)  
I = imread(filename);  
if ~ismatrix(I)  
    I=rgb2gray(I);  
end  
Iout = cat(3,I,I,I);  
end
```



EK-10 Tez Çalışması Kapsamında Hazırlanan Eser Tanımı

Hajmohamad, A. and Koyuncu, H., 2023, 3t2FTS: A Novel Feature Transform Strategy to Classify 3D MRI Voxels and its Application on Tumor Grading, *Under review in an indexed journal*.

