



FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI

HÜCRE-DAMAR SEGMENTASYONUNDA
KULLANILAN DERİN SİNİR AĞI MODELLERİNİN
HİPER-PARAMETRE OPTİMİZASYONU

DOKTORA TEZİ

ZEKİ KUŞ

İSTANBUL, 2023



FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI

HÜCRE-DAMAR SEGMENTASYONUNDA
KULLANILAN DERİN SİNİR AĞI MODELLERİNİN
HİPER-PARAMETRE OPTİMİZASYONU

DOKTORA TEZİ

ZEKİ KUŞ
(191221002)

Danışman
(Dr. Öğr. Üyesi Berna Kiraz)

İSTANBUL, 2023

24/08/2023

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği doktora programı öğrencisi 191221002 numaralı **Zeki KUŞ**'un, Dr. Öğr. Üyesi Berna KİRAZ danışmanlığında hazırladığı "*Hücre-Damar Segmentasyonunda Kullanılan Derin Sinir Ağı Modellerinin Hiperparametre Optimizasyonu*" konulu tezi ile ilgili Tez Savunma Sınavı, 24.08.2023 Perşembe günü saat 16:00'da yapılmış, sorulara alınan cevaplar sonunda adayın tezinin **Kabulüne Oy Birliği** ile karar verilmiştir.

Tez adı değişikliği yapılması halinde: Tez adının
.....
şeklinde değiştirilmesi uygundur.

Jüri Üyesi	Karar
1. Dr. Öğr. Üyesi Berna KİRAZ (Danışman)	KABUL.....
2. Prof. Dr. A. Yılmaz ÇAMURCU	KABUL
3. Prof. Dr. Serhat ÖZEKES	KABUL.....
4. Prof. Dr. Ayşe Şima UYAR	KABUL.....
5. Doç. Dr. Buket DOĞAN	KABUL.....
6. (İkinci Danışman)*.....	

*2. Danışman varsa doldurulması gerekmektedir.

ETİK BİLDİRİM

Bu tezin yazılmasında bilimsel ahlak kurallarına uyulduğunu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, tezin herhangi bir kısmının bağlı olduğum üniversite veya bir başka üniversitedeki başka bir çalışma olarak sunulmadığını beyan ederim.

Zeki Kuş

HÜCRE-DAMAR SEGMENTASYONUNDA KULLANILAN DERİN SİNİR AĞI MODELLERİNİN HİPER-PARAMETRE OPTİMİZASYONU

Zeki Kuş

ÖZET

Derin sinir ağları (DSA), medikal görüntülerin segmentasyonu için önemli bir araçtır. Ancak, DSA'ların performansı, seçilen mimari ve hiper parametrelere bağlıdır. Bu hiper parametreler manuel olarak seçilebilir, ancak bu zaman alıcı ve zor olabilir. Bu nedenle, Sinir Mimarisi Arama (NAS) yöntemleri geliştirilmiştir. Bu çalışmada ilk olarak, UNAS-Net adlı yeni bir NAS yöntemi önerilmiştir. UNAS-Net, Diferansiyel Evrim (DE) ve Yerel Arama (LS) gibi meta-sezgisel yöntemlerle optimize edilmiştir. Önerilen yöntem, Optofil ve Hücre Çekirdeği veri kümeleri üzerinde test edilmiş ve U-Net mimarisinden segmentasyon performansı ve hesaplama karmaşıklığı açısından daha iyi performans sergilemiştir. İkinci olarak, beyin damarı segmentasyonu için diferansiyel evrim tabanlı NAS yaklaşımları önerilmiştir. Bu yaklaşımlar, medikal görüntü segmentasyonunda yaygın olarak kullanılan U-Net ve Attention U-Net ağlarını temel almıştır. Arama yöntemi olarak, Geleneksel DE ve karışıklık tabanlı diferansiyel gelişim (ODE) yöntemleri seçilmiştir. Deneyler, vesseINN ve KUVESG olmak üzere iki farklı herkese açık beyin damar segmentasyonu veri kümesi üzerinde gerçekleştirilmiştir. Önerilen yöntemler, farklı segmentasyon metrikleri açısından daha iyi segmentasyon performansı sağlamış ve temel yöntemlere göre 9.15 kat daha karmaşık mimariler üretmiştir. Retina damar segmentasyonu (RDS), retina hastalıklarının belirlenmesine ve izlenmesine yardımcı olduğu için medikal görüntü analizinde çok önemlidir ve otomatik yöntemlere ihtiyaç duyulmaktadır. Üçüncü olarak, RDS problemi için yüksek segmentasyon performansına ve daha düşük çıkarım süresine sahip mimarileri keşfeden yeni bir NAS yöntemi önerilmiştir: MedUNAS. En iyi ağ yapısını aramak için ODE ve

genetik algoritma (GA) yöntemleri kullanılmış ve ayrıca ayrık ve sürekli kodlama stratejileri karşılaştırılmıştır. Önerilen yöntemler, 15 kata kadar daha az parametreye sahip ağlarla dört veri kümesinde temel U-Net'ten daha iyi performans göstermiştir. Ayrıca, oluşturulan ağların yeni medikal görevlere etkili bir şekilde uyarlanabileceği ortaya konulmuştur. Son olarak, tahmin ediciler ile NAS sürecinin birlikte gerçekleştirildiği BaDENAS yaklaşımı önerilmiş ve RDS problemi için test edilmiştir. Sonuçlar, BaDENAS'ın temel alınan yöntemleri model karmaşıklığı, segmentasyon performansı ve yakınsama hızı açısından iyileştirdiğini ortaya koymuştur.

Anahtar kelimeler: Sinir Mimarisi Arama, Hücre Segmentasyonu, Damar Segmentasyonu, Derin Öğrenme, Medikal Görüntü Segmentasyonu

HYPER-PARAMETER OPTIMIZATION OF DEEP NEURAL NETWORKS FOR CELL-VESSEL SEGMENTATION

Zeki Kuş

ABSTRACT

Deep neural networks (DNNs) are important methods for the segmentation of medical images. However, the performance of DNNs depends on the architecture and hyperparameters. These hyperparameters can be selected manually, which can be time-consuming and difficult. Therefore, Neural Architecture Search (NAS) methods are developed. First, we propose a new NAS method called UNAS-Net. UNAS-Net is optimized by meta-heuristics such as Differential Evolution (DE) and Local Search (LS). UNAS-Net is evaluated on the Optofil and Cell Nucleus datasets and outperforms the U-Net regarding segmentation performance and computational complexity. Second, DE-based NAS approaches for brain vessel segmentation are proposed. These approaches are based on U-Net and Attention U-Net, which are widely used in medical image segmentation. Traditional DE and opposition-based DE (ODE) are chosen as search methods. The experiments are performed on two publicly available cerebrovascular segmentation datasets: vesseINN and KUVESG. The proposed methods achieved better segmentation performance in different segmentation metrics and generated 9.15 times more complex architectures than the baselines. Retinal vessel segmentation (RVS) is very important in medical image analysis as it helps to identify and monitor retinal diseases, and automated methods are needed. Third, a new NAS method is proposed for the RVS that discovers architectures with high segmentation performance and lower inference time: MedUNAS. ODE and genetic algorithm (GA) methods are used to search for the best network structure, and also discrete and continuous encoding strategies are compared. The proposed methods outperformed the baseline U-Net on four datasets with networks with up to 15 times fewer parameters. Furthermore, it is demonstrated

that the generated networks can be effectively adapted to new medical tasks. Finally, the BaDENAS approach, which combines predictors with the NAS process, is proposed and tested for the RVS problem. The results show that BaDENAS improves the baselines regarding model complexity, segmentation performance and convergence speed.

Keywords: Neural Architecture Search, Cell Segmentation, Vessel Segmentation, Deep Learning, Medical Image Segmentation



ÖNSÖZ

Yapılan tez çalışmasında, otomatik hücre-damar segmentasyonu gerçekleştiren derin sinir ağı mimarilerinin aranması hedeflenmiş ve bu kapsamda üç farklı yeni sinir mimarisi arama yöntemi önerilmiştir. Önerilen yöntemler farklı hücre, beyin damar ve retinal kan damarı segmentasyonu problemleri üzerinde test edilmiş ve daha az karmaşık ağlar ile rekabetçi sonuçlar elde edilmiştir.

Bu tez çalışmasının gerçekleştirilmesinde, değerli bilgilerini benimle paylaşmaktan çekinmeyen, tez sürecinin tüm aşamalarında danışmanlığa ihtiyaç duyduğum her an kıymetli zamanını ayırıp sabırla ve büyük bir ilgiyle fayda sağlayabilmek adına elinden gelenden fazlasını yapan, her sorun yaşadığımda yanına çekinmeden gidebildiğim, doktora çalışmamın zor zamanlarında olumlu konuşmaları ile beni motive eden, akademik ahlak ve iyi bir insan olmak adına çok şey öğrendiğim, güler yüzünü ve samimiyetini benden esirgemeyen, danışmanım olduğu için kendimi şanslı hissettiğim ve bana kazandırdığı değerli bilgilerden her zaman faydalanacağım çok değerli danışman hocam Dr. Öğr. Üyesi Berna KİRAZ'a teşekkürü borç bilirim.

Prof. Dr. Ali Yılmaz ÇAMURCU ve Prof. Dr. Serhat ÖZEKES'e ayırdıkları zaman, gösterdikleri ilgi, yararlı yorum ve önerileri için teşekkür ederim. Ayrıca sözlü savunma komitemin diğer üyeleri Prof. Dr. Ayşe Şima Uyar ve Doç. Dr. Buket Doğan'a savunma komitemde yer almayı kabul ettikleri ve zaman ayırdıkları için teşekkür ederim.

Tez önerisi aşamasında yeni bir konuya başlamak için beni teşvik eden, destekleyen, yardımlarını esirgemeyen ve birlikte çalışma fırsatı bulduğum değerli hocam Dr. Öğr. Üyesi Musa AYDIN'a ayrıca teşekkür ederim.

Tez çalışmanın oluşturulmasında ve geliştirilmesinde bilgisini, tecrübesini ve değerli zamanını esirgemeyen değerli hocam Prof. Dr. Burhanettin CAN'a teşekkür ederim.

Çalışma konusunun belirlenmesinde ve çalışmanın hazırlanma sürecinin her aşamasında bilgisini, tecrübesini ve değerli zamanını esirgemeyerek bana her fırsatta yardımcı olan, farklı disiplinler ile çalışma imkânı sunan ve çalışma disiplini adına çok şey öğrendiğim değerli hocam Prof. Dr. Alper KİRAZ'a teşekkür ederim.

Son olarak, hayatım boyunca benden hiçbir zaman desteğini esirgemeyen bu hayattaki en büyük şansım olan aileme yapmış oldukları tüm fedakarlıklar için sonsuz teşekkürler ederim.

Ağustos, 2023

Zeki Kuş



İÇİNDEKİLER

ÖZET.....	iv
ABSTRACT	vi
ÖNSÖZ.....	viii
TABLO LİSTESİ	xiii
ŞEKİL LİSTESİ.....	xv
KISALTMALAR	xvii
GİRİŞ	1
BİRİNCİ BÖLÜM.....	9
1. HÜCRE VE DAMAR SEGMENTASYONU	9
1.1. GÖRÜNTÜ SEGMENTASYONU	9
1.1.1. Derin Öğrenme Tabanlı Yöntemler ile Medikal Görüntülerin Segmentasyonu.....	10
1.1.1.1. Evrişimli Sinir Ağı (ESA)	10
1.1.1.2. Tam Evrişimli Sinir Ağı.....	11
1.1.1.3. U-Net.....	12
1.2. HÜCRE SEGMENTASYONU	14
1.3. DAMAR SEGMENTASYONU.....	16
1.4. SİNİR MİMARİSİ ARAMA.....	18
1.5. META-SEZGİSELLER	21
1.5.1. Diferansiyel Gelişim	21
1.5.2. Karşıtlık Temelli Diferansiyel Gelişim	23
1.6. VERİ KÜMELERİ	25
1.6.1. Optofil Veri Kümesi.....	25
1.6.2. Hücre Çekirdeği Veri Kümesi	26
1.6.3. KUVESG ve vesselNN Veri Kümeleri	27
1.6.4. DRIVE	28
1.6.5. CHASE_DB1	28
1.6.6. DCA1.....	28
1.6.7. CHUAC.....	28
İKİNCİ BÖLÜM	30
2. UNAS-NET	30
2.1. HÜCRE SEGMENTASYONU İÇİN UNAS-NET	30
2.1.1. Giriş.....	30
2.1.2. Yöntem.....	32
2.1.2.1. Arama Uzayı.....	32

2.1.2.2. Mimari Kodlama	34
2.1.2.3. Meta-Sezgisel Yöntemler.....	37
2.1.3. Deneyisel Çalışmalar	37
2.1.3.1. Veri Kümesi.....	38
2.1.4. Sonuçlar ve Tartışma	38
2.2. BEYİN DAMAR SEGMENTASYONU İÇİN UNAS-NET	43
2.2.1. Giriş.....	43
2.2.2. Yöntem.....	46
2.2.2.1. Arama Uzayı.....	46
2.2.2.2. Mimari Kodlama	48
2.2.2.3. Attention UNAS-Net.....	50
2.2.3. Meta-Sezgisel Yöntemler.....	51
2.2.4. Deneyisel Çalışmalar	51
2.2.4.1. URes-Net.....	52
2.2.4.2. Veri Kümesi.....	52
2.2.4.3. Performans Ölçütleri.....	52
2.2.4.4. Geliştirme Detayları	53
2.2.5. Sonuçlar ve Tartışma	55
2.2.5.1. U-Net Modellerinin Karşılaştırılması	55
2.2.5.1. Ablasyon Çalışmaları	58
2.2.5.1.1. Kayıp Fonksiyonu	59
2.2.5.1.2. Model Budama	60
2.2.5.1.3. Eşik Değeri Kıyaslaması.....	61
2.2.5.1.4. Genelleştirme Performansı	63
ÜÇÜNCÜ BÖLÜM	66
3. MedUNAS	66
3.1. MedUNAS	66
3.1.1. Giriş.....	66
3.1.2. Yöntem.....	70
3.1.2.1. Arama Uzayı.....	70
3.1.2.2. Kodlama Şeması.....	73
3.1.2.3. Arama Yöntemleri	76
3.1.2.3.1. Genetik Algoritma	76
3.1.3. Deneyisel Çalışmalar	77
3.1.3.1. Veri Kümeleri.....	77
3.1.3.2. Performans Ölçütleri.....	78
3.1.3.3. Geliştirme Detayları	78
3.1.4. Sonuçlar ve Tartışma	79
3.1.4.1. Literatürdeki NAS Çalışmaları ile Karşılaştırma	80
3.1.4.2. Literatürdeki Son Teknoloji Çalışmalar ile Karşılaştırma	83
3.1.4.3. Ablasyon Çalışmaları	87

3.1.4.3.1. Kayıp Fonksiyonları	88
3.1.4.3.2. İnce Ayar Olmadan Çapraz-Eğitim Değerlendirmesi.....	89
3.1.4.3.3. İnce Ayar ile Çapraz-Eğitim Değerlendirmesi.....	89
3.1.4.3.4. Genelleştirme Performansı	91
3.2. FARKLI KODLAMA YÖNTEMLERİNİN KARŞILAŞTIRILMASI	92
DÖRDÜNCÜ BÖLÜM	97
4. BaDENAS	97
4.1. Giriş	97
4.2. Yöntem	97
4.2.1. Arama Uzayı.....	97
4.2.2. Kodlama Şeması.....	97
4.2.3. Arama Yöntemleri	98
4.2.3.1. Diferansiyel Gelişim	98
4.2.3.2. Sinir Mimarileri için Bayes Optimizasyonu (BANANAS).....	98
4.2.3.3. BaDENAS.....	99
4.2.4. Deneysel Çalışmalar	101
4.2.4.1. Veri Kümesi	101
4.2.4.2. Geliştirme Detayları	101
4.2.4.3. Performans Ölçütleri	102
4.2.5. Sonuçlar ve Tartışma.....	102
4.2.5.1. Literatürdeki NAS Çalışmaları ile Karşılaştırma	102
SONUÇ.....	105
KAYNAKÇA	110

TABLO LİSTESİ

Sayfa

Tablo 2.1: Hücreler için seçilebilecek olası operasyonlar.....	34
Tablo 2.2: Segmentasyon performansı ve hesaplama karmaşıklığı açısından farklı yöntemlerin karşılaştırılması. FLOPs değerleri gigabayt cinsinden verilmiştir.....	41
Tablo 2.3: Ara köşeler için seçebileceği olası işlemler	48
Tablo 2.4: Seçilen performans ölçütlerinin kısa özeti. Kategori sütunu, her bir performans ölçütünün ait olduğu performans ölçütü grubunu gösterir.....	53
Tablo 2.5: Seçilen yöntemlerin farklı değerlendirme ölçütleri açısından karşılaştırılması. Her satırda, en iyi sonuç veren yaklaşım koyu renkle işaretlenmiştir.	56
Tablo 2.6: Model karmaşıklığı, çıkarım süresi ve boyutu açısından farklı yöntemlerin karşılaştırılması. FLOPs değerleri gigabayt cinsinden; # Parametre ve Modelin Diskteki Boyutu ise megabayt cinsinden, CPU zamanı saniye cinsinden verilmiştir.	57
Tablo 2.7: UNAS-Net ODE ve Attention-UNAS-Net ODE için farklı kayıp fonksiyonlarının performans karşılaştırması.....	60
Tablo 2.8: Farklı hücre sayısı değerleri için performans karşılaştırması	61
Tablo 2.9: Eşik değeri karşılaştırmaları	62
Tablo 2.10: Önerilen yöntemlerin Optofit veri kümesi üzerindeki performans sonuçları. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir. U-ODE: UNAS-Net ODE; Att-ODE: Attention-UNAS-Net ODE	64
Tablo 2.11: DRIVE veri kümesi üzerinde farklı NAS çalışmalarının performans sonuçları. Kırmızı ve mavi renkler her bir performans ölçütü için en iyi iki değeri göstermektedir. U-ODE: UNAS-Net ODE; Att-ODE: Dikkat-UNAS-Net ODE.....	65
Tablo 3.1: O kümesi için seçilebilecek olası operasyonlar	72
Tablo 3.2: Önerilen yöntemlerin farklı NAS çalışmaları ile DRIVE veri kümesi üzerinde karşılaştırılması. Kırmızı ve mavi renkler sırasıyla her bir ölçüt için elde edilen en iyi iki sonucu göstermektedir.	81
Tablo 3.3: Önerilen yöntemlerin farklı NAS çalışmaları ile CHASE_DB1 üzerinde karşılaştırılması. Kırmızı ve mavi renkler sırasıyla her bir ölçüt için elde edilen en iyi iki sonucu göstermektedir.	81
Tablo 3.4: DRIVE ve CHASE_DB1 üzerinde önerilen yöntemlerin son teknoloji yöntemlerle karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir. Att-U: Attention U-Net. **: İlgili yöntemin U-şekli ağ olduğunu ifade etmektedir.....	84

Tablo 3.5: DCA1 ve CHUAC üzerinde önerilen yöntemlerin son teknoloji yöntemlerle karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir. Att-U: Attention U-Net. **: İlgili yöntemin U-şekilli ağ olduğunu ifade etmektedir.....	85
Tablo 3.6: DRIVE ve CHASE_DB1 veri kümeleri için ince damar segmentasyonu kıyaslaması.....	87
Tablo 3.7: MedUNAS ODE için farklı kayıp fonksiyonlarının karşılaştırılması	88
Tablo 3.8: MedUNAS ODE için ince ayar olmadan çapraz-eğitim değerlendirmesi sonuçları.....	89
Tablo 3.9: MedUNAS ODE için ince ayar ve ön-eğitim etkisinin çapraz-eğitim kümelerinde değerlendirilmesi.....	90
Tablo 3.10: MedUNAS ODE için ince ayar ile hücre ve beyin damar görüntüleri üzerinde geliştirme performansının değerlendirilmesi.	92
Tablo 3.11: MedUNAS ve UNAS-Net çalışmalarında önerilen kodlama şemalarının DRIVE ve CHASE_DB1 veri kümeleri için literatürdeki NAS çalışmaları ile karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir.....	93
Tablo 3.12: MedUNAS ve UNAS-Net çalışmalarında önerilen kodlama şemalarının DRIVE ve CHASE_DB1 veri kümeleri için literatürdeki son-teknoloji çalışmalar ile karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir.....	94
Tablo 4.1: Farklı yöntemlerin segmentasyon performansı, standart sapma ve karmaşıklık açısından karşılaştırılması. Kırmızı ve mavi renkler sırasıyla en iyi iki sonucu göstermektedir.	104

ŞEKİL LİSTESİ

Sayfa

Şekil 1.1: Evrişimli sinir ağlarının yapısı (Kuş, 2019).....	11
Şekil 1.2: TESA genel yapısı (Liu vd., 2021).....	12
Şekil 1.3: U-Net yapısı (Aydın, 2021).....	14
Şekil 1.4: Derin sinir ağı yöntemi ile hücre segmentasyonu işlemi (Aydın, (2021))	15
Şekil 1.5: Tekrarlayan hücre yapısı gösterimi (Ying vd. (2019)).....	20
Şekil 1.6: Optofil veri kümesi için orijinal ve gerçek referans görüntüleri	26
Şekil 1.7: Hücre çekirdeği veri kümesi için orijinal ve gerçek referans görüntüleri.	26
Şekil 1.8: vesseINN (mavi kesikli çizgi) ve KUVESG veri kümeleri için (kırmızı kesikli çizgi) orijinal ve gerçek referans görüntü çiftleri. Ölçek çubuğu 100 µm'yi gösterir.....	27
Şekil 1.9: Dört farklı retinal kan damarı veri kümesi için örnek görüntüler. Üst satır orijinal görüntüleri ve alt satır bu görüntülere karşılık gelen gerçek referans görüntülerini temsil eder.	29
Şekil 2.1: UNAS-Net'in genel yapısı. a) Hücre yapısı, b) UNAS-Net genel yapısı .	33
Şekil 2.2: Çözüm vektöründen UNAS-Net mimarisinin oluşturulması	36
Şekil 2.3: F1-Skoru hesaplaması	39
Şekil 2.4 IoU hesaplaması.....	40
Şekil 2.5: Dört farklı yöntemle yapılan deneyler sonucunda farklı hücre görüntüleri için elde edilen segmente edilmiş görüntüler. Kırmızı kareler, segmentasyonun yanlış yapıldığı bölgeleri temsil etmektedir.	42
Şekil 2.6: İki farklı veri kümesi için meta-sezgiseller tarafından oluşturulan en iyi UNAS-Net modelleri.	43
Şekil 2.7: UNAS-Net'in genel yapısı. Ölçek çubuğu 100 µm'yi gösterir	46
Şekil 2.8: Hücre tabanlı mikro arama uzayındaki kodlayıcı ve kod çözücü hücrelerin	47
Şekil 2.9: UNAS-Net Mimarisi için kodlama (encoding - sol kısım) ve eşleme (mapping) adımları: Adım 1: Komşuluk matrisi oluşturulur; Adım 2: Etiketler seçilir; Adım 3: İki hiper parametre için değerler seçilir; Adım 4: Komşuluk matrisi etiketler ile birleştirilir.....	50
Şekil 2.10: Attention-UNAS-Net'e genel bakış. (a) Attention-UNAS-Net Mimarisi, (b) Dikkat Kapısı. Ölçek çubuğu 100 µm'yi gösterir.	51
Şekil 2.11: Farklı medikal görüntü segmentasyonu modellerinin görsel olarak karşılaştırılması. İlk satır, her yöntem için hata haritalarını (error maps) (Beyaz: Gerçek pozitif pikseller (TP); Kırmızı: Yanlış negatif pikseller (FN); Mavi: Yanlış pozitif pikseller (FP)	59

Şekil 2.12: vesselNN veri seti üzerinde düşük ve yüksek eşik değeri analizi sonuçlarının grafiksel gösterimi. Ayrıntıları daha iyi görüntülemek için görüntülerde dört adet dikdörtgen kullanıldı.	63
Şekil 3.1: MedUNAS genel yapısı	71
Şekil 3.2: Hücre tabanlı mikro arama alanının bir özeti ve genel kodlayıcı ve kod çözücü hücrelerin mimarisi.	72
Şekil 3.3: MedUNAS mimarisi için kodlama adımlarına genel bakış. Mor kesikli çizgi GA adımlarını, mavi kesikli çizgi ise ODE adımlarını göstermektedir. Adım 1 yalnızca ODE için gerçekleştirilir.	74
Şekil 3.4: Genetik Algoritma ve Karşıtlık Temelli Diferansiyel Gelişim kullanılarak oluşturulan mimarilerin t-SNE izdüşümü. Seviyeler doğrulama doğruluk değerlerini göstermektedir. Daha yüksek seviyeler daha yüksek doğruluğa sahip çözümü temsil etmektedir.	83
Şekil 3.5: Çeşitli medikal görüntü segmentasyon modellerinin görsel karşılaştırması. Beyaz pikseller gerçek pozitif pikselleri, kırmızı pikseller yanlış negatif pikselleri ve mavi pikseller yanlış pozitif pikselleri göstermektedir.	87
Şekil 3.6: Farklı segmentasyon modelleri tarafından elde edilen görüntülerin görsel olarak karşılaştırılması. Kırmızı renk yanlış segmente edilen negatif pikselleri (FN), mavi renk yanlış segmente edilen pozitif pikselleri (FP) ve beyaz renk ise doğru segmente edilen pikselleri (TP) temsil etmektedir.	95
Şekil 3.7: Önerilen yöntemler için arama performansı kıyaslaması. Yöntemler model karmaşıklığı ve segmentasyon performansı açısından karşılaştırılmıştır. Görüntülerdeki koyu noktalar çözüm kalitesinin arttığını göstermektedir.	96
Şekil 4.1: Farklı yöntemlerin yakınsama performansı açısından karşılaştırılması ..	104

KISALTMALAR

DE	Diferansiyel Gelişim
DAG	Döngü İçermeyen Yönlü Çizgeler
EA	Evrimsel Algoritmalar
ESA	Evrişimli Sinir Ağı
GA	Genetik Algoritma
LS	Yerel Arama
NAS	Sinir Mimarisi Arama
ODE	Karşıtlık Temelli Diferansiyel Gelişim
TESA	Tam Evrişimli Sinir Ağı

GİRİŞ

Görüntü segmentasyonu, bir görüntünün, gri tonlama, renk, uzamsal doku ve geometrik şekiller gibi özelliklere göre birkaç ayrı alana bölünmesidir. Bu alanlar, aynı özelliklerle tutarlı veya benzerken, farklı alanlar arasında net bir farklılık gösterir. Görüntü segmentasyonu, segmentasyonun farklı kaba ve hassas detay seviyesine göre anlamsal segmentasyon, örnek segmentasyon ve panoramik segmentasyon olarak üç ayrı gruba ayrılabilir. Medikal görüntülerin segmentasyonu, anlamsal segmentasyon görevi olarak kabul edilir. Günümüzde, uydu görüntülerinin, medikal görüntülerin veya otonom araçların çalışması boyunca elde edilen görüntülerin segmentasyonu gibi farklı çalışma alanları mevcuttur (Ess vd., 2009; Geiger vd., 2012). Medikal görüntülerin segmentasyonu problemi için derin sinir ağları ile başarılı sonuçlar elde edilmiştir. Bu problem için özelleşmiş bir derin sinir ağı olan U-Net modeli (Ronneberger vd., 2015) medikal görüntülerin segmentasyonu için yoğun olarak kullanılmaktadır ve yapılan çalışmalarda temel alınan yöntem olarak seçilmektedir. Ancak, elle tasarlanmış olan bu derin sinir ağı modeli seçilmesi gereken birçok hiper-parametreye sahip, kompleks bir yapıdır. Bu nedenle, U-Net ağının otomatik optimizasyonunu gerçekleştirecek olan yöntemlere ihtiyaç giderek artmaktadır. Bu ihtiyaca bir çözüm üretilmesi amacıyla otomatik makine öğrenmesi probleminin alt kümesi olan sinir mimarisi arama çalışmaları önerilmiştir. Sinir mimarisi arama çalışmalarında, derin sinir ağının performansını dolaylı olarak etkileyen öğrenme oranı, yığın boyutu vb. gibi hiper-parametreler sabit tutulup sinir ağı mimarisinin optimize edilmesine, belirlenmesine odaklanılır. Üretilcek farklı sinir ağlarının sınırlarının belirlenmesi için bir arama uzayı kullanılır ve bu arama uzayı üzerinde farklı arama yöntemleri ile (genellikle sezgisel yöntemler) derin sinir ağı mimarilerinin otomatik olarak araştırılması hedeflenir. Bu sayede, verilen bir problem için tanımlanan arama uzayının sınırları içerisinde hem model karmaşıklığı hem de segmentasyon performansı açısından daha iyi sonuçlar veren ağların araştırılması hedeflenmektedir. Gerçekleştirilen sinir mimarisi arama problemlerinde evrimsel yaklaşımların arama yöntemi olarak daha başarılı sonuçlar verdiği

görülmüştür (Real vd., 2019; Chen vd., 2022). Gözlemlenen bu sonuçlar neticesinde, yapılan tez çalışmasında hücre ve damar görüntülerinin otomatik segmentasyonunun gerçekleştirilmesi amacıyla evrimsel yaklaşım temelli üç farklı yöntem önerilmiş ve bu yöntemler ile beş farklı çalışma gerçekleştirilmiştir. İlerleyen paragraflarda yapılan beş farklı çalışmaya giriş yapılmıştır.

Medikal görüntü segmentasyonu problemlerinden biri olan hücre segmentasyonu bu çalışmada ele alınmıştır. Hücre, tüm canlı organizmalarda temel, yapısal, işlevsel ve biyolojik birimdir. Hücreleri görüntüleme ve onlardan bilgi toplama, ‘görüntü sitometrisi’, ‘akış sitometrisi’ ve ‘hücre sıralayıcılar’ gibi çeşitli bilimsel alanlar için önemlidir (Chaffer vd., 2011). Yüksek çözünürlüklü floresan mikroskopundaki son gelişmeler sayesinde hücrelerin ve hücre altı parçalarının daha hassas görüntülenmesi mümkün hale geldi (Vonesch vd., 2006). Bilgisayarların artan hesaplama kabiliyeti ve derin öğrenme alanındaki gelişmeler tıbbi görüntüleme alanındaki çalışmaları hızlandırmıştır (Deshmukh & Mankar, 2014). Hücresel görüntülerin otomatik analizi, daha yüksek doğruluk ve verimlilikle hücre sayımı ve segmentasyon sağlar ve tekrarlanabilir bilgiler sunar. Öte yandan, uzman insanlar tarafından gerçekleştirilen el ile segmentasyon, daha fazla zaman alıcıdır ve öznel yorumlar nedeniyle daha fazla hataya açıktır. Bu nedenle, hücre görüntülerinin otomatik segmentasyonunu gerçekleştiren derin sinir ağlarına olan ihtiyaç her geçen gün artmaktadır. Medikal görüntülerin segmentasyonu için özelleşmiş bir derin öğrenme ağı olan U-Net (Ronneberger vd., 2015) ile otomatik hücre segmentasyonu için başarılı sonuçlar elde edilmiştir. Ancak, U-Net el ile tasarlanmış bir ağ yapısıdır ve optimize edilmesi gereken birçok hiper-parametreye sahiptir. Pek çok optimizasyon tekniği hem segmentasyon performansı hem de hesaplama karmaşıklığı açısından verimli olması gereken ağ mimarilerini bulmaya odaklanmıştır. “Sinir Ağı Mimarisi Arama” (NAS) olarak adlandırılan bu teknikler, çeşitli biyomedikal görüntüleme problemlerine uygulanmıştır. Bu nedenle, yapılan çalışmada U-Net mimarisinin optimizasyonu için yeni bir sinir mimarisi arama yaklaşımı önerilmiş ve UNAS-Net olarak isimlendirilmiştir. Önerilen yaklaşım, DE, ODE ve LS arama yöntemleri ile kullanılmış ve iki farklı hücre segmentasyonu veri kümesi üzerinde test edilmiştir: Optofil ve Hücre Çekirdeği veri kümesi. Sonuçlar

incelendiğinde DE tabanlı yöntemlerin daha az karmaşık ağlar ile her iki veri kümesi için de segmentasyon performansı açısından daha iyi ağları ürettiği görülmüştür.

Yapılan diğer bir çalışmada ise UNAS-Net'in performansı beyin damar segmentasyonu veri kümesi olan KUVESG üzerinde değerlendirilmiştir. Beyin, yüksek miktarda enerji tüketen kırılğan bir organdır. Beynin korunması kan-beyin bariyeri tarafından sağlanır ve enerji kaynakları serebral damar sistemi aracılığıyla sağlanır. Bu yapılardan herhangi birinin bozulmasının, Alzheimer hastalığı, multipl skleroz, serebral küçük damar hastalığı, vasküler demans dahil olmak üzere birçok nörodejeneratif bozuklukta hastalık mekanizmasına önemli bir katkısının olduğu kabul edilmektedir (Sweeney vd., 2018). Bu nedenle, serebral damarların görüntülenmesi ve analizi, biyomedikal araştırmalarda, serebrovasküler sistemin bozulmasına yol açan yeni mekanizmaları ortaya çıkarmak ve yeni tedavi hedefleri bulmak için sıklıkla kullanılır. Ancak, beyin damarları ağ benzeri tübüler bir yapıdır ve bu karmaşık yapıdaki uzunluk ve dallanma gibi damarla ilgili parametreleri manuel yöntemlerle güvenilir bir şekilde ölçmek zordur. Bu nedenle, bu zorlukların üstesinden gelmek için daha önce otomatik veya yarı otomatik yöntemler önerildi (Eladawi vd., 2017; Moccia vd., 2018). Bununla birlikte, yüksek ve düzensiz arka plan sinyali, kan damarlarının doğru tanımlanmasını engelleyebilen ve bu tür analizlerde artan hata oranına yol açabilen mikroskopide sıklıkla karşılaşılan bir sorundur. Bu nedenle, doğru segmentasyon, doğru bir damar analizi için kritik bir adımdır ve otomatik damar segmentasyonu gerçekleştirebilen sistemler olan ihtiyaç artmaktadır. Yapılan çalışmada, UNAS-Net ile beyin damar görüntülerinin otomatik segmentasyonu gerçekleştirilmiştir. Aynı zamanda, U-Net'in bir varyantı olan Attention U-Net üzerinde sinir mimarisi arama işlemi gerçekleştiren Attention UNAS-Net yapısı önerilmiştir. Önerilen yöntemler, segmentasyon performansı ve hesaplama karmaşıklığı açısından temel alınan U-Net ve Attention U-Net'ten daha iyi performans göstermiştir. Aynı zamanda bu yöntemler üzerinde kayıp fonksiyonu, hücre sayısı ve eşik değeri gibi etkenlerin sonuca olan etkisinin ölçülmesi amacıyla çeşitli ablasyon çalışmaları gerçekleştirilmiş ve önerilen yöntemlerin genelleştirme performansına olan etkisi incelenmiştir. Segmentasyon performansı ve hesaplama karmaşıklığındaki bu tür önemli gelişmeler, bu çalışmada önerilen yöntemlerin

meslekten olmayan kullanıcılar tarafından da yaygın olarak kullanılabilmesi açısından önemlidir.

Retinal kan damarı segmentasyonu, retina içerisindeki fundus görüntülerinden kan damarlarının çıkarılmasını içeren tıbbi görüntü analizinde önemli bir görevdir. Bu işlem, diyabetik retinopati ve yaşa bağlı makula dejenerasyonu da dahil olmak üzere retinal vasküler hastalıkların teşhisi ve izlenmesi için kritik öneme sahiptir (Abramoff vd., 2010). Retinal kan damarlarının doğru segmentasyonu, klinisyenlere damar morfolojisi ve kıvrımlılığı gibi değerli bilgiler sağlayarak bu hastalıkların tespit edilmesine ve yönetilmesine yardımcı olur (Li vd., 2021). Öneme rağmen, retinal kan damarlarının karmaşık yapısı ve değişken görünümü nedeniyle retinal damar segmentasyonu zordur (Wang vd., 2019). Bu nedenle, retinal damar segmentasyonu için gürbüz ve verimli algoritmalar geliştirmek, tıbbi görüntüleme aktif bir araştırma konusu haline gelmiştir. Bu probleme çözüm üretmek amacıyla yapılan bu çalışmada, U-şekilli ağlar için yeni bir sinir mimari arama yöntemi önerilmiş ve MedUNAS olarak isimlendirilmiştir. Önerilen çalışmada, UNAS-Net'ten farklı bir hücre tabanlı arama uzayı önerilmiştir. En iyi ağ yapısını aramak için de önerilen arama uzayı üzerinde ODE ve GA yöntemleri uygulanmıştır. Önerilen yöntemler DRIVE, CHASE_DB1, DCA1 ve CHUAC olmak üzere herkese açık olan dört adet retinal damar segmentasyonu veri kümesi üzerinde değerlendirilmiştir. Sonuçlar, segmentasyon performansı ve model karmaşıklığı açısından son teknoloji yöntemler ve sinir mimarisi arama çalışmaları ile karşılaştırılmıştır. Deneysel sonuçlara göre, önerilen yöntemler son teknoloji çalışmalara göre en az karmaşık modelleri (CHASE_DB1 hariç) üretmekte ve yine de oldukça rekabetçi sonuçlar elde etmektedir. Ayrıca, kayıp fonksiyonlarının etkisini değerlendirmek için ablasyon çalışmaları gerçekleştirilmiş ve modelin farklı alanlar için genelleştirme performansı ve çapraz veri kümelerindeki performansı incelenmiştir. Önerilen yaklaşımın, klinik uygulamada retinal damar segmentasyon sürecini otomatikleştirmek için değerli bir araç olarak hizmet edebilir ve klinisyenlerin çeşitli retina hastalıklarının ilerlemesini daha verimli ve doğru bir şekilde tanımlamasına ve izlemesine olanak sağlayabilir.

Bu çalışmada önerilen UNAS-Net ve MedUNAS yöntemleri hücre tabanlı mimariyi kullansalar da arama uzayı ve kodlama stratejileri bakımından birbirinden

ayrışmaktadır. Bu nedenle, yapılan tez çalışmasında hücre şeklindeki ağlar için farklı kodlama stratejilerinin performansı retinal kan damar segmentasyonu veri kümeleri üzerinde karşılaştırılmıştır. Adil bir karşılaştırma yapılabilmesi amacıyla her iki yöntem için de arama yöntemi olarak ODE seçilmiş ve farklı hücre yapılarının arama uzayında araştırılması için kullanılmıştır. UNAS-Net (Kus vd., 2022) ve MedUNAS yöntemlerinde kullanılan kodlama stratejilerinin karşılaştırıldığında UNAS-Net çalışmasında önerilen yöntemin en az %50 daha az parametre ile rekabetçi sonuçlar elde ettiği gözlemlenmiştir.

Yapılan son çalışmada ise NAS sürecinin verimliliğinin tahmin ediciler ile nasıl geliştirilebileceği araştırılmıştır. Tanımlanan arama uzayı ve çözüm kalitesinin değerlendirilmesi için seçilen ölçütün hesaplanma süresine bağlı olarak NAS çalışmalarının verimliliği değişkenlik gösterebilmektedir. Birçok derin öğrenme tabanlı NAS çalışmasında üretilen çözümlerin kalitesinin değerlendirilmesi uzun eğitim süreleri ile mümkün olabilmektedir. Bu durum, çalışma sürelerini uzatmakta ve makul sürelerde arama uzayında daha fazla çözümün ziyaret edilmesini zorlaştırmaktadır. Bu nedenle, tahmin ediciler ile NAS sürecinin birleştirilmesine duyulan ihtiyaç artmaktadır. Bu çalışmada, bu soruna çözüm üretebilmek amacıyla NAS ve tahmin edicileri birleştiren yeni bir yaklaşım önerilmiş ve BaDENAS olarak isimlendirilmiştir. Önerilen yöntem, BANANAS (White vd., 2021) ve DE yöntemlerinin avantajlarını birleştiren yeni bir NAS yaklaşımıdır. BaDENAS, retinal kan damarı segmentasyonu problemi için farklı NAS çalışmaları ile karşılaştırılmış ve daha az parametre ile rekabetçi sonuçlar elde etmiştir. Aynı zamanda, temel alınan BANANAS ve DE yöntemlerinden daha hızlı yakınsama sağlamış ve daha az karmaşık model ile yüksek segmentasyon başarısı göstermiştir.

Tez Çalışmasının Literatüre Katkısı:

Bu tez çalışmasının literatüre olan katkısı şu şekilde özetlenebilir:

1) Hücre ve damar görüntülerinin iki boyutlu bir şekilde otomatik segmentasyonunu gerçekleştiren iki farklı yeni sinir mimarisi arama yöntemi önerilmiştir: UNAS-Net ve MedUNAS. Önerilen yöntemler, hücre, beyin damar ve retinal kan damarı segmentasyonu problemleri için temel alınan yöntemlerin hem model karmaşıklığı hem de segmentasyon performansı açısından iyileştirebileceğini göstermiştir. Aynı zamanda, her iki yöntemde de kullanılan “Karşıtlık Temelli

Diferansiyel Gelişim Algoritması” bildiğimiz kadarıyla sinir mimarisi arama çalışmalarında ilk defa arama yöntemi olarak kullanılmıştır.

2) Tahmin ediciler (yapay sinir ağı, rastgele orman vb.) ile meta-sezgisel yöntemlerin sinir mimarisi arama problemi için birlikte kullanıldığı yeni bir yöntem önerilmiştir: BaDENAS. Elde edilen sonuçlar ile sinir mimarisi arama işlemi sırasında üretilen her bir çözümün gerçek kalitesinin ölçülmesine gerek kalmadan tahmini kalitenin ölçülmesiyle de yani daha az arama/değerlendirme maliyetiyle en az karmaşıklık ile yüksek rekabetçi sonuçların elde edilebileceği gösterilmiştir.

3) UNAS-Net ve MedUNAS yöntemlerinde önerilen farklı kodlama stratejileri (üretilen çözüm vektöründen sinir mimarisinin oluşturulması) karşılaştırılmıştır. Yapılan kıyaslama, sürekli ve ayrık arama uzayı seçiminin sinir mimarisi arama sürecine olan etkisinin karşılaştırılması amacıyla değerlidir. Aynı zamanda, önerilen yöntemlerin eğitim dışındaki farklı yeni medikal görevler üzerinde nasıl performans sergilediğinin gösterilmesi için çeşitli ablasyon çalışmaları gerçekleştirilmiş ve önerilen yöntemlerin kısa bir eğitim ile farklı alanlardaki segmentasyon görevlerine hızlı bir şekilde uyum sağlayabildiği görülmüştür. Bu sonuç, önerilen yöntemlerin farklı medikal görevler için genelleştirilebilir olması açısından önemlidir.

4) Literatürde, medikal görüntülerin segmentasyonu için sinir mimarisi arama işlemi gerçekleştiren çalışmaların sayısı azdır. Yapılan tez çalışmasında önerilen tüm yöntemler ile medikal görüntüler üzerinde sinir mimarisi arama işlemi gerçekleştirilmiştir. Bu tez çalışması, ilgili alanda yapılan çalışmaların artması adına önemlidir. Önerilen yöntemler medikal segmentasyon problemleri üzerinde test edilmiştir ancak bu yöntemler diğer segmentasyon, sınıflandırma vb. problemlere de kolaylıkla uygulanabilir.

5) Beyin damar görüntülerinin segmentasyonu için kullanılan KUVESG veri kümesi herkese açık yeni bir veri kümesi olarak diğer araştırmacılar ile paylaşılmıştır. Farklı araştırmacılar tarafından önerilen modellerin yaptığımız çalışma ile karşılaştırılabilmesi açısından bu paylaşım önemlidir. Benzer şekilde, bu alanda çalışan araştırmacıların önerdiğimiz yöntemler için elde edilen sonuçları yeniden üretebilmesi, önerilen yöntemleri farklı problemler üzerinde test edebilmesi

için UNAS-Net, MedUNAS ve BaDENAS yöntemlerinin kaynak kodları herkese açık bir şekilde paylaşılmıştır.

Tez Çalışmasından Türetilen Yayınlar:

Yapılan tez çalışması boyunca doktora tezinden türetilen yayınların listesi aşağıda verilmiştir:

Konferans yayınları:

- **Kuş, Z., Aydın, M., Kiraz, B., Can, B.** (2023). Neural Architecture Search Using Metaheuristics for Automated Cell Segmentation. In: Di Gaspero, L., Festa, P., Nakib, A., Pavone, M. (eds) Metaheuristics. MIC 2022. Lecture Notes in Computer Science, vol 13838. Springer, Cham. https://doi.org/10.1007/978-3-031-26504-4_12
- **Kuş, Z., & Kiraz, B.** (2023). BaDENAS: Bayesian Based Neural Architecture Search for Retinal Vessel Segmentation. In *2023 31th Signal Processing and Communications Applications Conference (SIU)*. IEEE.

Dergi yayınları:

- **Kuş, Z., Kiraz, B., Göksu, T. K., Aydın, M., Özkan, E., Vural, A., Kiraz, A., & Can, B.** (2023). Differential evolution-based neural architecture search for brain vessel segmentation. In *Engineering Science and Technology, an International Journal* (Vol. 46, p. 101502). Elsevier BV. <https://doi.org/10.1016/j.jestch.2023.101502>
- **Kuş, Z., Kiraz, B.** (2023). Evolutionary Architecture Optimization for Retinal Vessel Segmentation. *IEEE Journal of Biomedical and Health Informatics* (Minör Revizyon).
- **Kuş, Z., Kiraz, B.** (2023). Retina Damar Segmentasyonunda Sinir Mimarisi Arama için Farklı Kodlama Yöntemlerinin Karşılaştırılması. *International Journal of Advances in Engineering and Pure Sciences* (Hakem İncelemesi).

Tez Organizasyonu:

Birinci bölümde, görüntü segmentasyonu probleminden bahsedilmiş ve medikal görüntü segmentasyonu için yapılan çalışmalara bir giriş yapılmıştır. Aynı bölümde, medikal görüntü segmentasyonu için yaygın olarak kullanılan derin öğrenme ağları anlatılmış, hücre ve damar segmentasyonu problemleri tanımlanmış, bu alanda gerçekleştirilen çalışmalar anlatılmıştır. Sinir mimarisi arama probleminin tanımı yapılmış, tekrarı engellemek amacıyla çalışmada önerilen yöntemler tarafından ortak olarak kullanılan meta-sezgisel yöntemlerden olan DE ve ODE yöntemleri detaylı olarak anlatılmış ve yine tekrarı engellemek amacıyla çalışma boyunca kullanılan veri kümelerinden detaylıca bahsedilmiştir. İkinci bölümde, hücre ve beyin damar görüntülerinin segmentasyonu için kullanılan UNAS-Net yöntemi detaylıca anlatılmış, farklı çalışmalar ile yapılan kıyaslamalar ve elde edilen sonuçlar verilmiştir. Üçüncü bölümde, retinal kan damar görüntülerinin otomatik segmentasyonu için kullanılan MedUNAS yöntemi detaylı bir şekilde anlatılmış, farklı NAS çalışmaları ve son-teknoloji yöntemler ile yapılan detaylı karşılaştırmalar verilmiş ve elde edilen sonuçlar tartışılmıştır. Aynı bölümde, iki farklı kodlama stratejisine sahip olan UNAS-Net ve MedUNAS yöntemlerinin retinal kan damar segmentasyonu problemi üzerinde karşılaştırılması gerçekleştirilmiştir. Dördüncü bölümde ise BaDENAS'ın performansı retinal kan damar segmentasyonu veri kümesi üzerinde farklı NAS çalışmaları ile karşılaştırılmış, elde edilen sonuçlar yakınsama hızı, model karmaşıklığı ve segmentasyon performansı açısından detaylı bir şekilde tartışılmıştır. Sonuç bölümde, tez çalışmasında önerilen yöntemlerin genel bir analizi ve tartışması gerçekleştirilmiş, gelecek çalışmalar için bilgi verilmiştir.

BİRİNCİ BÖLÜM

1. HÜCRE VE DAMAR SEGMENTASYONU

1.1. GÖRÜNTÜ SEGMENTASYONU

Görüntü segmentasyonu (Image Segmentation), bilgisayarlı gözü araştırmalarında çalışılan klasik bir problemdir ve görüntü anlamlandırma, üzerinde yoğun bir şekilde çalışılan bir konudur. Görüntü segmentasyonu, bir görüntünün gri tonlama, renk, uzamsal doku ve geometrik şekiller gibi özelliklere göre birkaç ayrı bölgeye bölünmesini ifade eder. Elde edilen bu özellikler aynı alanlar arasında tutarlılık veya benzerlik gösterirken, farklı alanlar arasında net bir farklılık göstermektedir. Görüntü segmentasyonu, segmentasyonun farklı kaba ve hassas detay seviyesine göre anlamsal segmentasyon (semantic segmentation), örnek segmentasyon (instance segmentation) ve panoramik segmentasyon (panoramic segmentation) olarak üç ayrı gruba ayrılabilir. Tıbbi görüntülerin segmentasyonu, anlamsal segmentasyon görevi olarak kabul edilir. Günümüzde, uydu görüntülerinin, medikal görüntülerin veya otonom araçların çalışması boyunca elde edilen görüntülerin segmentasyonu gibi farklı çalışma alanları mevcuttur (Ess vd., 2009; Geiger vd., 2012). Önerilen derin sinir ağı yapılarındaki büyük artışla birlikte, daha doğru segmentasyon sonuçları elde etmek için görüntü segmentasyon yöntemleri adım adım geliştirilmektedir. Ancak farklı segmentasyon örnekleri için tüm görüntülere uygun evrensel bir segmentasyon algoritması yoktur.

Geleneksel görüntü segmentasyon yöntemleri, günümüzden yaygın olarak kullanılan derin öğrenmeye dayalı segmentasyon yöntemleriyle karşılaştırılmaz. Ancak, eşik (threshold) tabanlı segmentasyon (Xu vd., 2010), bölge tabanlı görüntü segmentasyonu (Cigla vd., 2008) ve kenar algılama tabanlı segmentasyon (Yu-Qian vd., 2006) gibi geleneksel görüntü segmentasyon yöntemlerinde önerilen fikirler hala öğrenilmeye değerdir (Ma vd., 2009; Ferreira vd., 2014; Ma vd., 2010). Geleneksel yöntemler, görüntüyü segmentlere ayırmak için sayısal görüntü işleme ve matematik bilgisini kullanır. Hesaplama yükü daha azdır ve segmentasyon hızı yüksektir, ancak

elde edilen ayrıntılar, sınırlar açısından segmentasyonun kesin doğruluğu garanti edilemez. Günümüzde, derin öğrenmeye dayalı yöntemler, görüntü segmentasyonu alanında kayda değer başarılar elde etmiştir. Segmentasyon doğruluğu bakımından, geleneksel segmentasyon yöntemlerine göre daha iyi sonuçlar elde edilmektedir. “Tam Evrişimli Sinir Ağı” (fully convolutional neural network), görüntünün anlamsal segmentasyonu için derin öğrenmeyi başarıyla gerçekleştiren ilk ağıdır. İlerleyen zamanlarda, hassas kenarların işlenmesinde güçlü bir avantaja sahip olan U-Net (Ronneberger vd., 2015), Mask R-CNN (He vd., 2017), RefineNet (Lin vd., 2017) ve DeconvNet (Noh vd., 2015) gibi iyi sonuçlar veren segmentasyon ağları önerilmiştir.

1.1.1. Derin Öğrenme Tabanlı Yöntemler ile Medikal Görüntülerin Segmentasyonu

Görüntü segmentasyonu işlemlerinde yoğun bir şekilde kullanılan “Evrişimli Sinir Ağları (ESA)”, başarılı özellik çıkarma ve daha iyi ifade yeteneğine sahiptir. Görüntü özelliklerinin elle çıkarılmasını veya görüntülerin aşırı ön işlenmesini (pre-processing) gerektirmez. Bu nedenle ESA, son yıllarda tıbbi görüntü segmentasyonunda sıklıkla kullanılmaktadır ve karar destek sistemi olarak büyük başarı elde etmiştir. Bu çalışmada derin öğrenme tabanlı tıbbi görüntü segmentasyon yöntemleri üç alt bölümde incelenmiştir: Evrişimli Sinir Ağları, Tam Evrişimli Sinir Ağı (Fully Convolutional Neural Network - TESA) ve U-Net.

1.1.1.1. Evrişimli Sinir Ağı (ESA)

Evrişimli sinir ağları (Convolutional Neural Network - ESA) (LeCun vd., 1989; LeCun vd., 1998), görüntü sınıflandırma, nesne konumlandırma (localization), nesne algılama, görüntü segmentasyonu gibi bilgisayarlı görü (computer vision) çalışmalarında çokça kullanılmakta olan bir derin öğrenme mimarisidir. ESA, bilgisayarlı görü çalışmaları için hazırlanan birçok hazır veri setinde başarılı ve rekabetçi sonuçlar vermektedir (LeCun vd., 1989; Goodfellow vd., 2016). ESA, kendisini oluşturan katmanların en az birinde, filtrelerin girdiler üzerinde dolaşması (konvolüsyon) işleminin uygulandığı ve sonucunda özellik çıkarımının gerçekleştirildiği derin sinir ağlarıdır (Şekil 1.1: Evrişimli sinir ağlarının yapısı (Kuş,

2019). ESA, anlatılan bu özelliği (konvolüsyon) ile diğer yapay sinir ağlarından ayrılır.

Şekil 1.1’de gösterildiği gibi evrişimli sinir ağları temel olarak 4 farklı yapıdan oluşmaktadır:

Girdi Katmanı: ESA’nın ilk katmanını oluşturur. Sinir ağının bir çıktı elde edebilmesi için gerekli olan görüntü girdi olarak verilir.

Evrişim İşlemi: Bir önceki katmandan gelen girdiler üzerinde seçilen filtreler (kernel, receptive field size) uygulanarak yeni özellik kümeleri oluşturulur. Seçilen filtreler (ağırlık matrisleri) ile verilen girdi üzerinde dolaşarak farklı özellikler elde edilmeye çalışılır. Girdiye uygulanan evrişim filtreleri sonucunda her biri farklı özelliği temsil etmeye çalışan özellik haritaları elde edilir.

Ortaklama İşlemi: Ortaklama (pooling) işlemi ya da bazı örneklerde alt örnekleme (subsampling) olarak geçen yöntem evrişimli sinir ağlarında girdi ve özelliklerin boyutunun düşürülmesi için kullanılır.

Tam Bağlantılı Katman: Evrişimli sinir ağlarında özellik çıkarımı işleminden sonra, elde edilen özelliklerin sınıflandırılması için kullanılan son katmanları içerir.

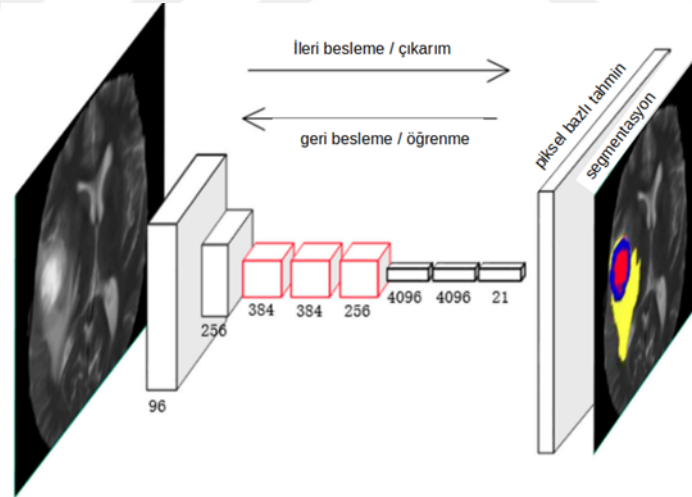


Şekil 1.1: Evrişimli sinir ağlarının yapısı (Kuş, 2019)

1.1.1.2. Tam Evrişimli Sinir Ağı

Tam evrişimli sinir ağı (TESA), anlamsal segmentasyon için kullanılan en başarılı ve gelişmiş sonuçları veren derin öğrenme tekniklerinin öncü çalışmasıdır (Şekil 1.2). VGG ve ResNet gibi genel olarak görüntü sınıflandırma problemi için kullanılan ESA ağlarında, ağın sonuna tamamen bağlı bazı katmanlar eklenir. Bu mimarilerde, sınıf veya sınıflara ait olasılık bilgisi softmax katmanından sonra elde

edilir ancak bu olasılık bilgisi tek boyutludur. Yani, her pikselin ait olduğu sınıf değil, yalnızca tüm görüntünün ait olduğu sınıf bilgisi tanımlanır. Bu nedenle kullanılan bu tam bağlantılı yöntem, görüntü segmentasyonu için uygun değildir. Long vd. (2015) bahsedilen problemlere yanıt olarak tam evrişimli sinir ağını önerdi. Klasik ESA yapısında, ilk beş katman evrişim katmanıdır. Altıncı ve yedinci katmanlar, 4096 (tek boyutlu vektör) uzunluğunda tamamen bağlı katmanlardır. Sekizinci katman, 1000 sınıfın her biri için olasılık değerlerinin tutulduğu 1000 uzunluğunda tam bağlantılı bir katmandır. TESA, her pikselin iki boyutlu bir özellik haritasını elde etmek için, katman 5'ten 7'ye doğru olan üç katmanı, evrişim filtre boyutları 7×7 , 1×1 ve 1×1 olan evrişim katmanları ile değiştirir. Bu değişiklik ile segmentasyon sorunu çözülmüştür. TESA, herhangi bir boyuttaki giriş görüntülerini kabul edebilir. TESA, son evrişim katmanının özellik haritasını “sık örneklemek” (up-sampling) ve onu giriş görüntüsü ile aynı boyuta geri döndürmek, yeniden inşa etmek için “ters evrişim” (deconvolution) katmanını kullanır. Böylece, orijinal girdi görüntüsündeki “uzamsal bilgi” (spatial information) korunurken her piksel için bir



Şekil 1.2: TESA genel yapısı (Liu vd., 2021)

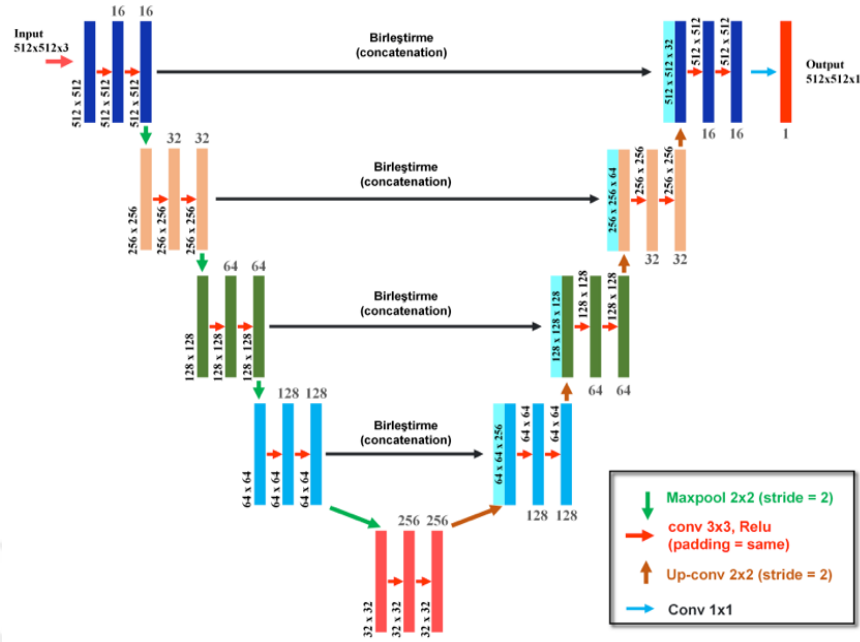
tahmin oluşturulabilir. Son olarak, nihai görüntü segmentasyonunu tamamlamak için örneklenen özellik haritası üzerinde piksel piksel sınıflandırma gerçekleştirilir.

1.1.1.3. U-Net

Ronneberger vd. (2015), TESA'yı temel alarak medikal görüntü segmentasyonunda yaygın olarak kullanılan U-Net ağını tasarladı. Etkileyici

performansı nedeniyle, U-Net ve çeşitleri, bilgisayarlı görünün çeşitli alt alanlarında yaygın olarak kullanılmaktadır. Bu yaklaşım 2015 MICCAI konferansında sunuldu ve 4000'den fazla atıf aldı. Şimdiye kadar, U-Net'in birçok çeşidi oluşturulmuştur. Evrişimli sinir ağlarının birçok yeni tasarım yöntemi vardır. Ancak birçoğu hala U-Net'in ana fikrini, yeni modüller ekleyerek veya diğer tasarım konseptlerini entegre ederek sunmaktadır.

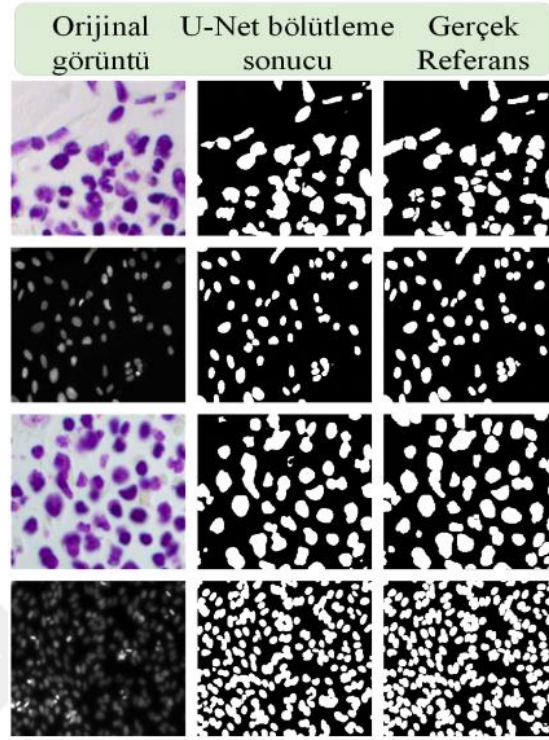
U-Net ağı özelleşmiş bir evrişimsel sinir ağı modelidir ve iki ana bölümden oluşur (Şekil 1.3). Bunlardan ilki kodlayıcı bölümdür, küçültme-daraltma yolu olarak adlandırılır. Küçültme yolu geleneksel evrişimsel sinir ağındaki evrişim adımını temsil eder, katmanlar boyunca girdinin boyutu azaltılır. Küçültme bölümünde veri boyutu, 2×2 ' lik filtre boyutuna sahip maksimum ortaklama işlemi (adım sayısı (stride) = 2) ile “aşağı örneklenir” (down-sampling). Aşağı örnekleme yapıldığı “kodlayıcı” (encoding) bölümünde her basamakta 3×3 ' lük filtreye sahip, aktivasyon fonksiyonu ReLU olan 2 adet evrişim katmanı bulunmaktadır (sıfır doldurma ile-zero padding). Giriş katmanında çıkartılan özellik haritası sayısı 16' dır ve bu sayı küçültme yolunda her adımda 2 katına çıkartılır. “Kod çözücü” (decoding) genişletme yolunun her adımında ise 2×2 ' lik sık örnekleme yapılır ve elde edilen özellik kanalları, aynı basamakta bulunan küçültme yolundaki özellik kanalları ile birleştirilir (concatenation). Birleştirme adımından sonra, genişletme yolundaki her basamakta 3×3 ' lük filtre boyutuna sahip, aktivasyon fonksiyonu ReLU olan 2 adet evrişim katmanı bulunmaktadır. Genişletme yolunda, her yukarı örneklemeden sonra, özellik haritalarının sayısı yarıya iner. Bu sebeple genişletme yolu ile küçültme yolu simetriktir. Son olarak, 16 adet özellik haritasından segmentasyon haritasını çıkarmak için 1×1 ' lik evrişim uygulanır. U-Net ağı yapısında, kodlama işlemi ile girdi olarak verilen görüntülerin daha az piksel ile, daha küçük boyutlarda bir temsili öğrenilmeye, girdiden özellikler çıkarılmaya çalışılır. Kod-çözme işleminde ise çıkarılan düşük boyutlu özellikler üzerinden orijinal girdi görüntüsü boyutunda bir görüntü elde edilmeye çalışılır, yani çıkarılan özellikler kullanılarak görüntü “tekrar inşa” (reconstruct) edilir.



Şekil 1.3: U-Net yapısı (Aydin, 2021)

1.2. HÜCRE SEGMENTASYONU

Hücre, tüm canlı organizmalarda temel yapısal, işlevsel ve biyolojik birimdir. Hücreleri ve hücre altı bölümleri görüntüleme, bilgi çıkarma ve inceleme yeteneği, çeşitli araştırma alanları için gereklidir (Chaffer vd., 2011). Yüksek çözünürlüklü floresan mikroskobundaki son gelişmeler, hücrelerin ve hücre altı yapıların ayrıntılı görselleştirilmesinin önünü açtı (Vonesch vd., 2006). Bilgi işlem tekniklerinin gelişimi ve görüntü segmentasyonu, sınıflandırması için bilgisayarlı görü, makine öğrenimi alanlarında yeni tekniklerin geliştirilmesi bu gelişmelere eşlik etmiştir (Deshmukh vd., 2014). Hücresel görüntülerin otomatik analizi, daha yüksek doğrulukta ve verimde hücre sayımı, segmentasyonu sağlar ve tekrarlanabilir bilgiler sunar. Bu tarz yüksek doğruluk ve verimlilik, araştırmacıların veya insan uzmanların öznel ve zaman alıcı yorumlarına güvenmek yerine farklı biyolojik sorunları ele almalarını sağlayabilir. Şekil 1.4'te hücre segmentasyonu işlemleri için yaygın olarak kullanılan ve başarılı sonuçlar elde edilen derin öğrenme sinir ağı U-Net için elde edilen hücre segmentasyonu sonuçları gösterilmiştir.



Şekil 1.4: Derin sinir ağı yöntemi ile hücre segmentasyonu işlemi (Aydin, (2021))

Literatürde yapılan çalışmalarda, 2 ve 3-boyutlu görüntülerdeki hücreleri segmentlere ayırmak için farklı yaklaşımlar önerilmiştir (Deshmukh vd., 2014; Meijering, 2012). Bazı yaklaşımlar yalnızca tek kanallı görüntülere dayanır, sitoplazmayı segmentlere ayırmak yerine yalnızca hücre çekirdeklerinin segmentasyonunu gerçekleştirir. Örneğin, temas eden ve örtüşen (üst üste gelen) çekirdekleri ayırmak için “havza tabanlı” (watershed-based) görüntü segmentasyonu (Wang vd., 2008; Sharif vd., 2012) ve “seviye kümesi” (levelset) yöntemleri (Nath vd., 2006; Dzyubachyk vd., 2008) kullanılmıştır. “Aktif kontur” (contour) modelleri ve Snake’in algoritması da segmentasyon çalışmalarında kullanılan geleneksel görüntü işleme teknikleri arasındadır (Wang vd., 2007).

Geleneksel görüntü işleme teknikleri yanında, “parlak alan” (bright field), “faz kontrast” görüntülerinde (phase contrast) piksel tabanlı sınıflandırma ve hücrelerin segmentasyonu için makine öğrenmesi yaklaşımları kullanılmıştır (Yin vd., 2010; Su vd., 2013). Song vd. (2014) tarafından gerçekleştirilen çalışmada ise hücre görüntülerinin segmentasyonu için evrişimli sinir ağı temelli bir derin öğrenme yaklaşımı önerildi. Ancak bu çalışmada kümelenmiş olan hücreler net bir şekilde ayrıştırılamamıştır. Delgado-Ortet vd. (2020) tarafından kırmızı kan hücrelerinin

segmentasyonu için başka bir derin öğrenme mimarisi önerilmiştir. Yapılan çalışmada, kan hücrelerinin sınıflandırılması için “Segmentasyon Sinir Ağı” olarak isimlendirdikleri bir derin sinir ağını ve evrişimli sinir ağını beraber kullanmışlardır.

1.3. DAMAR SEGMENTASYONU

Kan damarları, canlı organlar için hayati önem taşıyan oksijen ve besinleri taşıdıkları için büyük önem taşır. Kan damarlarındaki herhangi bir sorun, organların yetersiz beslenmesine neden olabilir ve bu nedenle hastaların yaşam kalitesini etkileyebilir (Atli vd., 2021). Genişlik, uzunluk, dallanma açıları, kıvrımlılık ve çatallanmalar gibi kan damarı özelliklerinin analizi, oftalmoloji, beyin cerrahisi, onkoloji, kardiyoloji ve larengoloji (Fraz vd., 2012; Moccia vd., 2018; Sreejini vd., 2019) dahil olmak üzere birçok alanda (Mekheimer vd., 2015; Eladawi vd., 2018) sorunların erken saptanmasına katkıda bulunur. Tanı göstergesi olarak faydalı ve kullanışlıdır (Eladaw vd., 2017). Patoloji örneklerinin etkin bir şekilde izlenebilmesi için öncelikle tıbbi görüntüleme kan damar segmentasyonu yapılır (Akbar vd., 2019). Elle kan damar segmentasyonu mümkün olsa da, mesleki beceri gerektiren meşakkatli ve monoton bir iştir; ve dolayısıyla zaman, iş gücü ve gözlemcilerin raporlama değişkenliği açısından uygulanabilir bir prosedür değildir. Bu nedenle, hem klinisyenlere hem de hastalara daha iyi tedavi planlaması ve zaman tasarrufu sağlamak amacıyla, yarı otomatik veya tam otomatik sistemlerin kan damar segmentasyonu için kullanılması bir gereklilik haline gelmiştir. Aşağıdaki faktörler sebebiyle kan damar segmentasyonu işlemi zorlaşmaktadır:

- (1) Damar şekilleri basit bir örüntü, model izleme;
- (2) Diğer yapıların (lezyonlar ve optik disk) varlığı (Rigamonti vd., 2012) ve
- (3) çeşitli görüntüleme koşulları (düşük görüntü kontrastı, gürültü ve patolojik nedenler)

“El yapımı” (handcrafted) ve “gözetimli” (supervised) yöntemler dahil olmak üzere kan damarlarının segmentasyonu için çok miktarda çalışma önerilmiştir. El yapımı yöntemler, damar yapılarının doğal özelliklerine dayanırken, denetimli yöntemler etiketlenmiş gerçek referans değerlerine bel bağlar. Gözetimli yöntemlerle öğrenilen öznitelikler, analitik yöntemlerle tanımlanması zor olan (Rigamonti vd., 2012) daha ayırt edici özelliklerdir ve daha iyi segmentasyon performansı elde etme

eğilimindedir. Gözetimli yöntemler arasında olan derin sinir ağı mimarileri, Van Grinsven vd. (2016) tarafından yayınlanan ilk derin öğrenme temelli makale ile retina kan damarı segmentasyonunda kayda değer çaba gösterilmiştir, genel olarak kendi aralarında rekabet etmekte ve en gelişmiş (state-of-the-art) sonuçları elde etmektedir (Soomro vd., 2019).

Literatürde, kan damar segmentasyonu için önerilen farklı yöntemleri sınıflandıran, kategorize eden çalışmalar mevcuttur. Örneğin, bu yöntemler Moccia vd. (2018) tarafından dört ana kategoriye, Kirbas vd. (2004) tarafından altı ana kategoriye ve Fraz vd. (2012) tarafından yedi ana kategoriye ayrılmıştır. Tez çalışmasında, gözetimli öğrenme kategorisine giren derin sinir ağları kullanılacağı için ilgili kategorideki yöntemler ile yapılan çalışmalardan bahsedilmiştir. Gözetimli yöntemler arasında yer alan “Yapay Sinir Ağı” (YSA) modelleri, ESA’ların gelişiminden önce yaygın olarak kullanılmaktaydı. Örneğin, Li vd. (2015) tarafından fundus görüntülerinin retinal damar “eşleme yamalarını” (mapping patches) elde eden bir “eşleme işlevi” (mapping function) bulmak için bir YSA mimarisi önerilmiştir. Önerilen mimari, girdi ve çıktı arasındaki karmaşık ilişkiyi elde etmek için 5 katmanlı bir yapıdan oluşmaktadır. ESA’ların gelişmesi ile çok daha karmaşık eşlemeler elde edilmiştir. Wang vd. (2015) tarafından gerçekleştirilen çalışmada, retina kan damarı segmentasyonu için ESA ve “Rastsal Orman” (Random Forest - RF) sınıflandırıcılarını birleştiren bir algoritma önerilmiştir. ESA, hiyerarşik bir özellik çıkarıcı olarak kullanılır ve ardından RF, öğrenilen özelliklerden yararlanır. Son olarak, toplu (ensemble) RF’ler ile segmentasyon işlemi gerçekleştirilir. Khalaf vd. (2016), sınıf içi varyansı azaltmak için ESA çıktı katmanının 3 sınıflı bir problem için çalışacak şekilde değiştirilmesini önermiştir. Çalışmada tanımlanan sınıflar ise arka plan, küçük damarlar ve büyük damarlar olarak isimlendirilmiştir. Verilen görüntüde bu sınıflar için segmentasyon gerçekleştirilmiştir. Bu ağın doğrulaması yalnızca “DRIVE” (Damar Çıkarımı için Dijital Retina Görüntüleri - Digital Retinal Images for Vessel Extraction) veritabanında gerçekleştirilmiştir. Son yıllarda araştırmacılar, birden fazla ESA katmanını çeşitli şekillerde istifleyerek daha derin mimariler oluşturmakta ve giderek daha karmaşık ilişkiler elde etmektedir. ESA’ların bu tür konfigürasyonlarına literatürde “derin öğrenme” (DÖ) mimarileri denir. Liskowski vd. (2016), fundus görüntülerinde kan damarlarının segmentasyonu için

bir DÖ mimarisi önermektedir. Mimari, bir yamanın merkez pikselini sınıflandırır. Yapılandırılmış tahmin tanımıyla segmentasyon sürecini daha da geliştirmişlerdir. Bu tanımda, mimari, bir seferde yalnızca merkez pikseli tahmin etmek yerine, s boyutundaki “yama” (patch) içerisine ortalanmış bir “pencere” (window) içindeki tüm pikseller için tahminler üretir. Guo vd. (2018) retina damar segmentasyonu için pikselleri 64x64'lük bir yama boyutunda sınıflandırabilmek amacıyla 3 evrişim katmanından oluşan sığ bir (shallow) DÖ mimarisi önermiştir. Literatürdeki bazı ağlar, özellik temsillerini zenginleştirmek için katmanlar arasında hiyerarşik özellikler de taşır. Örneğin, Huang vd. (2017), önceki katmanlarda öğrenilen özellikleri diğer katmanlara ileri beslemeli bir şekilde taşıyan “DenseNet” isimindeki ESA mimarisini önermiştir. Feng vd. (2020) ise ara konumlarda bazı katman özelliklerini birleştiren çapraz bağlı bir evrişimsel sinir ağı (CcNet) mimarisi önermiştir. U-Net (Livne vd. (2019)) ve ondan türetilmiş olan mimariler (Zhang vd., 2018; Atli vd., 2019), ağın öğrenme yeteneğini artırmak için önceki katman özelliklerini sonraki katmanlarla birleştiren benzer bir yaklaşımı önermişlerdir.

1.4. SİNİR MİMARİSİ ARAMA

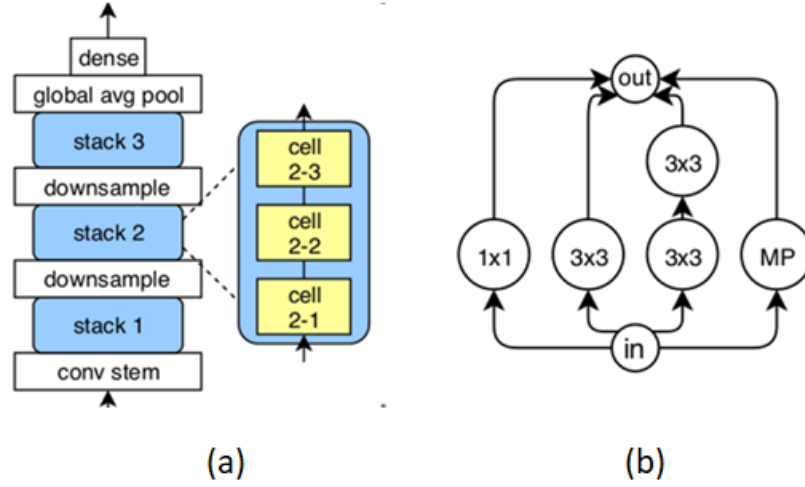
Otomatik Makine Öğrenmesi (AutoML) denince, Hiper-parametre Optimizasyonu (HPO) ve Sinir Mimarisi Arama (Neural Architecture Search - NAS) problemleri akla gelen en temel ve yaygın problemlerdendir (Hutter vd., 2019). Her makine öğrenmesi sisteminin ayarlanması gereken bileşenleri vardır ve bu bileşenlerin uygun şekilde belirlenmesi performansı doğrudan etkilemektedir. Ancak HPO'nun ve NAS'ın pratikte karşılaşılan bazı zorlukları vardır. En önemli problem, özellikle derin ağların çeşitli konfigürasyonlara bağlı fonksiyon hesaplamalarının maliyetli oluşudur. Bunun yanında konfigürasyon uzayının çok büyük oluşu, hatta bazı hiper-parametrelerin şartlı olarak diğer parametrelere bağlı oluşu ve parametrelerin değer aralıklarının belirlenmesi zorunluluğu problemi zorlaştırmaktadır. HPO için kullanılan en temel tekniklerden Izgara Arama ve Rastgele Arama teknikleri (Bergstra ve Bengio, 2012) elle ayarlamaya göre daha az zahmetli bir şekilde arama yapılmasını sağlar; ancak iki yöntemin de başarısı konfigürasyon uzayının boyutu arttıkça azalır.

Evrimsel Algoritmalar (EA) (Back, 1996) ile derin öğrenme ağlarında HPO gerçekleştiren birçok çalışma literatürde mevcuttur (Wang vd., 2018; Dufourq ve Bassett, 2017; Baldominos vd., 2018; Xie ve Yuille, 2017). Bu çalışmalar kullanılan veri kümeleri, tanımlanan arama uzayının büyüklüğü, üretilen ağın performansının test edilme yöntemi ya da çaprazlama ve mutasyon gibi EA operatörleri ve çözümlerin “gösterimi” (encoding) açısından farklılık göstermektedirler. Ayrıca bu çalışmaların büyük çoğunluğu ESA’lar üzerine yoğunlaşmaktadır. Bunun sebebi, hem bu ağların görüntü sınıflandırma/segmentasyon problemlerde bilinen en iyi yöntemler olması, hem de bu ağların ayarlanması gereken çok sayıda hiper-parametreye sahip olmalarıdır. Çalışmalarda bu hiper-parametrelerin bazıları sabitlenmiş, diğer hiper-parametrelerin değerlerinin seçimi üzerinde yoğunlaşmıştır; hangi parametrelerin optimizasyona dahil edilmesi konusunda ortak bir tutum henüz gelişmemiştir.

Otomatik makine öğrenmesi alt grubuna ait olan NAS çalışmaları temel olarak arama uzayı ve arama yöntemi olmak üzere iki parçadan oluşmaktadır. Yapılan çalışmalarda, özel bir problem için belirlenen arama uzayı kullanılarak farklı arama yöntemleri ile derin sinir ağı mimarileri otomatik olarak araştırılmaktadır. Temel amaç, seçilen değerlendirme metriği açısından iyi sonuçlar veren ve bunu daha az karmaşık yapılar ile gerçekleştirebilen ağların arama uzayı üzerinde verimli bir şekilde keşfedilebilmesidir. Bu arama sürecinin verimli bir şekilde gerçekleştirilebilmesi için arama uzayının doğru bir şekilde belirlenmesi ve arama yöntemlerinin seçimi büyük önem arz etmektedir.

Son zamanlarda literatürde sinir ağı mimarisi arama üzerine çeşitli yaklaşımlar geliştirilmiştir; pekiştirmeli öğrenme (Baker vd., 2016; Zoph vd., 2016), gradyan tabanlı (Grathwohl vd., 2018; Shin vd., 2018), evrimsel tabanlı (Liu vd., 2018; Baldeon vd., 2021; Liu vd., 2021). Genel olarak, tüm metotlar ağ yapısının küçük parçalara ayrılması; her parçanın kendi içinde “çizge” (graph) olarak gösterilip düğümler arasında hiçbir bağlantı kısıtı koymadan arama yapılması esasına dayanmaktadır. Bu yapılar, hücre tabanlı ağ yapıları olarak isimlendirilmektedir ve arama uzayının daraltılması için genellikle tekrarlayan hücre yapısı gösterimi (Ying vd., 2019) kullanılmaktadır. En çok bilinen ESA’ların aksine; hücre içinde operatörler arası tekrar eden bir örüntü kuralı olmadığı için çok farklı yapıda hücreler

ortaya çıkmaktadır. Tekrarlayan hücre yapısı Şekil 1.5'te gösterilmiştir (Ying vd., 2019). Bu şekilde, farklı yapıda ağların birleştirildiği Inception (Szegedy vd., 2015) benzeri ağların oluşturulmasına imkân sağlanmaktadır. Tüm ağ üst üste yığılmış hücre bloklarından oluşur; her hücre bloğu ise 3 adet hücreden oluşur. Sağ tarafta ise tek bir hücrenin içeriği gösterilmektedir. Her bir hücre, rastgele oluşturulmuş çizge (graph) benzeri bir yapıya sahiptir. Her bir düğüm üç operasyondan birini temsil edebilir (3x3 evrişim, 1x1 evrişim ve 3x3 en büyük ortaklama); kenarlar ise bu düğümlerin nasıl bağlandığını gösterirler. Tek koşul, bu çizgenin döngüsüz yönlü bir çizge olması ve her düğümün mutlaka girdi (in) ve çıktı (out) olarak isimlendirilen düğümlerle bağlı olmasıdır. Bunun yanında, arama uzayının büyüklüğünün sınırlanması için toplam düğüm ve kenar sayıları sınırlandırılmaktadır. Her bir hücre için ağ yapısı temsili için iki boyutlu komşuluk matris gösterimi, düğümlerdeki operasyonların temsili için ise vektör gösterimi kullanılmıştır.



Şekil 1.5: Tekrarlayan hücre yapısı gösterimi (Ying vd. (2019))

Literatürde tekrarlayan hücre yapısını örnek alan birçok çalışma bulunmaktadır. Zoph ve diğerleri (Zoph vd., 2018) tarafından NASNet adında bir NAS yaklaşımı önerilmiştir. Önerilen yaklaşımda, ağ yapısı hücreler adı verilen küçük bloklara bölünür ve daha sonra tüm ağ mimarisi (hücre tabanlı arama uzayı) yerine bu hücreler üzerinde arama yapılır. Her hücre, herhangi bir kenar veya düğüm sınırlaması olmaksızın bir çizge (graph) olarak temsil edilir (literatürde bazı sınırlamalar vardır (Ying vd., 2019)). Bu ağ yapısında, hücre içindeki işlemler arasında tekrar eden bir örüntü kuralı olmadığından farklı hücre yapıları üretilir. Liu

ve diğerleri (Liu vd., 2018), hücre tabanlı arama alanını kullanan ve görüntü sınıflandırma ve doğal dil işleme sorunları için başarılı sonuçlar sağlayan türevlenebilir mimari aramayı (DARTS) sunmuşlardır. Önerilen NAS yöntemlerinin adil bir karşılaştırması ve ağların eğitimi sırasında ortaya çıkan hesaplama süresini azaltmak için NAS-Bench-101 (Ying vd., 2019), NAS-Bench-201 (Dong vd., 2020) ve NAS-Bench-301 (Siems vd., 2020) gibi NAS karşılaştırma ölçütleri (benchmarks) yayınlanmıştır. Bunlar dışında, evrimsel NAS yaklaşımları farklı problemler için başarılı sonuçlar elde etmiştir (Real vd., 2019; Chen vd., 2022).

Yapılan farklı çalışmalar ve elde edilen sonuçlar incelendiğinde bu çalışmada evrimsel yöntemler kullanılarak NAS çalışması gerçekleştirilmiş ve daha başarılı sonuçlar vermesi sebebiyle hücre tabanlı arama uzayı kullanılmıştır. Yapılan çalışmada önerilen farklı NAS yöntemleri, bu yöntemler için kullanılan arama uzayları, kodlama stratejileri ve farklı arama yöntemleri Bölüm 2, 3 ve 4'te detaylı bir şekilde açıklanmıştır.

1.5. META-SEZGİSELLER

1.5.1. Diferansiyel Gelişim

Storn vd. (1997) tarafından önerilen DE, özellikle sürekli optimizasyon (continuous optimization) problemleri için iyi sonuçlar verdiği bilinen bir evrimsel algoritmadır. DE'de her aday çözüm, gerçek değerli (real valued) bir vektörle temsil edilir. Başlangıç popülasyonu rastgele oluşturulur ve değerlendirilir. Başlatma adımını takiben DE, yeni bir popülasyon oluşturmak için mutasyon, çaprazlama ve seçim operatörlerini kullanır.

Bu çalışmada, her bir aday çözüm ($X_{i,G}$, burada $i = 1, 2, \dots, NP$ 'e kadar, NP popülasyon büyüklüğü ve G nesil sayısıdır) $[0,1]$ aralığında gerçek değerli sayılar vektörüdür. Uygunluk fonksiyonu (fitness function) olarak dice skoru kullanılmıştır (Baldeon vd., 2021; Zhu vd., 2019) ve denklem 1.1'de gösterildiği gibi hesaplanır:

$Dice Skoru = \frac{2 y \cap \hat{y} }{ y + \hat{y} }$	(1.1)
--	--------------

Denklem 1.1'de, y gerçek referans (ground truth) görüntüsünü ve $\hat{y}$ segmente edilmiş görüntüyü temsil eder. Yapılan çalışmada, rastgele mutasyon ($DE/rand/1$) ve

ikili çaprazlama operatörleri (*DE/rand/1/bin*) kullanılmıştır. Mevcut popülasyondaki (hedef vektör) her bir çözüm için G jenerasyonundaki bir mutant vektör ($V_{i,G}$) şu şekilde oluşturulur:

	$DE/rand/1 : V_{i,G} = X_{r1,G} + F \times (X_{r2,G} - X_{r3,G})$	(1.2)
--	---	--------------

Denklem 2’de, $r1, r2, r3 \in \{1, 2 \dots NP\}$ ’den seçilen rastgele tam sayı değerlerini ve $F \in [0, 2]$ aralığından seçilen ölçek faktörünü (scaling factor) temsil etmektedir. Mutant vektörler üretildikten sonra, karşılık gelen “trial” vektörler ($U_{i,G}$), denklem 3’te gösterilen ikili çaprazlama kullanılarak oluşturulur:

	$U_{i,G}^k = \{V_{i,G}^k, \text{ if } rand() \leq CR; \text{ aksi durumda } X_{i,G}^k, \text{ for } k = [1, \dots, D]\}$	(1.3)
--	--	--------------

Denklem 3’te, CR çaprazlama olasılığını, $rand()$ uniform dağılımdan üretilen rastgele değeri ve D karar değişkenlerinin sayısını belirtir. Sınır ihlali (boundary constraint violation) durumunda, sınırları aşan değerler, uniform dağılımdan $[0,1]$ (Delacruz vd., 2020) örneklenen rastgele bir değerle değiştirilir.

Son olarak seçim işlemi uygulanır. Seçim aşamasında, her bir hedef vektör, amaç fonksiyonu değeri açısından karşılık gelen trial vektör ile karşılaştırılır. Hedef vektör, karşılık gelen trial vektörden daha iyiye, sonraki nesil için hedef vektör seçilir (Qin vd., 2009; Mallipeddi vd., 2011). Bu çalışmada ele alınan problem bir maksimizasyon problemi olduğundan, gelecek nesil için seçim işlemi denklem 4’te gösterildiği gibi gerçekleştirilmiştir:

	$X_{i,G+1} = \{U_{i,G}, \text{ if } f(U_{i,G}) > f(X_{i,G}); \text{ aksi durumda } X_{i,G}, \text{ for } i = [1, \dots, NP]\}$	(1.4)
--	--	--------------

Denklem 1.4’te $f(.)$ karşılık gelen vektörün uygunluk değerini (fitness value) gösterir Algoritma 1.1’de diferansiyel gelişim algoritması sözde kodu gösterilmiştir.

Algoritma 1.1: Diferansiyel Gelişim Algoritması sözde kodu

```
1  $P \leftarrow$  başlangıç popülasyonunu oluştur
2 while Sonlandırma kriteri gerçekleşene kadar devam et do
3   for  $P$  içerisindeki tüm modeller,  $model$  do
4      $hedef = model.mimari$ 
5      $mutant = \text{Mutasyon}(hedef)$ 
6      $mutant = \text{Çaprazlama}(hedef, mutant, CR)$ 
7      $mutant.uygunluk \leftarrow \text{Değerlendir}(mutant)$ 
8     if  $mutant.uygunluk < hedef.uygunluk$  then
9        $P_{sonraki.ekle}(mutant)$  // Bir sonraki jenerasyona bu
        çözümünü ekle
10    else
11       $P_{sonraki.ekle}(hedef)$  // Bir sonraki jenerasyona bu
        çözümünü ekle
12   $P = P_{sonraki};$ 
```

Yapılan çalışmada, DE için seçilen parametreler şu şekildedir: Popülasyon boyutu: 20; Mutasyon çarpanı = 0,5; Çaprazlama olasılığı = 0,5; Mutasyon yöntemi = rand1; Çaprazlama yöntemi = İkili; Tamir yöntemi = Rastgele (de-la-Cruz-Martínez vd., 2020).

1.5.2. Karşıtlık Temelli Diferansiyel Gelişim

ODE (Rahnamayan vd., 2008) yönteminde, popülasyon başlatma ve oluşturma adımlarında karşıtlığa dayalı öğrenme (Opposition Based Learning - OBL) kullanılır. Bu çalışmada, ODE için DE'nin (*DE/rand/1/bin*) klasik varyantı kullanılmıştır.

Karşıtlık temelli başlatma, dört adımdan oluşur. (1) Başlangıç popülasyonu (P) rastgele oluşturulur. (2) Karşıt popülasyonu (OP) oluşturmak için başlangıç popülasyonundaki her çözümün karşıtı hesaplanır. Başlangıç popülasyonundaki bir çözümün karşıtı denklem 1.5'te gösterildiği gibi hesaplanır:

$OP_{i,k} = a_k + b_k - P_{i,k}$ $\text{for } i: \{1, \dots, NP\}; k: \{1, \dots, D\}$	(1.5)
--	--------------

Denklem 2.5'te, $P_{i,k}$ ve $OP_{i,k}$ sırasıyla mevcut popülasyon ve karşıt popülasyondaki i . bireyin (individual) k . karar değişkenini; a_k ve b_k , k . karar değişkeninin alt ve üst sınırlarını; D karar değişkenlerinin sayısını ve NP popülasyon büyüklüğünü temsil etmektedir.

(3) Başlangıç ve karşıt popülasyonlar birleştirilir. (4) En iyi NP çözüm, birleştirilmiş popülasyondan seçilir ($P \leftarrow select(P \cup OP)$).

Öte yandan, karşıt temelli nesil atlama adımı (opposition based generation jumping), atlama olasılığına ($JR: 0.3$ olarak seçilmiştir) bağlı olarak gerçekleştirilir. Atlama adımında, mevcut popülasyonun karşıt popülasyonu hesaplanır. Sabit değerler yerine, çözümdeki her bir karar değişkeninin karşıt değeri denklem 1.6'da gösterildiği gibi dinamik olarak hesaplanır:

$OP_{i,k} = Min_k^G + Max_k^G - P_{i,k}$ $for\ i: \{1, \dots, NP\}; k: \{1, \dots, D\}$	(1.6)
---	--------------

Denklem 1.6'da, Min_k^G ve Max_k^G sırasıyla k . karar değişkeninin G jenerasyonu P popülasyonunda aldığı minimum ve maksimum değerleri belirtir. Son olarak, mevcut popülasyon ve karşıt popülasyon birleştirilir ve birleştirilmiş popülasyondan NP en iyi çözüm seçilir ($P \leftarrow select(P \cup OP)$). Karşıtlık temelli diferansiyel gelişim algoritmasının sözde kodu Algoritma 1.2'de gösterilmiştir.

Algoritma 1.2: Karşıtlık Temelli Diferansiyel Gelişim Algoritması sözde kodu. P_0 : Başlangıç popülasyonu, OP_0 : Başlangıç popülasyonunun karşıtı, N_p : Popülasyon boyutu, P : Mevcut popülasyon, OP : Mevcut popülasyonunun karşıtı, $[a_j, b_j]$: j . değişkenin değer aralığı, MAX_j^P : j . değişkenin mevcut popülasyondaki minimum değeri, MIN_j^P : j . değişkenin mevcut popülasyondaki maksimum değeri gösterir.

```

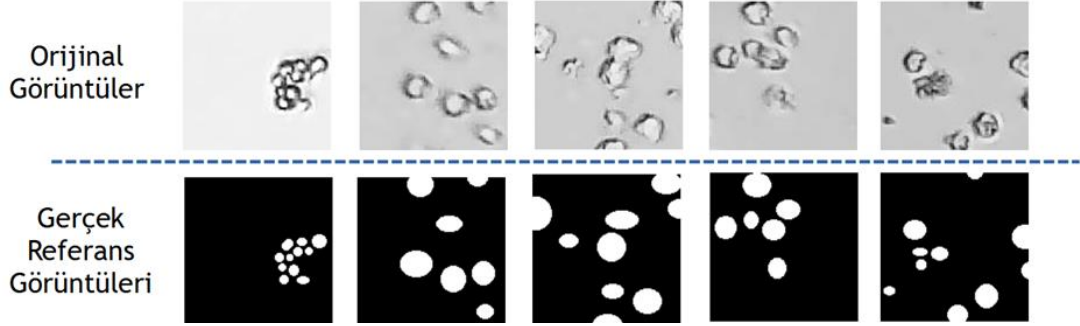
1 /* Karşıtlık Temelli Popülasyon Başlatma */
2  $P_0 \leftarrow$  başlangıç popülasyonunu oluştur
3 for  $i = 0; i < N_p; i++$  do
4     for  $j = 0; j < D; j++$  do
5          $OP_{0,i,j} = a_j + b_j - P_{0,i,j}$ 
6  $P \leftarrow \{P_0, OP_0\}$  kümelerinden  $N_p$  adet en iyi çözümü seç
7 /* Karşıtlık Temelli Popülasyon Başlatma */
8 while Sonlandırma kriteri gerçekleşene kadar devam et do
9     for  $P$  içerisindeki tüm modeller, model do
10         hedef = model.mimari
11         mutant = Mutasyon(hedef)
12         mutant = Caprazlama(hedef, mutant, CR)
13         mutant.uygunluk  $\leftarrow$  Degerlendir(mutant)
14         if mutant.uygunluk < hedef.uygunluk then
15              $P_{sonraki.ekle}(\text{mutant})$  // Bir sonraki jenerasyona bu
16                 çözümü ekle
17         else
18              $P_{sonraki.ekle}(\text{hedef})$  // Bir sonraki jenerasyona bu
19                 çözümü ekle
20      $P \leftarrow P_{sonraki}$ ;
21 /* Karşıtlık Temelli Jenerasyon Atlaması */
22 if  $\text{rand}(0, 1) < J_r$  then
23     for  $i = 0; i < N_p; i++$  do
24         for  $j = 0; j < D; j++$  do
25              $OP_{i,j} = MIN_j^P + MAX_j^P - P_{i,j}$ 
26      $P \leftarrow \{P, OP\}$  kümelerinden  $N_p$  adet en iyi çözümü seç
27 /* Karşıtlık Temelli Jenerasyon Atlaması */

```

1.6. VERİ KÜMELERİ

1.6.1. Optofil Veri Kümesi

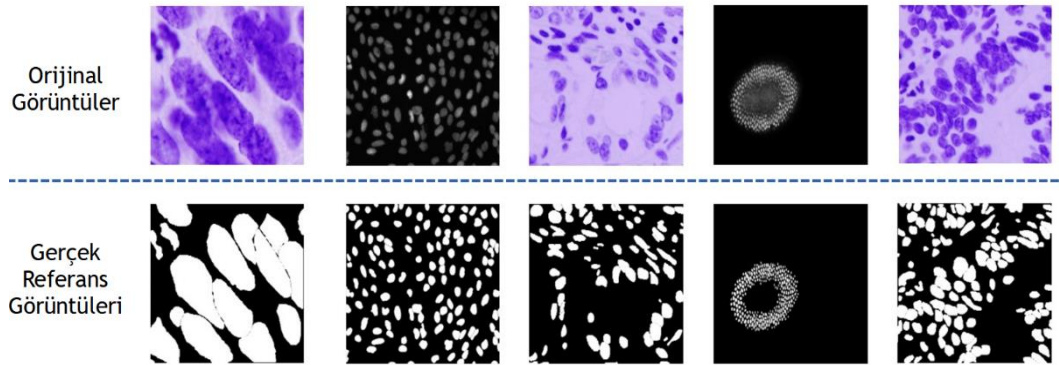
Bu veri kümesi, parlak alan otomatik hücre sayma cihazı (Quantacell, Optofil) kullanılarak elde edilen A2780 hücre kültürü görüntülerinden oluşur. Veri seti 128×128 boyutunda 4914 adet gri tonlamalı görüntü içerir. Veri seti üç parçaya ayrılmıştır: eğitim, test ve doğrulama. Bu parçalar sırasıyla: 3539, 982 ve 393 adet görüntüyü içermektedir. Gerçek referans görüntüler (segmente edilmiş görüntüler – ground truth) FiJi yazılımı (Schindelin vd., 2012) aracılığıyla oluşturulmuştur. A2780 hücreleri arasından seçilen beş örnek görüntü için orijinal ve gerçek referans görüntü çiftleri Şekil 1.6'da gösterilmektedir.



Şekil 1.6: Optofil veri kümesi için orijinal ve gerçek referans görüntüleri

1.6.2. Hücre Çekirdeği Veri Kümesi

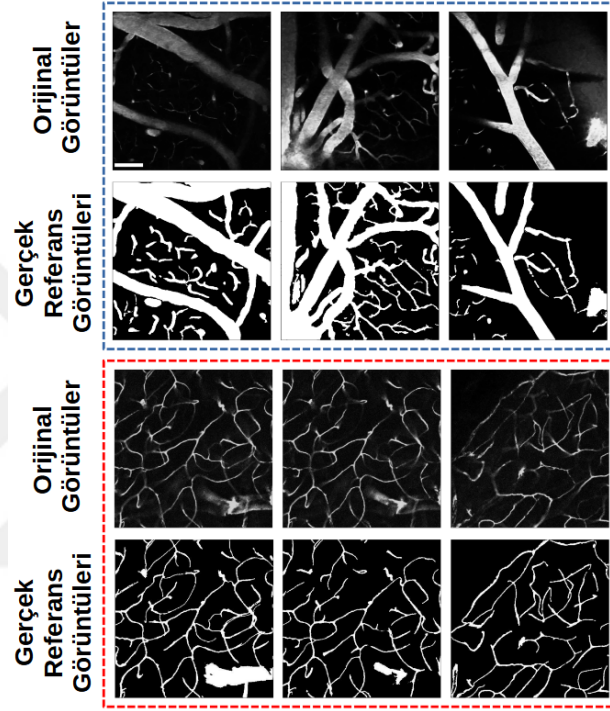
Hücre çekirdeği veri seti, segmente edilmiş çekirdek görüntülerini içermektedir. Görüntüler çeşitli koşullar altında elde edilmiştir. Araştırmacılara açık olarak paylaşılan bu veri kümesinde 670 adet renkli görüntü bulunmaktadır. Ayrıca, her bir renkli görüntüye karşılık gelen segmente edilmiş görüntüler bulunmaktadır ve bu görüntüler gerçek referans görüntüler olarak adlandırılır. Her görüntü farklı boyutta olduğundan dolayı ‘ön işleme’ (preprocessing) adımı uygulanması gerekmiştir. Bu nedenle, (Aydın vd, 2022) tarafından gerçekleştirilen çalışmada paylaşılan ve ilgili veri kümesi için önceden işlenmiş görüntüleri içeren veri kümesi kullanılmıştır. Bu veri kümesi de üç parçaya ayrılmıştır: eğitim, test ve doğrulama. Bu parçalar sırasıyla: 483, 134 ve 53 adet görüntüyü içermektedir. Şekil 1.7’de, ilgili veri seti için hem orijinal hem de gerçek referans görüntülerini içeren bir dizi görüntü çifti gösterilmiştir.



Şekil 1.7: Hücre çekirdeği veri kümesi için orijinal ve gerçek referans görüntüleri

1.6.3. KUVESG ve vesselNN Veri Kümeleri

vesseINN (Teikari vd., 2016), iki foton mikroskobu ile elde edilen açık kaynaklı bir hacimsel serebrovasküler veri kümesidir. Bu veri kümesi, fare korteksi ve insan skuamöz hücreli karsinom tümörlerinin görüntülerini içeren toplam 12 adet hacimsel yığından (volumetric stack) oluşur. Yapılan tez çalışmasında ise *vesseINN*



Şekil 1.8: vesseINN (mavi kesikli çizgi) ve KUVESG veri kümeleri için (kırmızı kesikli çizgi) orijinal ve gerçek referans görüntü çiftleri. Ölçek çubuğu 100 µm'yi gösterir

içerisindeki 12 yığından sadece 9 tanesi kullanılarak 512×512 boyutunda toplam 179 görüntü elde edildi ve bu çalışmada kullanıldı. İkinci veri kümesi olan *KUVESG* (Özkan vd., 2022), çoklu foton mikroskobundan alınan fare korteks görüntülerinden oluşur. Bu veri kümesi, 1024×1024 boyutunda 39 adet görüntüden oluşur. *KUVESG* veri kümesindeki her görüntü 4 eşit parçaya bölünerek 512×512 boyutunda 156 adet görüntü elde edilir. Bu iki veri kümesinin birleştirilmesi ile toplam 335 adet tek kanallı 8 bitlik gri tonlamalı görüntü elde edilmiştir. Bu görüntülerden 301 adedi modelin eğitiminde, 30 adedi ise test için kullanılmıştır. Şekil 1.8, vesseINN ve KUVESG için seçilen görüntü çiftlerini göstermektedir.

1.6.4. DRIVE

DRIVE veri kümesi (Staal vd., 2004) Hollanda'nın diyabetik retinopati tarama programından görüntüler içerir ve 20'si eğitim ve 20'si test için kullanılan etiketlere sahip 40 retina görüntüsünden oluşur. Görüntüler 565×584 piksel çözünürlüğe sahiptir ve üç kanallı renkli görüntülerdir. Yapılan çalışmada gerçekleştirilen tüm deneyler için eğitim ve test setlerinin orijinal bölümleri kullanılmış ve ilk gözlemcinin işaretlediği etiketler gerçek referans görüntüsü olarak seçilmiştir (Wentao vd., 2022; Liv d., 2023). DRIVE veri kümesi için orijinal ve bu görüntülere karşılık gelen gerçek referans görüntüleri Şekil 1.9'da gösterilmiştir.

1.6.5. CHASE_DB1

CHASE_DB1 veri kümesi (Carballal vd., 2018) 999×960 piksel çözünürlüğe sahip 28 adet renkli retina görüntüsünden oluşan bir veri kümesidir. İki bağımsız uzman bu görüntüleri manuel olarak işaretlemiş ve bu çalışmada ilk uzmanın işaretlemeleri gerçek referans görüntüsü olarak kullanılmıştır. Modellerin eğitilmesi için ilk 20 görüntü kullanılırken, son sekiz görüntü değerlendirme için kullanılmıştır (Wentao vd., 2022; Liv vd., 2023). CHASE_DB1 veri kümesi için orijinal ve bu görüntülere karşılık gelen gerçek referans görüntüleri Şekil 1.9'da gösterilmiştir.

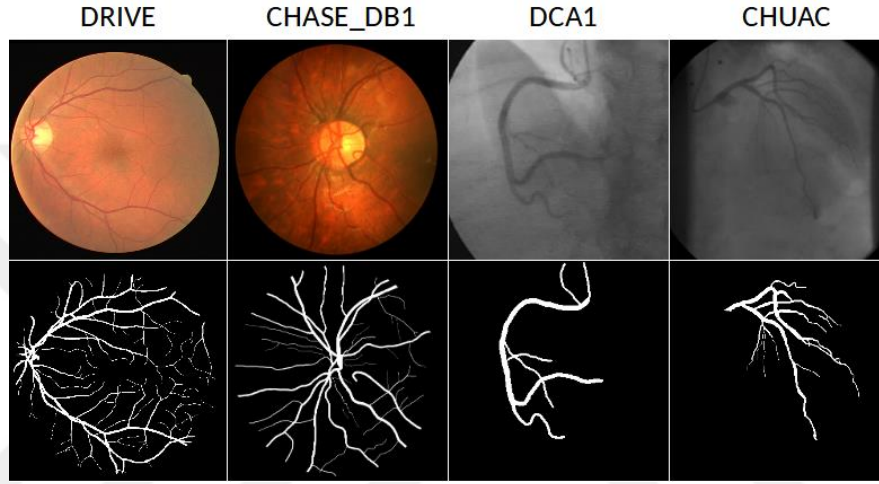
1.6.6. DCA1

DCA1 veri kümesi (Sanchez vd., 2019), uzman bir kardiyoloğun işaretlediği gerçek referans görüntüleri ile 134 adet X-ışını koroner anjiyogramı görüntüsünü içerir. Her görüntü, PGM formatında kodlanmış 300×300 piksel çözünürlüğe sahip gri ölçekli bir görüntü olarak temsil edilir. Bu çalışmada, DCA1 veri kümesi eğitim kümesi olarak kullanılan 100 adet görüntü ve 34 adet test görüntüsünden oluşan iki farklı alt kümeye ayrılmıştır. Eğitim ve değerlendirme adımlarında sırasıyla bu alt kümeler kullanılmıştır. DCA1 veri kümesi için orijinal ve bu görüntülere karşılık gelen gerçek referans görüntüleri Şekil 1.9'da gösterilmiştir.

1.6.7. CHUAC

CHUAC veri kümesi (Carballal vd., 2018), CHUAC Hemodinamik Biriminden temin edilen, her biri 189×189 piksel çözünürlüğe sahip 30 koroner

anjiyografi görüntüsünden oluşmaktadır. Doktorlar tarafından görüntü düzenleme programı kullanarak 512×512 piksel boyutunda gerçek referans görüntülerini manuel olarak oluşturmuşlardır. Bu çalışmada, diğer çalışmalarda olduğu gibi ilk 20 görüntü eğitim için kullanılırken, son 10 görüntü test için kullanılmaktadır (Samuel vd., 2021). CHUAC veri kümesi için orijinal ve bu görüntülere karşılık gelen gerçek referans görüntüleri Şekil 1.9’da gösterilmiştir.



Şekil 1.9: Dört farklı retinal kan damarı veri kümesi için örnek görüntüler. Üst satır orijinal görüntüleri ve alt satır bu görüntülere karşılık gelen gerçek referans görüntülerini temsil eder.

İKİNCİ BÖLÜM

2. UNAS-NET

2.1. HÜCRE SEGMENTASYONU İÇİN UNAS-NET

2.1.1. Giriş

Hücre, tüm canlı organizmalarda temel, yapısal, işlevsel ve biyolojik birimdir. Hücreleri görüntüleme ve onlardan bilgi toplama, “görüntü sitometrisi”, “akış sitometrisi” ve “hücre sıralayıcılar” gibi çeşitli bilimsel alanlar için önemlidir (Chaffer vd., 2011). Yüksek çözünürlüklü floresan mikroskopundaki son gelişmeler sayesinde hücrelerin ve hücre altı parçalarının daha hassas görüntülenmesi mümkün hale geldi (Vonesch vd., 2006). Bilgisayarların artan hesaplama kabiliyeti ve derin öğrenme alanındaki gelişmeler tıbbi görüntüleme alanındaki çalışmaları hızlandırmıştır (Deshmukh & Mankar, 2014). Hücresel görüntülerin otomatik analizi, daha yüksek doğruluk ve verimlilikle hücre sayımı ve segmentasyon sağlar ve tekrarlanabilir bilgiler sunar. Öte yandan, uzman insanlar tarafından gerçekleştirilen el ile segmentasyon, daha fazla zaman alıcıdır ve öznel yorumlar nedeniyle daha fazla hataya açıktır.

Yapılan önceki çalışmalarda, 2B görüntülerde hücrelerin segmentasyonu için farklı görüntü işleme teknikleri önerilmiştir (Deshmukh & Mankar, 2014; Meijering, 2012). Örneğin, üst üste binen hücreleri ayırmak için “havza tabanlı görüntü segmentasyonu” (watershed-based)(Sharif vd., 2012; Wong vd., 2008) ve “seviye kümesi” (level-set) yöntemleri kullanılmıştır. “Aktif kontur modelleri” ve “Snake algoritması” (Wang vd., 2007) da segmentasyon çalışmalarında kullanılan geleneksel görüntü işleme teknikleri arasındadır.

Geleneksel görüntü işleme tekniklerine ek olarak, “parlak alan” (bright field) ve “faz kontrastlı” (phase contrast) görüntülerde hücrelerin piksel tabanlı sınıflandırılması ve segmentasyonu için makine öğrenmesi yaklaşımları kullanılmaktadır (Su vd., 2014; Yin vd., 2010). Son zamanlarda derin öğrenme

ağları, biyomedikal görüntülerin segmentasyonu ve sınıflandırılması gibi farklı problemlerde sıklıkla kullanılmaktadır (Mookiah vd., 2021; Habibzadeh vd., 2018; Hollandi vd., 2020). Song vd., (2014) tarafından önerilen çalışmada, hücre görüntülerinin segmentasyonu için “evrişimli sinir ağı” (ESA) tabanlı bir derin öğrenme yaklaşımı önerilmiştir. Ancak, kümelenmiş hücreler bu çalışmada net bir şekilde ayrıştırılamamıştır. Delgado-Ortet vd., (2020) ise kırmızı kan hücresi görüntülerinin segmentasyonu için kullanılan evrişimli sinir ağına dayalı başka bir yöntem önermektedir. Bu çalışmada, kan hücrelerinin sınıflandırılması için "Segmentasyon Sinir Ağı" adını verdikleri derin bir sinir ağı ve evrişimli sinir ağını birlikte kullanmışlardır. 2021 yılında Wang vd. tarafından, hücre çekirdeklerinin ve karaciğer görüntülerinin segmentasyonu için BiX-NAS olarak isimlendirilen bir çalışma önerilmiştir. Klasik U-Net (Ronneberger vd., 2015) yapısında birden fazla kodlayıcı ve kod çözücü bloktan oluşan yapı bir defa çalıştırılır ve segmente edilmiş görüntü elde edilir. BiX-NAS, diğer çalışmalardan ve klasik U-Net yapısından farklı olarak iki adımlı (iteration) ‘kodlayıcı’ (encoding) ve ‘kod çözücü’ (decoding) blok yapısı önermektedir. Aynı zamanda, klasik U-Net yapısında her bir kodlayıcı blok, elde etmiş olduğu çıktıyı kendi seviyesindeki kod çözücü bloğa iletir. BiX-NAS çalışmasında ise birbirine karşılık gelen kodlayıcı ve kod çözücü bloklar birleştirilmek zorunda değildir. Farklı seviyelerdeki kodlayıcı ve kod çözücü blokların birbiri ile birleştirilmesine izin verilir ve bu birleşim işlemlerini BiX-NAS yapısı belirler. Yapılan bu çalışmalar incelendiğinde, biyomedikal görüntülerin segmentasyonunda genel olarak üstün performansı nedeniyle U-Net modelinin diğer ESA mimarilerine göre daha fazla tercih edildiği görülmektedir.

U-Net (Ronneberger vd., 2015), yukarıdaki paragrafta bahsedildiği gibi medikal görüntülerin segmentasyonu için başarılı sonuçlar veren özelleşmiş bir derin öğrenme mimarisidir. Bu yapı, U-şeklinde dizilmiş kodlayıcı ve kod çözücü bloklardan oluşmaktadır. Kodlayıcı bloklar, girdi olarak verilen görüntülerden öğrenilebilen ağırlıkları kullanarak özellikler çıkarır. Çıkarılan bu özellikler, kod çözücü katmanlarda segmente edilmiş görüntünün oluşturulması için kullanılır. Ancak, U-Net modeli elle oluşturulmuş bir ağ yapısıdır ve optimize edilmesi gereken birçok hiper parametresi vardır. Pek çok optimizasyon tekniği hem segmentasyon performansı hem de hesaplama karmaşıklığı açısından verimli olması gereken ağ

mimarilerini bulmaya odaklanmıştır. “Sinir Ağı Mimarisi Arama” (NAS) olarak adlandırılan bu teknikler, tümör segmentasyonu ve damar segmentasyonu gibi çeşitli biyomedikal görüntüleme problemlerine uygulanmıştır.

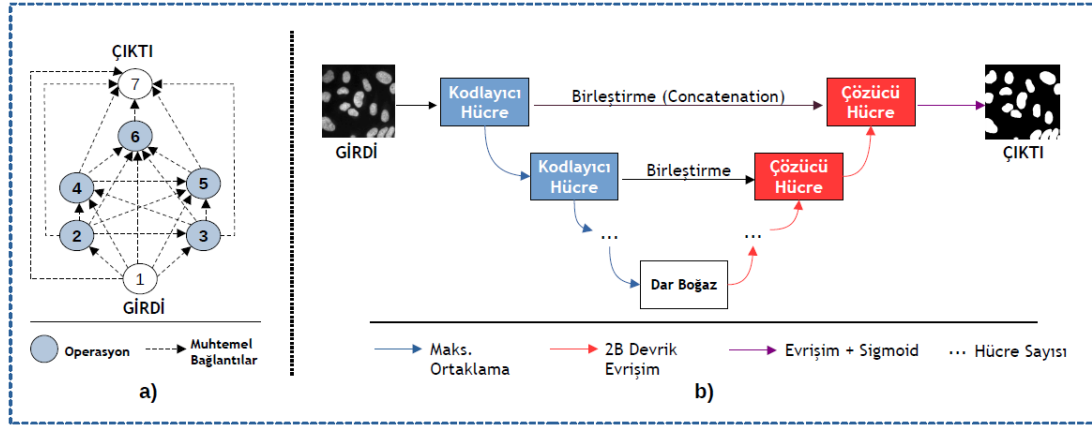
2.1.2. Yöntem

Son yıllarda, NAS için çeşitli yaklaşımlar geliştirilmiştir ve bu çalışmaların çoğu çizgelere (graph) dayanmaktadır. Çizgeler, hücre adı verilen her bir yapıyı temsil etmek için kullanılır. Ağ mimarileri, birden çok hücrenin birleştirilmesiyle oluşturulur. Her hücre, arama uzayını (Ying vd., 2019) azaltmak için kullanılabilen, genellikle tekrarlanan hücre olarak adlandırılan farklı bir çizge yapısı veya aynı çizge yapısı içerebilir. “Son teknoloji” (state-of-the-art) ESA’lardan farklı olarak, hücre içindeki işlemler arasında tekrarlanan örüntü kuralı olmadığından dolayı çok farklı yapılara sahip hücreler oluşturulur.

UNAS-Net (Kuş vd, 2023a) iki ana bölümden oluşur: meta-sezgisel yöntemler ve arama uzayı. Meta-sezgisel yöntemler olarak Diferansiyel Gelişim (DE), Karşıtlık Temelli Diferansiyel Gelişim (ODE) ve Yerel Arama (LS) yöntemleri seçilmiştir. İlerleyen bölümlerde, kullanılan meta-sezgisel yöntemler ve arama uzayı konfigürasyonunun detayları anlatılacaktır.

2.1.2.1. Arama Uzayı

U-Net yapısında olduğu gibi, önerilen UNAS-Net yapısı da ardışık olarak yerleştirilen “kodlayıcı” (encoder) ve “kod çözücü” (decoder) hücrelerden oluşur. Şekil 2.1’de UNAS-Net’in genel yapısı gösterilmektedir. Önerilen yapıda, bu hücrelerin tamamı kodlayıcı ve kod çözücü hücreler olarak adlandırılmıştır. Klasik U-Net yapısında olduğu gibi kodlayıcı hücreler, verilen görüntü üzerinden yeni özelliklerin elde edilmesi, çıkarılması için kullanılır. Her hücre, V (vertice - düğüm) adet düğüm ve E (edge - kenar) adet kenardan oluşan “döngü içermeyen yönlü çizgeler” (directed acyclic graph - DAG) ile temsil edilir. Hücreler içerisindeki her V düğümü, L etiketine sahiptir. Bu L etiketleri her bir düğümün kullandığı operasyonu göstermek için kullanılır. Hücreler için seçilebilecek olası operasyonlar (L kümesindeki elemanlar) Tablo 2.1’de gösterilmiştir.



Şekil 2.1: UNAS-Net'in genel yapısı. a) Hücresel yapı, b) UNAS-Net genel yapısı

Tablo 2.1'de gösterilen operasyonlar, DARTS makalesi referans alınarak belirlenmiştir (Liu vd., 2018). Hücre içerisindeki kenarlar ise bu düğümlerin birbirine nasıl bağlandığını gösterir. Her hücre en fazla 5 ara düğüme ve 9 kenara, bağlantıya sahip olabilir. Hücredeki ilk ve son düğümler sabittir ve sırasıyla *GİRİ* ve *ÇIKTI* olarak adlandırılır. Her düğüm, kendisinden sonra gelen düğümlere bağlanabilir. Bu sayede, klasik U-Net yapısından farklı olarak kodlayıcı hücreler içerisinde, verilen bir görüntüye tek tip bir operasyon uygulamak yerine bir dizi farklı operasyon uygulanır. Kodlayıcı hücreleri, “aşağı örnekleme” (subsampling - görüntü boyutlarını düşürmek için kullanılır) adı verilen “maksimum ortaklama” (max pooling) işlemi takip eder. Ardışık kodlayıcı hücreler ve aşağı örnekleme işlemlerinden sonra, “darboğaz katmanı” (bottleneck layer) kullanılır. Bir sonraki adımda, çıkarılan öznetelikler kullanılarak görüntüler kod çözücü hücreler kullanılarak yeniden oluşturulur. Kodlayıcı hücrenin tersi olan kod çözücü hücrede, görüntülerin “yukarı örnekleme” (upsampling - görüntü boyutlarını arttırmak için kullanılır) için “devrik evrişim” (transpose convolution) uygulanır. Son olarak, üretilen çıktıya “sigmoid aktivasyon fonksiyonu” (sigmoid activation function) uygulanarak ikili değerlerden oluşan bir görüntü elde edilir. Her kodlayıcı ve kod çözücü hücrede aynı tip hücre yapısı kullanılır.

Hücrelerdeki V ve E sayıları arttıkça arama uzayı katlanarak büyür. Bu nedenle, toplam düğüm ve kenar sayısı sınırlandırılmıştır. Bu sınırlamalar belirlenirken (Ying vd., 2019) çalışması referans alınmıştır:

- Tablo 2.1’de gösterilen 11 adet operasyon, hücreler içerisinde seçilebilir ($|L| = 11$)
- Maksimum kenar sayısı 9 olarak belirlenmiştir ($E \leq 9$)
- Maksimum düğüm sayısı 7 olarak belirlenmiştir ($V \leq 7$)

Tablo 2.1: Hücreler için seçilebilecek olası operasyonlar.

Operasyon Adı
1×1 2B Evrişim (1 × 1 conv)
3×3 2B Evrişim (3 × 3 conv)
5×5 2B Evrişim (5 × 5 conv)
7×7 2B Evrişim (7 × 7 conv)
3 × 3 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim (3×3 depconv)
5 × 5 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim (5×5 depconv)
7×7 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim (7×7 depconv)
3×3 2B Açılan (Dilated) Evrişim (3×3 dilconv)
5×5 2B Açılan (Dilated) Evrişim (5×5 dilconv)
7×7 2B Açılan (Dilated) Evrişim (7×7 dilconv)
Atlama bağlantısı (Skip-Connection - skip)

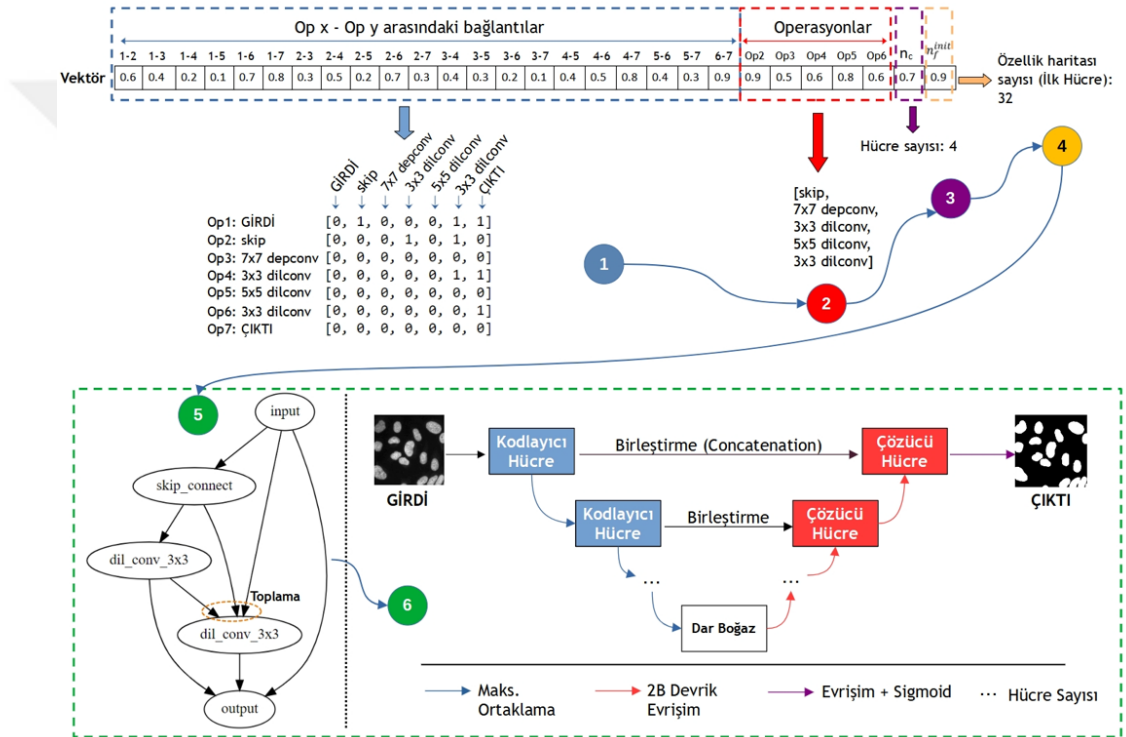
2.1.2.2. Mimari Kodlama

Çizgeler çeşitli şekillerde temsil edilebilir. İkili matrisler, çizgelerin kodlanması için kullanılan en yaygın temsillerden biridir. Yapılan tez çalışmasında, her hücre içindeki DAG'ları temsil etmek için matrisler kullanılmıştır. Maksimum operasyon sayısı 7 olarak belirlendiğinden dolayı, düğümler arasındaki bağlantıları göstermek için 7×7 'lik bir “ikili üst üçgensel matris” (binary upper-triangular matrix) kullanıldı. Önerilen kodlama yöntemi kullanılarak 337 milyar farklı çizge üretilebilir ($2^{21} * 11^5 = 337 \text{ Milyar}$). Ancak bu sayı, maksimum kenar kısıtlaması, “izomorfik” (isomorphic) ağlar (kodlaması farklı olsa bile aynı işlemi gerçekleştiren çizgeler) ve uygun olmayan çizgeler (girdi düğümünden çıktı düğümüne erişilebilecek herhangi bir yol içermeyen çizgeler - infeasible graphs) nedeniyle azalacaktır (Ying vd., 2019).

NAS için literatürde “ayrık” (discrete) ve “sürekli” (continuous) arama uzayları kullanılmıştır. Yapılan çalışmalar incelendiğinde, sürekli bir arama uzayı kullanmanın daha iyi sonuçlar verdiği görülmektedir (Awad vd., 2020). Bu nedenle, yapılan tez çalışmasında, arama uzayını daha iyi araştırmak amacıyla çözümler sürekli uzayda temsil edilmektedir. Çözümleri temsil etmek için 28 uzunluğunda bir vektör kullanılmıştır. Bu vektördeki ilk 21 eleman, düğümler arasındaki bağlantıları gösterir; sonraki 5 eleman seçilen operasyonları gösterir ve son iki eleman sırasıyla hücre sayısını ve özellik haritalarının sayısını gösterir. Vektördeki tüm bu değerler $[0, 1]$ aralığındadır ve doğrudan ağ mimarisini oluşturmak için kullanılamaz. Bu nedenle, sürekli değerler ile ağ mimarisi arasında bir “eşleme” (mapping) işlemi gerekmektedir (Awad vd., 2020).

Eşleme aşamasında, sürekli değerler, seçilebilecek değerlerin sayısına eşit olacak şekilde parçalara bölünür (Awad vd., 2020). Bu vektördeki ilk 21 eleman, düğümler arasındaki bağlantıları temsil eder. Bu değerler sadece ikili değer alabilir: eğer değer 1 ise iki düğüm arasında bir bağlantı vardır; aksi halde bağlantı yoktur. Yani bu vektördeki ilk 21 eleman için seçilebilecek iki değer vardır: $\{0, 1\}$. Bu iki değer için $[0, 1]$ aralığı iki eşit parçaya ayrılır; eğer eşlemesi yapılacak olan sürekli değer $[0, 0.5)$ aralığında ise ilgili eleman 0 olarak kodlanır, $[0.5, 1)$ aralığında ise ilgili eleman 1 olarak kodlanır. Operasyonlar için de eşleme işlemi benzer şekilde yapılır. Operasyonlar için de eşleme işlemi benzer şekilde yapılır. Operasyonlar için seçilebilecek 11 değer vardır; $1 \times 1 \text{ conv}$, $3 \times 3 \text{ conv}$, $5 \times 5 \text{ conv}$, $7 \times 7 \text{ conv}$, $3 \times 3 \text{ depconv}$, $5 \times 5 \text{ depconv}$, $7 \times 7 \text{ depconv}$, $3 \times 3 \text{ dilconv}$, $5 \times 5 \text{ dilconv}$, $7 \times 7 \text{ dilconv}$, skip . Eşlemesi yapılacak olan sürekli değer $[1/11, 2/11)$ aralığında ise $1 \times 1 \text{ conv}$ operasyonu, $[2/11, 3/11)$ aralığında ise $3 \times 3 \text{ conv}$ operasyonu seçilir. Eşleme işlemi diğer tüm operasyonlar için benzer şekilde gerçekleştirilir. Şekil 2.2, üretilen sürekli vektörden UNAS-Net mimarisine nasıl geçildiğini göstermektedir. 1'den 6'ya kadar olan daireler işlem sırasını gösterir. Dairelerin renkleri ve oklarla gösterilen işlemler birbiri ile doğrudan ilişkilidir. Örneğin: Şekil 2.2'de, 1 numaralı daire mavi renk ile gösterilmiştir ve bu adımda, mavi okla gösterilen komşuluk matrisi oluşturma işlemi gerçekleştirilir. Oluşturulan bir çözüm vektöründen UNAS-Net mimarisi oluşturulurken aşağıdaki adımlar sırayla uygulanır:

- 1) Komşuluk matrisi oluşturulur,
- 2) Seçilen operasyonlar belirlenir,
- 3) Hücre sayısı belirlenir,
- 4) Özellik haritalarının sayısı tanımlanır,
- 5) Çizge, komşuluk matrisi ve seçilen operasyonlar kullanılarak oluşturulur ve her hücrede kullanılır,
- 6) Önceki adımlarda elde edilen çizge ile UNAS-Net oluşturulur.



Şekil 2.2: Çözüm vektöründen UNAS-Net mimarisinin oluşturulması

2.1.2.3. Meta-Sezgisel Yöntemler

Gerçekleştirilen çalışmada farklı sinir ağları oluşturmak için kullanılan meta-sezgisel yöntemler şu şekildedir: Diferansiyel Gelişim (DE), Karşıtlık Temelli Diferansiyel Gelişim (ODE) ve Yerel Arama (LS). Popülasyon tabanlı evrimsel algoritma olarak kategorize edilen DE, Price ve Storn tarafından 1995 yılında önerilmiştir (Storn & Price, 1995). Yalnızca birkaç kontrol parametresine sahip, etkili, gürbüz (robust) ve basit bir global optimizasyon algoritmasıdır. DE, farklı kıyaslama veri kümeleri ve gerçek dünya problemleri için yakınsama hızı ve gürbüzlük açısından diğer birçok optimizasyon yönteminden daha iyi performans göstermektedir. Özellikle sürekli uzayın kullanıldığı NAS çalışmalarında (Awad vd., 2020) başarılı sonuçlar elde etmiştir. Bu çalışmada kullanılan DE ve ODE yöntemleri Bölüm 1.5'te detaylı olarak anlatılmıştır.

Yerel Arama (LS) yöntemi, NAS çalışmalarında temel yaklaşım (basis approach) olarak kullanılır (White vd., 2021; Ottelander vd., 2020). Bu nedenle, LS üçüncü meta-sezgisel olarak seçilmiştir.

Bu çalışmada kullanılan LS algoritmasının adımları aşağıdaki gibidir:

- (1) 10 rastgele çözüm üretilir.
- (2) Bu çözümler arasından en iyisi seçilir.
- (3) En iyi çözümün komşuları oluşturulur.
- (4) Bu çözümün komşularından herhangi biri en iyi çözümden daha iyiyse, bir sonraki adımda bu çözüm kullanılır ve sonlandırma koşulu sağlanana kadar 3 ve 4. adımlar tekrarlanır.

2.1.3. Deneysel Çalışmalar

Önerilen yöntemlerin performansını değerlendirmek için yapılan çalışmada iki farklı veri seti üzerinde deneyler yapılmıştır. Deneysel çalışmalar için gerçekleştirilen testler aşağıdaki özelliklere sahip bilgisayarda gerçekleştirilmiştir: Ryzen 5600X işlemci, 12 GB RTX 3060 GPU, 16 GB RAM. Farklı UNAS-Net modelleri oluşturmak için Pytorch kütüphanesi kullanılmıştır ve üretilen tüm modeller için aşağıdaki hiper parametreler sabit olarak seçilmiştir:

- Optimize edici: Adam,
- Kayıp fonksiyonu: Dice,

- Değerlendirme Metrikleri: F1-Skoru ve Kesişimlerin Birleşimi (IoU) (Calisto & Lai-Yuen, 2021),
- Öğrenme Hızı: 0.001 ve
- Yığın Boyutu: 8.

Üretilen her çözüm, tüm eğitim veri kümesi ile uzun eğitim dönemleri boyunca eğitilir ve üretilen çözümün kalitesi doğrulama veri kümesi üzerinde değerlendirilir. Ancak, bu adımları her çözüm için uygulamak verimli değildir. Bu nedenle, eğitim ve doğrulama kümelerinin yarısı, meta-sezgisel (Gülcü ve Kuş, 2020) yöntem tarafından üretilen ağırları eğitmek ve değerlendirmek için kullanılır. Değerlendirmelerin adil olması için her çözümde aynı eğitim ve doğrulama kümeleri kullanılır. Ayrıca, daha fazla çözümü değerlendirmek için “erken durdurma” (Early Stopping) yöntemi kullanılır. Her çözüm, maksimum 36 eğitim dönemi (Ying vd., 2019) boyunca eğitim kümesinin yarısı ile eğitilir. Eğitim dönemleri sırasında elde edilen doğrulama kaybı değeri art arda üç dönem için iyileşmezse, eğitim ilgili dönemde kesintiye uğrar. Eğitim dönemlerinin sonunda eğitilen çözüm tarafından elde edilen en iyi doğrulama F1-Skor değeri, bu çözümün kalitesini değerlendirmek için kullanılır. 500 çözüm eğitildiğinde meta-sezgiseller sonlandırılır. Çalıştırma sonunda elde edilen en iyi 3 model, tüm eğitim veri seti ile uzun eğitim dönemleri boyunca eğitilir ve tüm test seti ile test edilir. Test seti üzerinde elde edilen sonuçlar, modelin gerçek performansı olarak kullanılır.

2.1.3.1. Veri Kümesi

Üretilen mimarilerin segmentasyon performansı iki farklı hücre veri kümesi üzerinde değerlendirilmiştir: Optofil ve Cell Nuclei (Aydın vd, 2022). Bölüm 1.6’da bu veri kümeleri açıklanmıştır. Yapılan tez çalışmasında, her iki veri kümesi için de veri çoğaltma (data augmentation) işlemi uygulanmamıştır.

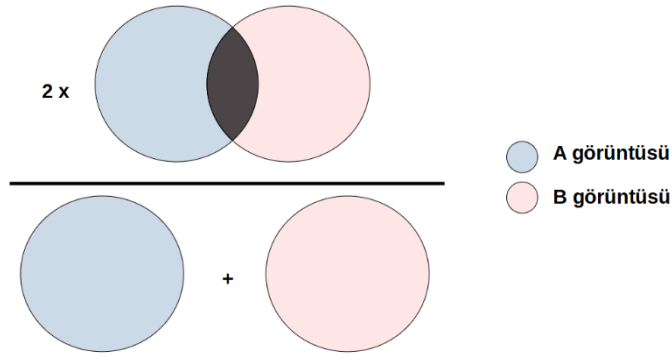
2.1.4. Sonuçlar ve Tartışma

Bu bölümde, UNAS-Net ile hücre segmentasyonu için yapılan deneysel çalışmaların sonuçları rapor edilmiş ve tartışılmıştır. U-Net omurgasının (backbone) optimizasyonu için DE, ODE ve LS olmak üzere üç meta-sezgisel yöntem kullanılmıştır. Seçilen meta-sezgisel yöntemler ile üretilen UNAS-Net ağırları

URes34-Net (kodlayıcı ve kod çözücü olarak Res-Net-34 mimarisini kullanan U-Net modeli) ile karşılaştırılmıştır.

Biyomedikal görüntü segmentasyonu problemi için geliştirilen modeller, yüksek segmentasyon performansına ve kısa çıkarım süresine (short inference time) sahip olmalıdır. Bu nedenle, modelleri segmentasyon performansı ve hesaplama karmaşıklığı açısından değerlendirmek önemlidir. Bu çalışmada kullanılan tüm yöntemlerin performansı, F1-Skoru, IoU ve kayan noktalı operasyonların (FLOPs) sayısı açısından karşılaştırılmıştır. Bu değerlendirme metrikleri şu şekilde açıklanabilir:

F1-Skoru (Dice Katsayısı – Dice Coefficient): Üretilen görüntülerin segmentasyon performansının değerlendirilmesi için sıklıkla kullanılır. İki görüntü arasındaki kesişen piksellerin sayısı iki ile çarpılır ve bu sonuç iki görüntüdeki piksellerin toplamına bölünür. Şekil 2.3'te F1-Skoru hesaplaması gösterilmiştir.



Şekil 2.3: F1-Skoru hesaplaması

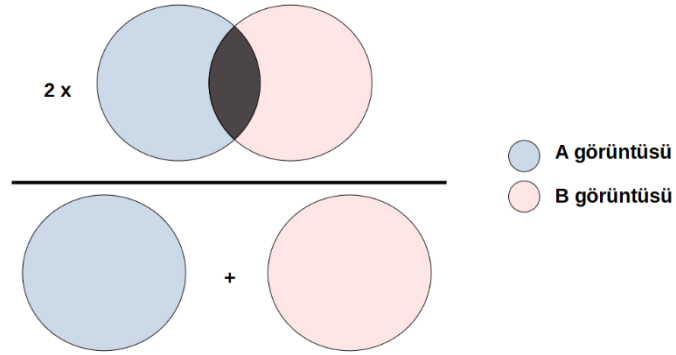
Birleşimlerin Kesişimi (Intersection over Union - IoU): Segmentasyon performansının değerlendirilmesi için kullanılan bir diğer metriktir. F1-Skoru metriğinden farklı olarak kesişen piksellerin sayısı iki ile çarpılmaz. Şekil 2.4'te IoU hesaplaması gösterilmiştir.

Kayan Noktalı Operasyon Sayısı (Floating Point Operations - FLOPs): Derin sinir ağlarının hesaplama karmaşıklığının, hesaplama yükünün hesaplanması için kullanılmaktadır. Kullanılan hesaplama işlemi denklem 2.1'de gösterilmiştir. Bu denklemde, F_h : Filtre yüksekliğini, F_w : Filtre genişliğini, C_{in} : Girdi özellik haritası sayısını, C_{out} : Çıkan özellik haritası sayısını, H_{out} : Uygulanan operasyon sonucu

oluşan görüntünün yüksekliğini, W_{out} : Uygulanan operasyon sonucu oluşan görüntünün genişliğini göstermektedir.

$FLOPs = [(F_h \times F_w \times C_{in}) \times C_{out} + C_{out}] \times (H_{out} \times W_{out})$	(2.1)
---	--------------

Tablo 2.2’de segmentasyon performansı ve hesaplama karmaşıklığı açısından farklı yöntemler karşılaştırılmaktadır. Bu tablodaki değerler, Optofil ve Hücre Çekirdeği (Cell Nuclei) veri kümeleri üzerinde elde edilen sonuçları göstermektedir ve elde edilen sonuçlar beş farklı rastgele sayısı dizisi (seed) için elde edilen ortalama sonuçlardır. Tablo 2.2’deki değerler karşılaştırılırken yüksek F1-Skoru ve IoU değerlerine sahip olan yöntemler daha başarılı olarak kabul edilir. FLOPs değerleri açısından karşılaştırma yapıldığında ise düşük FLOPs değerlerine sahip çözüm üreten yöntemler daha başarılıdır. Elde edilen sonuçlara bakıldığında, ODE’nin Optofil veri seti için en iyi F1-Skor ve IoU değerini elde ettiği görülmektedir. Ancak, hesaplama karmaşıklığı açısından bakıldığında ise DE’nin en iyi sonucu verdiği görülmektedir. DE, LS’den 0.14 puan daha kötü F1-Skoruna



Şekil 2.4: IoU hesaplaması

sahipken, hesaplama karmaşıklığı açısından 5.45 kat daha iyidir; ODE’den ise 0.34 puan daha kötü F1-Skoruna sahipken, hesaplama karmaşıklığı açısından 5.73 kat daha iyidir. Genel olarak bakıldığında ise seçilen meta-sezgisel yöntemlerin, Optofil veri seti için hem segmentasyon performansı hem de hesaplama karmaşıklığı açısından URes34-Net’ten daha iyi performans gösterdiği görülmektedir. Bu sonucu göz önünde bulundurarak, meta-sezgisel yöntemlerin daha az hesaplama karmaşıklığı ile U-Net’in segmentasyon performansını iyileştirdiği sonucuna varabiliriz.

Cell Nuclei veri seti için elde edilen sonuçlara bakıldığında DE'nin F1-Skoru açısından diğer yöntemlerden daha iyi olduğu ve ODE'nin en az karmaşıklığa sahip modeli ürettiği görülmektedir. F1-Skor ve IoU değerleri açısından farklı yöntemler arasında anlamlı bir fark görülmemektedir. Ancak, hesaplama karmaşıklığı açısından ODE, LS'den 3.91 kat; URes34-Net'ten ise 2.94 kat, DE'den 1.65 kat daha iyidir. Sonuç olarak DE tabanlı yöntemlerin hem segmentasyon performansı hem de hesaplama karmaşıklığı açısından daha iyi sonuçlar verdiği görülmektedir.

Tablo 2.2: Segmentasyon performansı ve hesaplama karmaşıklığı açısından farklı yöntemlerin karşılaştırılması. FLOPs değerleri gigabayt cinsinden verilmiştir.

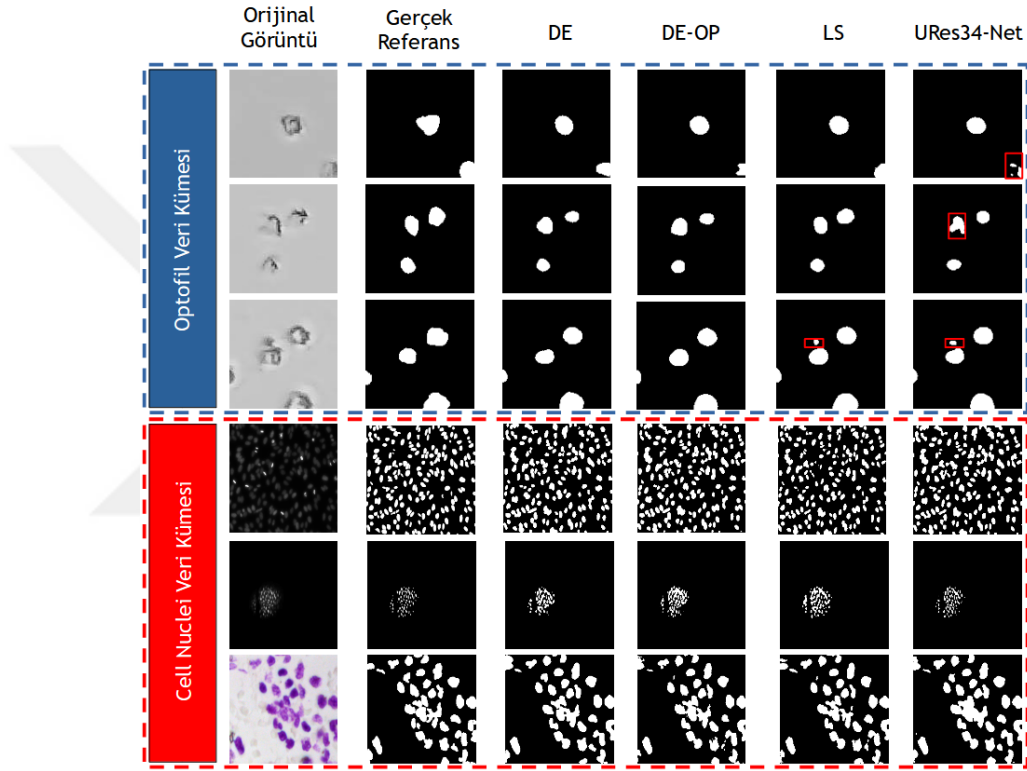
Yöntem	Optofil			Cell Nuclei		
	F1 Skoru	IoU	FLOPs (G)	F1 Skoru	IoU	FLOPs (G)
LS	83.22	70.57	22.80	91.16	83.95	82.99
DE	83.08	70.93	4.18	91.53	84.48	35.03
DE-OP	83.42	71.68	23.97	91.39	84.49	21.20
URes34-Net	82.69	70.01	30.85	91.11	83.83	62.51

Meta-sezgisellerin ve URes34-Net'in performansını daha iyi göstermek amacıyla her veri seti içerisinde üç örnek görüntü seçilmiştir. Şekil 2.5'te, seçilen örnek görüntüler, bu görüntülere ait gerçek referans görüntüleri ve kullanılan yöntemler tarafından elde edilen segmente edilmiş görüntüler gösterilmektedir. Meta-sezgiseller tarafından üretilen ağlar daha doğru segmentasyon gerçekleştirmiştir. Şekil 2.5'te gösterilen kırmızı kutular, Optofil veri seti için yanlış pozitif örnekleri gösterir. DE tabanlı yaklaşımlar, bu üç görüntü için iyi segmentasyon performansı göstermiştir.

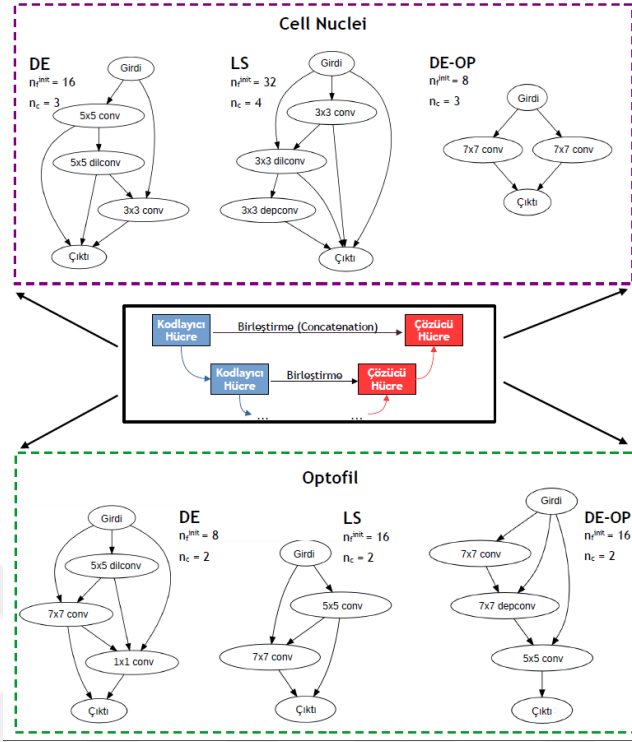
Şekil 2.6, her bir veri seti için farklı meta-sezgiseller tarafından elde edilen en iyi UNAS-Net modellerini göstermektedir. En iyi modeller incelendiğinde genellikle daha büyük filtre boyutuna sahip işlemlerin seçildiği görülmektedir. İlk filtre sayıları genellikle 8 ve 16'dır ve hücre sayıları genellikle 2 ve 3'tür. Ayrıca üretilen çoğu modelin maksimum 8 bağlantıya sahip olduğu görülmektedir.

Elde edilen sonuçlara bakıldığında DE tabanlı yöntemlerin, F1 Skoru, IoU ve FLOP sayısı açısından URes34-Net'den daha iyi performans gösterdiği

görülmektedir. DE, arama sırasında genetik operatörler aracılığıyla çeşitliliği koruyan popülasyona dayalı bir yaklaşım olduğundan, bu beklenen bir durumdur. Ancak, LS tek noktalı bir arama yöntemidir. Bu çalışmada, mevcut çözümde Hamming uzaklığı 1 olan yeni bir çözüm üretilmiştir. Bu nedenle, LS yerel optimumda takılabilir. Yerel optimumdan kaçmak için NAS için Yenilenen Yerel Arama (Iterated Local Search) kullanılabilir.



Şekil 2.5: Dört farklı yöntemle yapılan deneyler sonucunda farklı hücre görüntüleri için elde edilen segmente edilmiş görüntüler. Kırmızı kareler, segmentasyonun yanlış yapıldığı bölgeleri temsil etmektedir.



 ekil 2.6: İki farklı veri k mesi i in meta-sezgiseller tarafından olu turulan en iyi UNAS-Net modelleri.

2.2. BEYİN DAMAR SEGMENTASYONU İ İN UNAS-NET

2.2.1. Giri 

Beyin, y ksek miktarda enerji t keten kırılğan bir organdır. Beynin korunması kan-beyin bariyeri tarafından saėlanır ve enerji kaynakları serebral damar sistemi aracılıėıyla saėlanır. Bu yapılarından herhangi birinin bozulmasının, Alzheimer hastalığı, multipl skleroz, serebral k   k damar hastalığı, vask ler demans dahil olmak  zere bir ok n rodejeneratif bozuklukta hastalık mekanizmasına  nemli bir katkısının olduėu kabul edilmektedir (Sweeney vd., 2018). Bu nedenle, serebral damarların g r nt lenmesi ve analizi, biyomedikal ara tırmalarda, serebrovask ler sistemin bozulmasına yol a an yeni mekanizmaları ortaya  ıkarmak ve yeni tedavi hedefleri bulmak i in sıklıkla kullanılır.

Vask lat r, aė benzeri t b ler bir yapıdır ve bu karma ık yapıdaki uzunluk ve dallanma gibi damarla ilgili parametreleri manuel y ntemlerle g venilir bir  ekilde  l mek zordur. Bu nedenle, bu zorlukların  stesinden gelmek i in daha  nce otomatik veya yarı otomatik y ntemler  nerildi (Eladawi vd., 2017; Moccia vd.,

2018). Bununla birlikte, yüksek ve düzensiz arka plan sinyali, kan damarlarının doğru tanımlanmasını engelleyebilen ve bu tür analizlerde artan hata oranına yol açabilen mikroskopide sıklıkla karşılaşılan bir sorundur. Bu nedenle, doğru segmentasyon, doğru bir damar analizi için kritik bir adımdır.

Patoloji örneklerini etkin bir şekilde izlemek için ilk olarak tıbbi görüntülemelerde damar segmentasyonu yapılır (Akbar vd., 2019). Manuel damar segmentasyonu mümkün olsa da profesyonel beceri gerektiren zaman alıcı ve monoton bir iştir; bu nedenle zaman, iş gücü ve gözlemcilerin değişkenliğini raporlama açısından uygulanabilir bir prosedür değildir. Bu sebeple hem klinisyenlere hem de hastalara daha iyi tedavi planlaması ve zaman tasarrufu sağlamak amacıyla damar segmentasyonu gerçekleştiren yarı otomatik veya tam otomatik sistemlerin kullanılması bir zorunluluk haline gelmiştir. Aşağıdaki faktörler damar segmentasyonu sürecini zorlaştırır: (1) Damar şekilleri basit bir model izlemez; (2) Diğer yapıların varlığı (lezyonlar ve optik disk) (Rigamonti vd., 2012); Çeşitli görüntüleme koşulları (düşük görüntü kontrastı, gürültü ve patolojik sorunlar).

Damar segmentasyonu biyomedikal görüntü analizinde sıcak bir konu haline gelmiştir (Lesage vd.,2009; Jia vd., 2021). Damarların segmentasyonu için çeşitli yaklaşımlar önerilmiştir: "el yapımı" (hand-crafted) ve "denetimli" (supervised) yöntemler. El yapımı yöntemler, vasküler yapıların doğal özelliklerine dayanırken, denetimli yöntemler etiketli gerçek referans görüntülere (ground-truth) dayanır. Denetimli yöntemlerle öğrenilen özellikler, analitik yöntemlerle tanımlanması zor olan daha belirgin özelliklerdir (Rigamonti vd., 2012) ve daha iyi segmentasyon performansı elde etme eğilimindedir. Denetimli yöntemler arasında yer alan derin sinir ağı mimarileri, damar segmentasyonunda dikkate değer bir başarı göstermiştir (Soomro vd., 2019; Chen vd., 2021; Xiangbin vd., 2021; Nahian vd., 2021; Babu vd., 2022; Dong vd., 2022).

Hiper Parametre Optimizasyonu (HPO) ve Sinir Mimarisi Arama (Neural Architecture Search - NAS), otomatik makine öğrenimi (AutoML- Hutter vd., 2019) için en yaygın problemlerden bazılarıdır. Her makine öğrenimi sisteminin ayarlanması gereken birçok hiper parametresi vardır ve bu hiper parametrelerin seçimi performansı doğrudan etkiler. Sinir Mimarisi Araması için önerilen yaklaşımlar, belirli bir makine öğrenimi problemi için en uygun sinir ağını bulmayı

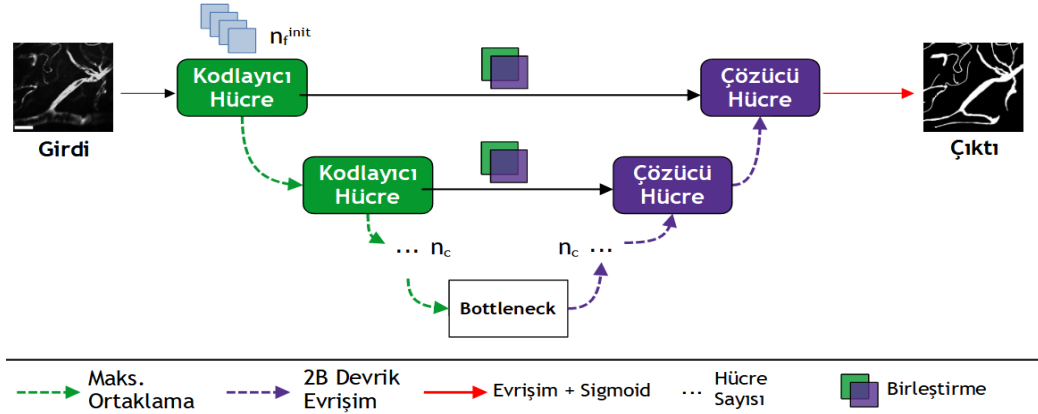
amaçlar. Bu nedenle, NAS çalışmaları, HPO çalışmalarının bir alt kümesi olarak düşünülebilir. Sinir ağları, seçilen yöntem (genellikle sezgisel yöntemler) kullanılarak önceden tanımlanmış arama uzayında oluşturulur ve performansları belirli bir veri kümesi üzerinde değerlendirilir. Üretilen sinir ağlarının düşük hesaplama karmaşıklığının yanı sıra iyi bir performans sağlaması beklenir.

Yapılan tez çalışmasında, 2B beyin damar görüntülerinin segmentasyonu için diferansiyel gelişim tabanlı sinir mimarisi arama yaklaşımları test edilmiştir (Kuş vd., 2023b). Optimal ağ mimarilerini aramak için “Geleneksel Diferansiyel Gelişim (Differential Evolution - DE)” ve “Karşıtlık Temelli Diferansiyel Gelişim (Opposition Based Differential Evolution - ODE)” kullanıldı. U-Net tabanlı modellerin biyomedikal görüntülerin segmentasyonunda başarılı olduğu gösterildiğinden, bu tez çalışmasında NAS için U-Net ve Attention-UNet mimarileri kullanılmıştır. NAS çalışmaları için sıklıkla kullanılan arama uzayları olan Nas-Bench-101(Ying vd., 2019) ve DARTS (Liu vd., 2018)’in avantajlarını birleştiren yeni bir arama uzayı önerilmiştir. Buna göre, *UNAS-Net + DE*, *UNAS-Net + ODE*, *Attention UNAS-Net + DE* ve *Attention UNAS-Net + ODE* olmak üzere dört farklı evrime dayalı (evolution-based) NAS yaklaşımı sunulmuştur. Deneyler, iki fotonlu bir mikroskoptan alınan görüntülerden oluşan iki veri kümesinin bir kombinasyonu üzerinde gerçekleştirilmiştir. Önerilen yaklaşımlar, yirmi bir segmentasyon performans ölçütü ve model karmaşıklığı açısından U-Net, Attention-UNet ve ResNet + U-Net ağı ile karşılaştırılmıştır. Deneysel sonuçlara dayanarak, UNAS-Net, farklı performans ölçütlerine göre segmentasyon performansını iyileştirmiş ve temel alınan U-Net ve Attention U-Net modellerinin hesaplama karmaşıklığını azaltmıştır. Ayrıca, DE tabanlı arama algoritmalarında tek bir amaç (single objective), yani kayıp fonksiyonunun (loss function) en aza indirilmesi düşünülse de bu yöntemler önerilen hücre tabanlı mikro arama uzayı (cell-based micro search space) ve makro arama uzayı (macro search space) sebebiyle hem yüksek segmentasyon performansına hem de düşük çıkarım sürelerine sahip modeller üretmiştir.

2.2.2. Yöntem

2.2.2.1. Arama Uzayı

UNAS-Net, sırasıyla alt örnekleme (downsampling) ve üst örnekleme (upsampling) işlemlerini gerçekleştiren kodlayıcı (encoder) ve kod çözücü (decoder) hücrelerden oluşur. Kodlayıcı hücreler, bir dizi evrişim işlemi uygulayarak özellik çıkarımını gerçekleştirir. Görüntü boyutunu küçültmek için kullanılan 2 adımlı (stride) maksimum ortaklama (max-pooling) işlemi, kodlayıcı hücreleri takip eder. Öte yandan, kod çözücü hücreler, kodlayıcı hücrelerde çıkarılan öznitelikleri kullanarak görüntüyü yeniden oluşturur ve devrik evrişimi (transpose convolution) kullanarak girdi boyutunu artırır. Son olarak, en son kod çözücü hücreden gelen çıktıya 1×1 boyutunda filtreye sahip 2B evrişim işlemi uygulanır ve ardından segmente edilmiş görüntüyü elde etmek için “*sigmoid*” aktivasyon fonksiyonu kullanılır. UNAS-Net'in genel yapısı Şekil 2.7'de gösterilmektedir.

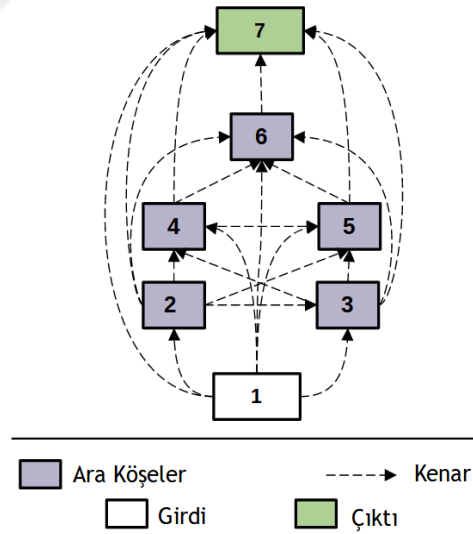


Şekil 2.7: UNAS-Net'in genel yapısı. Ölçek çubuğu $100 \mu\text{m}$ 'yi gösterir

Yapılan çalışmada, kodlayıcı ve kod çözücü hücrelerin yapısını tanımlamak için “*hücre tabanlı mikro arama yöntemi*” (Liu vd., 2018; Ying vd., 2019) kullanılmıştır. Her hücre, yönlü ve döngüsel olmayan bir çizge olarak temsil edilir ($G = (V, E)$), burada V ve E , sırasıyla köşeler ve kenarlar kümesini belirtir. Şekil 2.8'de örnek bir hücre yapısı gösterilmektedir. Çizge, bir girdi (v_1), bir çıktı (v_7) ve en çok 5 ara köşeden (intermediate vertices) ($v_2 - v_6$) oluşur. Her bir ara köşe ($v \in V \setminus \{v_1, v_7\}$), gelen girdiye uygulanacak işlemi temsil eden bir $l \in L$ etiketine (label) sahiptir, burada L etiket kümesidir. Yönlü her kenar ($e \in E$), köşeler arasındaki bağlantıları temsil eder. Bu tez çalışmasında arama uzayını daraltmak için köşe sayısı

($|V| \leq 7$), kenar sayısı ($|E| \leq 9$) ve olası işlem sayısı ($|L| \leq 14$) olacak şekilde kısıtlandı. Tablo 2.3'te bu tez çalışması için seçilen 14 farklı işlem gösterilmiştir. Bu işlemler DARTS (Liu vd., 2018) makalesi referans alınarak seçilmiştir.

Makro arama uzayında ise iki hiper parametre vardır: Hücre sayısı ve ilk hücredeki özellik haritalarının sayısı (Şekil 2.7). UNAS-Net mimarisi eşit sayıda kodlayıcı ve kod çözücü hücreye sahiptir. Mimarideki toplam hücre sayısı (N_c), kodlayıcı hücre sayısı (n_c), kod çözücü hücre sayısı (n_c) ve darboğaz hücresinin toplamı olarak hesaplanır ($N_c = 2n_c + 1$). Hücre sayısı, $n_c \in \{2, 3, 4, 5\}$, arama algoritmaları tarafından optimize edilecek bir karar değişkenidir. Öte yandan, verilen girdiden çıkarılan özellik haritalarının sayısı ağın performansını doğrudan etkiler. $n_f^{init} \in \{8, 16, 32\}$, ilk kodlayıcı hücredeki işlemler sonucunda oluşturulacak özellik haritalarının sayısını belirleyen bir diğer karar değişkenidir. Özellik haritalarının sayısı, her maksimum ortaklama işleminden sonra iki katına çıkar.



Şekil 2.8: Hücre tabanlı mikro arama uzayındaki kodlayıcı ve kod çözücü hücrelerin

Tablo 2.3: Ara köşeler için seçebilecek olası işlemler

Operasyon Adı
1×1 2B Evrişim (1 × 1 conv)
3×3 2B Evrişim (3 × 3 conv)
5×5 2B Evrişim (5× 5 conv)
7×7 2B Evrişim (7× 7 conv)
3 × 3 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim (3×3 depconv)
5 × 5 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim (5×5 depconv)
7×7 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim (7×7 depconv)
3×3 2B Açılan (Dilated) Evrişim (3×3 dilconv)
5×5 2B Açılan (Dilated) Evrişim (5×5 dilconv)
7×7 2B Açılan (Dilated) Evrişim (7×7 dilconv)
3×3 2B Asimetrik Evrişim (3×3 asymconv)
5×5 2B Asimetrik Evrişim (5×5 asymconv)
7×7 2B Asimetrik Evrişim (7×7 asymconv)
Atlama bağlantısı (Skip-Connection - skip)

2.2.2.2. Mimari Kodlama

Bu çalışmada, kodlayıcı/kod çözücü hücrelerden (21 karar değişkeni) oluşan bir aday çözümün yapısının temsil edilmesi için 28 elemandan oluşan ($\in [0,1]$) bir vektör kullanılır, evrişim işlemleri için 5 karar değişkeni ve makro arama uzayındaki hiper parametreler için 2 karar değişkeni.

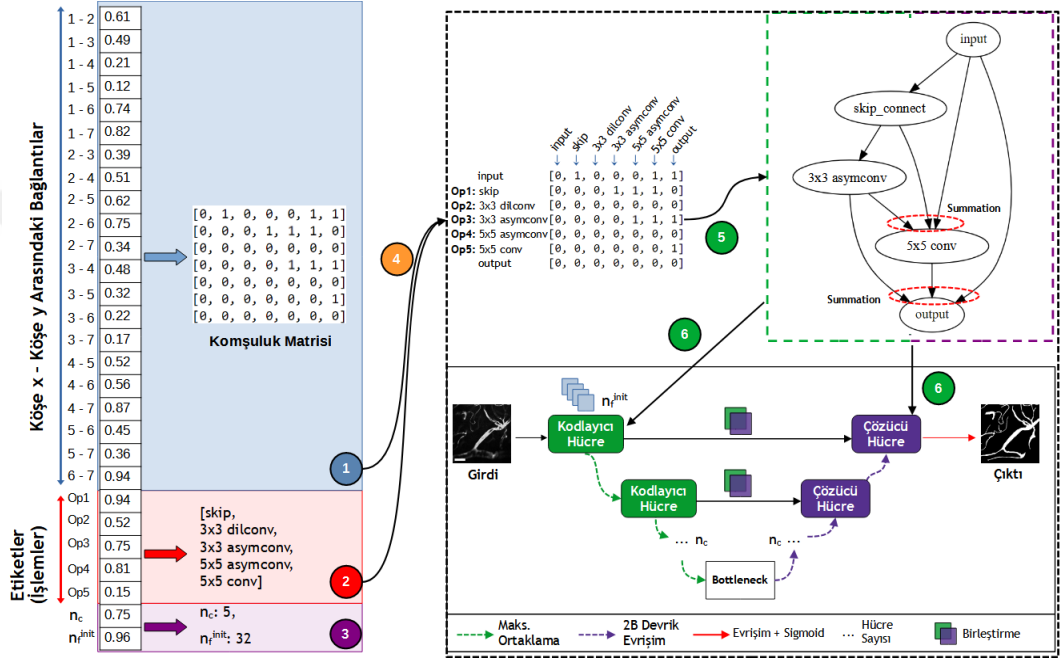
Hücreler yönlendirilmiş döngü içermeyen bir çizge (Directed Acyclic Graph - DAG) olarak temsil edildiğinden, DAG'yi tanımlamak için komşuluk matrisi kullanıldı. Maksimum 7 köşeden oluşan hücre yapısındaki komşuluklar, 7×7 'lik komşuluk matrisinin üst-üçgensel alanı ile temsil edilmiştir ve bu matriste 0 ve 1 değerleri iki köşenin birbirine bağlı olup olmadığını göstermek için kullanılır. Aday çözümdeki ilk 21 değişken, kodlayıcı/kod çözücü hücrenin yapısını temsil eden komşuluk matrisini gösterir. Ayrıca, hücre yapısında her biri yalnızca bir etikete sahip beş ara köşe vardır. Bu nedenle, aday çözümü temsil eden vektörün sonraki 5 elemanı (22 – 26), Tablo 2.3 içerisindeki on dört olası etiket arasından seçilen

etiketleri gösterir. Son olarak, aday çözümdeki son iki eleman, hücre sayısını (n_c) ve ilk hücrenin özellik haritalarının sayısını (n_f^{init}) gösterir.

Aday bir çözüm, gerçek değerlerden oluşan bir vektör tarafından temsil edildiğinden, gerçek değerli vektörden bir UNAS-Net mimarisi oluşturmak için bir eşleme işlemi (mapping) kullanılmalıdır. Şekil 2.9, örnek bir gerçek değerli vektör üzerinden eşleme işlemini adım adım gösterir. Bu adımlar şu şekilde verilmektedir:

- Çözümdeki ilk 21 eleman komşuluk matrisi için kullanılır. Çözümdeki her değer, köşeler arasında bir bağlantı olup olmadığını gösterir: Değer 0.5'ten büyükse, karşılık gelen köşe çifti arasında bir kenar olduğu anlamına gelir; bu nedenle, komşuluk matrisinde bu bağlantıya karşılık gelen giriş 1 olmalıdır. Şekil 2.9'da görüldüğü gibi, ilk eleman *köşe 1* ile *köşe 2* arasında bir bağlantı olup olmadığını gösterir. Değer 0.61 olduğu için komşuluk matrisinin birinci satırındaki ve ikinci sütundaki değer 1 olur. Ayrıca, ikinci eleman *köşe 1* ile *köşe 3* arasındaki bağlantı için kullanılır. Karşılık gelen değer 0.49'dur; bu nedenle, karşılık gelen değer 0 olur. Kalan elemanlar için, aynı işlemler gerçekleştirilir.
- Çözümdeki sonraki 5 eleman (22 - 26), seçilen işlemleri gösterir. On dört olası işlem (etiket) olduğundan, [0, 1] aralığı on dört eşit parçaya bölünür: [0, 0.07) aralığı, ilk olası işleme, yani 1×1 conv'a karşılık gelir (Tablo 2.3 içinde verilmiştir); [0.07, 0.14) aralığı, 3×3 conv'a karşılık gelir; [0.14, 0.21) aralığı, 5×5 conv'a karşılık gelir; ve benzeri. 22. eleman, hücre yapısında *köşe 2*'ye karşılık gelen etiketi belirtir. Karşılık gelen değer 0.94 olduğu için *köşe 2*'de 'skip' işlemi kullanılmıştır.
- Son iki eleman sırasıyla hücre sayısını (n_c) ve ilk hücrenin özellik haritalarının sayısını (n_f^{init}) gösterir. n_c dört farklı değer alabilir, bu nedenle [0, 1] aralığı dört eşit parçaya bölünür: [0, 0.25) aralığı 2 değerine karşılık gelir, [0.25, 0.50) aralığı 3 değerine karşılık gelir vb. Öte yandan, n_f^{init} karar değişkeni 8, 16 veya 32 değerlerini alabilir, yani [0, 1] aralığı üç eşit parçaya bölünür. Şekil 2.9'da, sondan ikinci ögenin değeri 0.75'tir, bu da n_c 'nin 5 olarak seçildiği anlamına gelir. Benzer şekilde, son öge 0.96'dır, bu da n_f^{init} değerinin 32 olduğunu gösterir.

- İlk iki adımın sonucunda komşuluk matrisi ve etiketler (her köşe için seçilen işlem) oluşturulur.
- Komşuluk matrisi ve etiketleri kullanarak kodlayıcı/kod çözücü hücrelerde kullanılacak olan hücre yapısı oluşturulur.
- UNAS-Net mimarisi, n_c , n_f^{init} ve hücre yapısı kullanılarak oluşturulur.

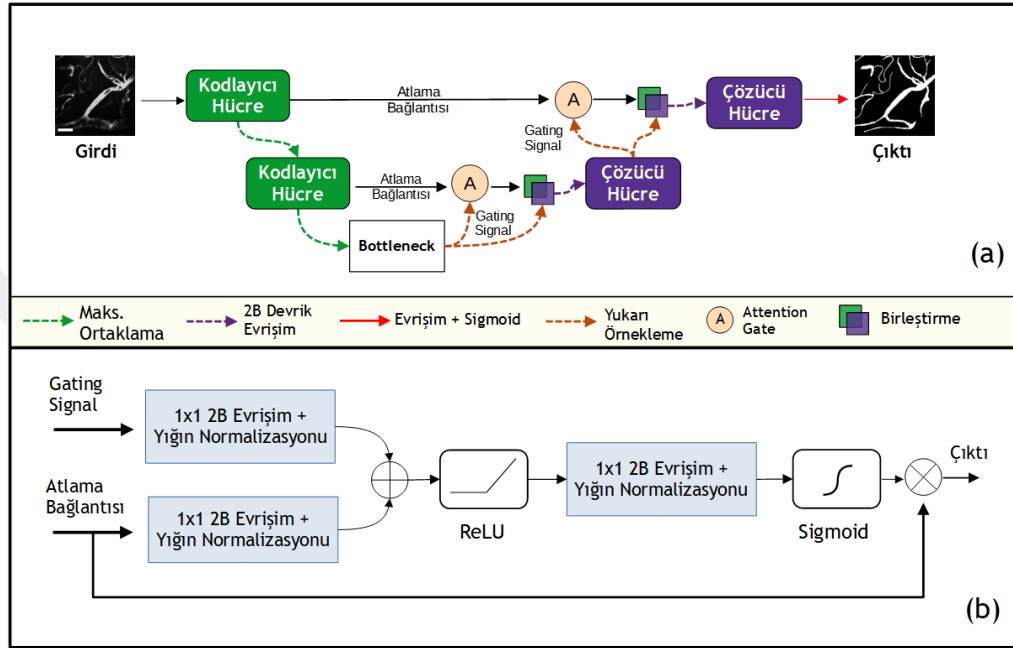


Şekil 2.9: UNAS-Net Mimarisi için kodlama (encoding - sol kısım) ve eşleme (mapping) adımları: Adım 1: Komşuluk matrisi oluşturulur; Adım 2: Etiketler seçilir; Adım 3: İki hiper parametre için değerler seçilir; Adım 4: Komşuluk matrisi etiketler ile birleştirilir.

2.2.2.3. Attention UNAS-Net

U-Net'in başka bir versiyonu olan Attention U-Net, tıbbi görüntülerin (Oktay vd., 2018) segmentasyonu için önerilmiştir. Dikkat kapıları (Attention gates) bu mimaride kullanılır. U-Net yapısında kodlama adımında öğrenilen özellikler, kod çözme adımından önce diğer özellikler ile birleştirilmelidir. Ancak ilk aşamalarda öğrenilen alt seviye özellikler (low-level features) ile sonraki aşamalarda öğrenilen üst seviye özellikler (high-level features) segmentasyon için aynı öneme sahip değildir. Bu nedenle, bu özelliklerin birleştirme aşamasında ağırlıklandırılması gerekir. Dikkat kapısı, farklı özellikleri ağırlıklandırmak için kullanılır.

Yapılan tez çalışmasında, Attention U-Net'i temel alan Attention UNAS-Net yapısı önerilmiştir. Bu yapı, UNAS-Net ile aynı arama uzayını ve kodlamayı kullanır. Ayrıca, kod çözme adımlarından önce dikkat kapılarından yararlanır. Attention UNAS-Net yapısı Şekil 2.10'da gösterilmiştir.



Şekil 2.10: Attention-UNAS-Net'e genel bakış. (a) Attention-UNAS-Net Mimarisi, (b) Dikkat Kapısı. Ölçek çubuğu 100 µm'yi gösterir.

2.2.3. Meta-Sezgisel Yöntemler

Yapılan çalışmada, farklı sinir ağları oluşturmak için kullanılan meta-sezgisel yöntemler şu şekildedir: Diferansiyel Gelişim (DE) ve Karşıtlık Temelli Diferansiyel Gelişim (ODE). Her iki meta-sezgisel yöntem de Bölüm 1.5'te detaylı olarak anlatılmıştır.

2.2.4. Deneysel Çalışmalar

Bu bölümde, UNAS-Net ile beyin damar segmentasyonu için yapılan deneysel çalışmalardan bahsedilmiştir. Önerilen modeller U-Net ve Attention U-Net dışında Kocak vd. (2021) tarafından önerilen URes-Net modeli ile de karşılaştırılmıştır. Tez çalışmasının bu bölümünde deneysel çalışmalarda kullanılan veri seti, performans ölçütleri ve yöntemler için ayrıntılı bilgi sunulmuştur.

2.2.4.1. URes-Net

Karşılaştırma amacıyla kullanılan ResNet + U-Net ağı (Koçak vd., 2021) iki kısımdan oluşur: birinci kısım daraltma yolu (narrowing path), ikinci kısım ise genişletme yolu (expansion path) olarak adlandırılır. Daraltma yolu, iki adet 3×3 boyutunda filtre kullanılan evrişim işleminin yinelenmesiyle oluşur ve katmanlar boyunca girdi verilerinin boyutu azaltılır. Girdi katmanından elde edilen özellik haritası sayısı 64 olup, daralan yolun her adımında bu sayı ikiye katlanmaktadır. Bu yolda, girdi boyutunu düşürmek için adım değeri 2 olan maksimum ortaklama yöntemi kullanılır. Genişletme yolunun her adımında, 2×2 boyutunda filtre ile yukarı örnekleme işlemi gerçekleştirilir ve ortaya çıkan özellik haritaları, aynı adımda daraltma yolundan gelen özellik haritaları ile birleştirilir.

2.2.4.2. Veri Kümesi

Deneyler, benzer iki veri kümesinin birleştirilmesiyle oluşturulan yeni bir veri kümesinde üzerinde gerçekleştirilmiştir: vesselNN ve KUVESG. Her iki veri kümesi de Bölüm 1.6’da anlatılmıştır.

2.2.4.3. Performans Ölçütleri

Derin öğrenmeye dayalı bir segmentasyon yönteminin performans değerlendirmesi için altı kategoriye ayrılan birden fazla performans ölçümü gerekir (Taha vd., 2015; Aydın vd., 2021). “Dice Score” ve “Intersection over Union” gibi örtüşmeye dayalı (overlap-based) ölçüler, kolayca yorumlanabilir, uygulanabilir ve karşılaştırılabilir oldukları için sıklıkla kullanılır (Dice vd., 1945; Zou vd., 2004; Yeghiazaryan vd., 2018). Hacim (volume), çift sayma (pair counting), olasılıksal (probabilistic) ve uzamsal mesafeye dayalı (spatial distance-based) vb. gibi ölçüt kategorileri ise daha az sıklıkla kullanılır. Damarların sınır bilgileri ve hizaları (alignment) bizim problemimiz için daha önemlidir. Bu nedenle, tez çalışmasında önerilen yöntemlerin performans değerlendirmeleri için Taha vd. (2015) tarafından önerilen beş kategoriye (hacim bazlı hariç) ait yirmi bir performans ölçütü kullanılmıştır. Bu performans ölçütleri Tablo 2.4’te listelenmiştir.

Tablo 2.4: Seçilen performans ölçütlerinin kısa özeti. Kategori sütunu, her bir performans ölçütünün ait olduğu performans ölçütü grubunu gösterir.

Kategori	Performans Ölçütü
Örtüşmeye Dayalı (Overlap Based)	Dice (DICE) Intersection over Union (IoU) Jaccard Index (JAC) Sensitivity (SNS) Specificity (SP) Sensibility (SB) Global consistency error (GCE) Conformity (CNF) Accuracy (ACC) Precision (PRC)
Çift Saymaya Dayalı (Pair Counting Based)	Rand Index (RI) Adjusted Rand Index (ARI)
Teorik Tabana Dayalı (Information Theoretic Based)	Mutual information (MI) Variation of information (VOI)
Olasılığa Dayalı (Probabilistic Based)	Interclass correlation (ICC) Probabilistic distance (PBD) Cohen's kappa (KAP) Area under ROC Curve (AUC)
Uzamsal Uzaklığa Dayalı (Spatial Distance Based)	Hausdorff distance (HD95) Average Hausdorff distance (AHD) Mahalanobis Distance (MHD)

2.2.4.4. Geliştirme Detayları

Derin öğrenme ağlarının gerçek performansının ölçülebilmesi için uzun eğitim dönemleri boyunca ağın tüm eğitim seti ile eğitilmesi ve öğrenme sürecinde yer almayan test seti üzerinde değerlendirilmesi gerekir. Ancak NAS çalışmalarında performansı GPU üzerinde değerlendirilmesi gereken birçok aday ağ bulunmaktadır. Tüm bu aday ağları, uzun eğitim periyotları boyunca tüm eğitim verileriyle eğitmek

mümkün değildir. Bu nedenle, gerçekleştirilen değerlendirme adımları iki aşamaya ayrılmıştır: Kısa vadeli değerlendirme ve uzun vadeli değerlendirme.

Kısa vadeli değerlendirme adımında, sezgisel yöntemin çalışması sırasında oluşturulan aday ağlar değerlendirilir. Bu adımda, oluşturulan aday ağları eğitmek için eğitim veri setinin yarısından oluşan bir alt küme kullanılır ve bu alt kümenin %10'u kullanılarak oluşturulan doğrulama kümesi üzerinden bir ağın kalitesi değerlendirilir. Bu setler oluşturulurken görüntüler rastgele seçilir ve oluşturulan ağların adil bir şekilde değerlendirilmesi için aynı görüntüler kullanılır. Toplam eğitim süresini daha verimli kullanmak için *Erken Durdurma (Early Stopping)* kullanıldı. Kısa vadeli değerlendirme adımında, her aday ağ maksimum 36 eğitim dönemi için eğitilir (Ying vd., 2019). Ancak, doğrulama kümesindeki bir ağ tarafından elde edilen doğrulama kaybı değeri, birbirini izleyen üç eğitim döneminde gelişmezse, eğitim o adımda kesilir. Bu şekilde, daha fazla aday ağ daha kısa sürede değerlendirildi. Eğitim sonunda elde edilen en iyi *dice score* değeri bu aday ağın kalitesini belirler ve çözümün uygunluk değeri olarak kullanılır. Seçilen sezgisel yöntemler, GPU üzerinde 500 ağ mimarisi eğitildikten sonra sonlandırılır.

Uzun vadeli değerlendirme adımında, sezgisel çalıştırma sonucunda elde edilen ilk beş ağın gerçek performansı değerlendirilir. Bu adımda, en iyi ağlar rastgele ağırlıklarla başlatılır (kısa süreli değerlendirme adımında öğrenilen ağırlıklar, adil bir değerlendirme yapmak için kullanılmaz) ve 200 eğitim dönemi boyunca tüm eğitim veri seti ile eğitilir. Eğitilen bu ağlar, daha önce eğitime dahil edilmemiş (kısa süreli değerlendirme adımı dahil) test setinde değerlendirilir ve ağın gerçek performansı elde edilir. Veri artırma (Data augmentation) çalışmanın hiçbir aşamasında uygulanmaz.

DE ve ODE'nin parametre ayarları Awad vd. (2020) ve Delacruz vd. (2020) tarafından önerilen şekilde seçildi: popülasyon boyutu (*NP*) 20 olarak ayarlandı; ölçeklendirme faktörü (*F*) 0.5'dir; çaprazlama olasılığı (*CR*) 0.5 olarak ayarlanmıştır. Ayrıca ODE'de atlama olasılığı (*JR*) 0.3 olarak seçilmiştir. NAS çalışması sırasında oluşturulan ağlar için Pytorch kütüphanesi kullanılmıştır. URes-Net ağı için Tensorflow 2 kütüphanesi kullanılmıştır. Aşağıdaki parametreler UNAS-Net ve URes-Net için ortak olarak kullanılmıştır: Optimize edici (Optimizer): Adam, Kayıp Fonksiyonu (Loss Function): Dice Loss, Öğrenme Oranı (Learning Rate): $1e-3$ ve

Yığın Boyutu (Batch Size): 2 (Bu değer hafıza sorunu nedeniyle seçilmiştir -Baldeon vd., 2021). Yapılan tez çalışmasında, deneysel çalışmalar için şu donanımlar kullanılmıştır: Ryzen 5600X işlemci, 12 GB RTX 3060 GPU, 16 GB RAM.

2.2.5. Sonuçlar ve Tartışma

2.2.5.1. U-Net Modellerinin Karşılaştırılması

Bu bölümde, yapılan deneysel çalışmaların sonuçları rapor edilmiş ve tartışılmıştır. Yapılan tez çalışmasında hem yüksek segmentasyon performansı hem de kısa çıkarım süresi elde eden modellerin üretilmesi hedeflenmiştir. Bu nedenle sayısal karşılaştırmalar iki farklı şekilde yapılır: Segmentasyon performansı ve model karmaşıklığı. Segmentasyon performansını değerlendirmek için Bölüm 2.2.4.3'te açıklanan 21 farklı ölçüt kullanılırken, model karmaşıklığının doğru ve güvenilir bir şekilde değerlendirilmesi için kayan nokta işlemleri (FLOP'lar) ölçütü kullanılır (Lu vd., 2019).

Önerilen yöntemler, tıbbi görüntü segmentasyonu için son teknoloji olarak kabul edilen U-Net ve Attention U-Net mimarileri ile karşılaştırılmıştır. Bu çalışmanın temel amacı, segmentasyon performansı ve model karmaşıklığı açısından temel yöntemlerden daha iyi modeller elde etmek için DE ve ODE sezgisellerini kullanmaktır. Segmentasyon performansı sonuçları Tablo 2.5'te gösterilmiştir. Bu tabloda gösterilen sonuçlar, her yöntemin beş bağımsız çalışmasından elde edilen sonuçların ortalamalarıdır. Tablo 2.5'te gösterilen ↑ ve ↓ işaretleri, önerilen yöntem kullanılarak temel alınan yöntemin performansının, karşılık gelen performans ölçütü açısından iyileştirilip iyileştirilmediğini gösterir. UNAS-Net için U-Net ve Attention-UNAS-Net için Attention U-Net temel alınan yöntem olarak seçilmiştir.

Tablo 2.5: Seçilen yöntemlerin farklı değerlendirme ölçütleri açısından karşılaştırılması. Her satırda, en iyi sonuç veren yaklaşım koyu renkle işaretlenmiştir.

Performans Ölçütü	U-Net	UNAS-Net		Attention U-Net	Attention-UNAS-Net		URes-Net
		DE	ODE		DE	ODE	
DICE	76.93	78.80↑	79.57↑	76.36	78.86↑	79.36↑	79.38
IoU	63.84	65.72↑	66.74↑	63.01	65.78↑	66.49↑	66.81
JAC	64.54	66.36↑	67.42↑	63.65	65.74↑	66.27↑	67.60
SNS	74.55	81.14↑	81.48↑	76.49	76.93↑	78.56↑	80.32
SP	98.39	97.52↓	97.81↓	97.71	98.02↑	97.90↑	97.85
SB	83.67	75.71↓	77.81↓	77.84	81.94↑	79.79↑	79.77
GCE	7.07	7.46↓	6.97↑	7.74	7.02↑	7.15↑	6.71
CNF	27.92	44.87↑	47.70↑	31.01	43.05↑	44.58↑	47.04
ACC	95.96	95.84↓	96.19↑	95.56	96.12↑	96.04↑	96.34
PRC	83.45	79.09↓	79.85↓	79.90	82.32↑	81.36↑	80.83
RI	92.39	92.15	92.75↑	91.69	92.63↑	92.50↑	93.01
ARI	71.14	72.38↑	73.66↑	69.88	72.34↑	72.62↑	73.97
MI	25.85	27.07↑	27.65↑	25.47	26.62↑	26.83↑	27.78
VOI	38.41	40.32↑	38.65↑	41.30	38.83↓	39.11↓	37.84
ICC	-3.44	-3.64↓	-3.36↑	-3.78	-3.40↑	-3.47↑	-3.21
PBD	36.04	27.57↑	26.15↑	34.49	28.47↑	27.71↑	26.48
KAP	75.15	76.79↑	77.78↑	74.27	76.46↑	76.82↑	77.89
AUC	86.47	89.33↑	89.64↑	87.10	87.48↑	88.23↑	89.09
HD95	26.63	14.72↑	15.93↑	30.81	13.79↑	11.33↑	10.56
AHD	9.73	9.38↑	8.94↑	10.22	9.40↑	9.52↑	9.02
MHD	13.64	13.54↑	12.05↑	14.48	12.00↑	12.48↑	10.93

Tablo 2.5, DE ve ODE sezgisel yöntemleri kullanılarak oluşturulan en iyi UNAS-Net ağının, çoğu segmentasyon ölçütünde U-Net'in segmentasyon performansından daha iyi performans gösterdiğini ortaya çıkarmaktadır. UNAS-Net + DE ve UNAS-Net + ODE sırasıyla 21 ölçütün 14'ünde ve 21 ölçütün 18'inde U-Net'i geride bıraktı. UNAS-Net + DE ve UNAS-Net + ODE sonuçları karşılaştırıldığında, UNAS-Net + ODE'nin 21 ölçütün 19'unda daha iyi segmentasyon performansı gösterdiği görülmektedir. Yöntemlerin model karmaşıklıkları kıyaslandığında ise UNAS-Net + DE ve UNAS-Net + ODE'nin UNAS-Net'ten sırasıyla 1.98 ve 5.19 kat daha az FLOPs (Floating Point Operations) gerçekleştirdiği Tablo 2.6'da gösterilmiştir. Model karmaşıklığı açısından, UNAS-Net + ODE tekniğinin UNAS-Net + DE'ye göre 2.62 kat daha az kayan noktalı işlem gerçekleştirdiği açıktır. Sonuç olarak UNAS-Net için ODE sezgiselinin hem bölütleme performansı hem de model karmaşıklığı açısından U-Net ve UNAS-Net +

DE'den daha iyi sonuçlar verdiği gözlemlenmiştir. Buna dayanarak, UNAS-Net için ODE sezgiselinin arama uzayını DE'den daha iyi araştırdığını söylenebilir.

Tablo 2.6: Model karmaşıklığı, çıkarım süresi ve boyutu açısından farklı yöntemlerin karşılaştırılması. FLOPs değerleri cıgabayt cinsinden; # Parametre ve Modelin Diskteki Boyutu ise megabayt cinsinden, CPU zamanı saniye cinsinden verilmiştir.

Yöntem	FLOPs (G)	# Parametre (M)	Model Boyutu (M)	CPU Zamanı
U-Net	436	31	121	32.72
UNAS-Net + DE	220	3	14	27.10
UNAS-Net + ODE	84	6	25	20.72
Attention U-Net	531	34	136	39.56
Attention-UNAS-Net + DE	58	5	21	13.86
Attention-UNAS-Net + ODE	175	10	40	27.55
URes-Net	436	34	407	32.46

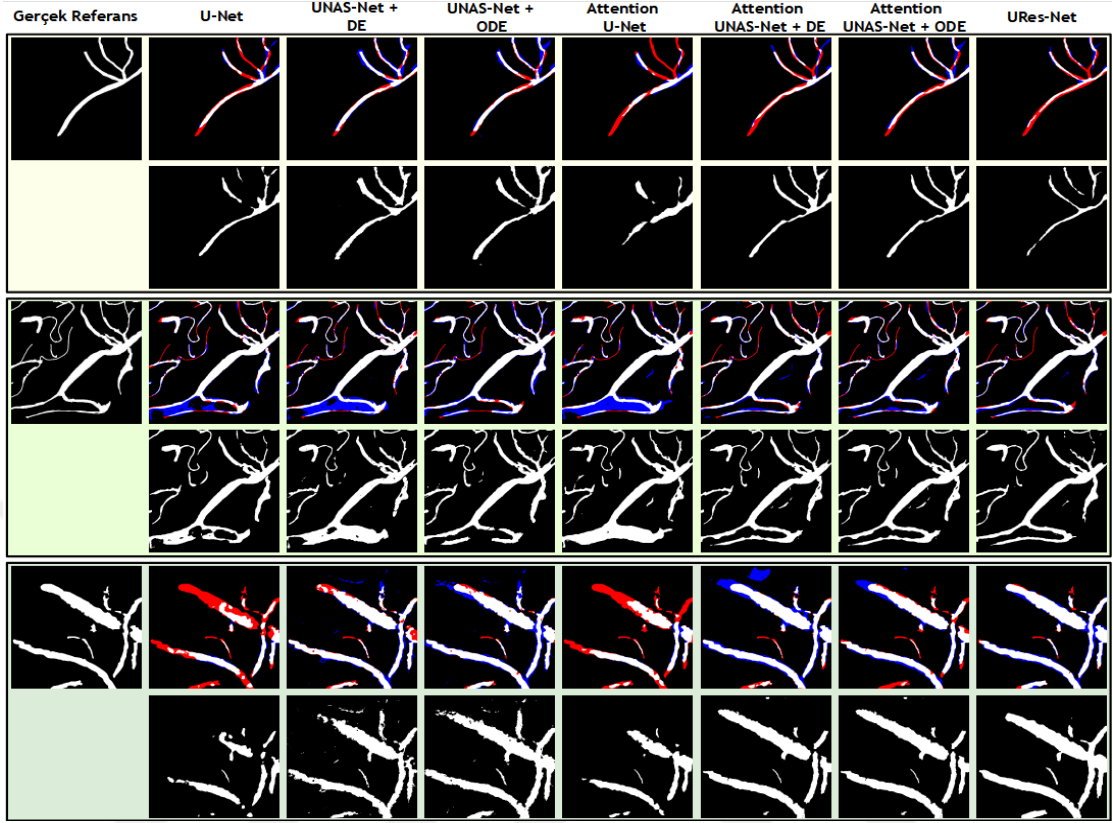
Önerilen bir diğer NAS tabanlı yöntem olan Attention-UNAS-Net, Attention U-Net ile karşılaştırıldı ve Attention U-Net ağıının segmentasyon performansı iyileştirildi. Tablo 2.5, Attention-UNAS-Net + DE ve Attention-UNAS-Net + ODE'nin, VOI dışındaki tüm ölçütler için, temel alınan Attention-UNet'ten daha iyi segmentasyon performansı elde ettiğini göstermektedir. Attention-UNAS-Net için DE ve ODE sezgiselleri arasında UNAS-Net'te olduğu gibi net bir fark yoktur. ODE sezgisel yöntemiyle üretilen en iyi ağ, 12 ölçüt açısından DE sezgiselinden daha iyi performans göstermiştir. Attention-UNAS-Net + DE ve Attention-UNAS-Net+ ODE yöntemleri, Attention-UNAS-Net'e kıyasla sırasıyla 9.15 ve 3.03 kat daha az karmaşık modeller üretmiştir. DE, Attention-UNAS-Net için ODE'den daha az karmaşık bir model oluştururken, ODE daha iyi segmentasyon sonuçları vermiştir.

Önerilen NAS yöntemleri, Kocak vd. (2021) tarafından önerilen URes-Net mimarisi ile de karşılaştırılmıştır. URes-Net, tüm ölçütler için U-Net ve Attention-UNet'in segmentasyon performansını iyileştirmiştir (U-Net için VOI ve Attention-UNet; U-Net için SP, SB, PRC hariç). URes-Net, U-Net ile aynı sayıda ancak Attention-UNet'ten 1.21 kat daha az kayan noktalı işlem gerçekleştirir. URes-Net, genel olarak UNAS-Net ve Attention-UNAS-Net tabanlı yöntemlerden daha iyi segmentasyon performansı göstermesine rağmen, hesaplama karmaşıklığı açısından 7 kata kadar daha karmaşıktır.

Ayrıca tüm yöntemler Şekil 2.11’te görsel olarak karşılaştırılmıştır. Gerçek referans görüntüsü (ground truth) ve model tarafından elde edilen çıktıdaki eşleşen piksel sayısı (yani gerçek pozitif piksellerin sayısı – true positive), medikal segmentasyon çalışmalarının performansını değerlendirmek için tek başına yeterli değildir. Ekstra olarak önerilen model, gerçek referans görüntüsünde damar olmayan alanları da işaretlememelidir. Bu tür bir hataya “*Yanlış Pozitif (False Positive - FP) Hata*” denir. Son olarak, gerçek referans görüntüsünde damarların olduğu alanlar çıktıda damar var olarak işaretlenmemiş olabilir. Buna Yanlış Negatif (False Negative - FN) Hata denir. Bu tanımlamalara göre ağın elde ettiği çıktıda TP piksellerinin daha fazla, FP ve FN piksellerinin ise mümkün olduğunca az olması beklenmektedir. Şekil 2.11, UNAS-Net modellerinin TP, FP ve FN pikselleri açısından temel U-Net ağına göre daha doğru segmentasyon gerçekleştirdiğini göstermektedir. Ancak görünen o ki UNAS-Net ince damarların segmentasyonunda kalın damarlar kadar başarılı değildir. Attention-UNAS-Net modelleri, Şekil 5’te temel alınan Attention-UNet ile karşılaştırılmıştır. Benzer şekilde Attention-UNAS-Net modelleri tarafından segmente edilen görüntüler, Attention-UNet’e göre daha fazla TP piksel sayısına ve daha az FP ve FN piksel sayısına sahiptir.

2.2.5.1. Ablasyon Çalışmaları

Bu bölümde, en iyi UNAS- Net ODE (U-ODE) ve Attention-UNAS-Net (Att-ODE) ağları (Tablo 2.5’te gösterilmiştir) için ablasyon çalışmaları gerçekleştirilmiştir. Kayıp fonksiyonunun, n_c (hücre sayısı) parametresinin ve eşik değerinin segmentasyon performansı üzerindeki etkisi incelenmiştir. Modeller, eşik değeri karşılaştırması hariç, 200 eğitim dönemi boyunca tüm eğitim görüntüleriyle eğitilmiştir. Ardından, tüm ablasyon çalışması deneyleri için uzun vadeli değerlendirme adımlarında olduğu gibi test setindeki tüm görüntülerle modeller test edilmiştir. Sonuç tabloları, en iyi modellerin (bkz. Tablo 2.5) beş farklı tohum değeri ile eğitilmesi ve tüm test görüntüleri üzerinde test edilmesi sonucunda elde edilen beş performans metriğinin ortalama değerlerini göstermektedir.



Şekil 2.11: Farklı medikal görüntü segmentasyonu modellerinin görsel olarak karşılaştırılması. İlk satır, her yöntem için hata haritalarını (error maps) (Beyaz: Gerçek pozitif pikseller (TP); Kırmızı: Yanlış negatif pikseller (FN); Mavi: Yanlış pozitif pikseller (FP))

2.2.5.1.1. Kayıp Fonksiyonu

Bu bölümde, segmentasyon performansı üzerinde doğrudan bir etkisi olan kayıp fonksiyonunun etkisi araştırılmıştır. En iyi kayıp fonksiyonunu belirlemek için UNAS-Net ODE ve Attention-UNAS-Net ODE'nin performansları farklı kayıp fonksiyonları ile test setleri üzerinde değerlendirilmiştir: Dice, Jaccard, İkili Çapraz Entropi (BCE), Dice + BCE ve Dice + BCE + Jaccard (Taha vd., 2015; Wentao vd., 2022). Tablo 2.7 hem UNAS-Net ODE hem de Attention-UNAS-Net ODE için farklı kayıp fonksiyonlarının sonuçlarını sunmaktadır. UNAS-Net ODE için, Dice + BCE + Jaccard DICE, IoU ve ACC açısından en iyi sonuçları verirken, BCE SP açısından en iyi sonuçları vermektedir. Ayrıca, Dice kayıp fonksiyonu SNS açısından en iyi sonuçları vermektedir. BCE hariç tüm kayıp fonksiyonları DICE, IoU, SP ve ACC değerleri açısından temel kayıp fonksiyonlarının (Dice kaybı) performansını iyileştirmektedir. Kayıp fonksiyonunun seçimi minimum 0.35 ve maksimum 0.94

puan iyileştirme sağlamaktadır. Öte yandan, Attention-UNAS-Net ODE için Dice + BCE + Jaccard, DICE ve IoU açısından en iyi sonuçları verirken, BCE kayıp fonksiyonu SP ve ACC açısından en iyi sonuçları vermektedir. Uygun kayıp fonksiyonunun seçilmesi segmentasyon performansını 0.17 ile 0.62 puan arasında artırabilir. Sonuç olarak, kayıp fonksiyonunun seçimi segmentasyon performansını olumlu yönde etkilemektedir. Tek bir kayıp fonksiyonu yerine birleşik bir kayıp kullanılması en iyi sonuçları vermektedir.

Tablo 2.7: UNAS-Net ODE ve Attention-UNAS-Net ODE için farklı kayıp fonksiyonlarının performans karşılaştırması

Kayıp Fonksiyonu	Performans Ölçütleri				
	UNAS-Net ODE				
	DICE	IoU	SNS	SP	ACC
Dice	79.57	66.74	81.48	97.81	96.19
Jaccard	79.87	67.31	78.19	98.46	96.49
BCE	77.94	64.82	76.14	98.54	96.52
Dice + BCE	79.66	67.03	77.92	98.35	96.40
Dice + BCE + Jaccard	80.13	67.68	78.60	98.38	96.54
Attention-UNAS-Net ODE					
Dice	79.36	66.49	78.56	97.90	96.04
Jaccard	78.79	66.13	76.90	98.36	96.41
BCE	78.13	65.14	76.20	98.52	96.60
Dice + BCE	78.45	65.63	75.75	98.48	96.35
Dice + BCE + Jaccard	79.53	66.91	77.05	98.47	96.48

2.2.5.1.2. Model Budama

Ağ parametreleri ve kayan nokta işlemlerinin (FLOP) sayısı, model performansının ve çıkarım süresinin belirlenmesinde önemli bir rol oynamaktadır. Tablo 2.6, önerilen arama uzayı tarafından üretilen ağların temel ağlara kıyasla daha az karmaşık yapılara sahip olduğunu göstermektedir. Ancak, üretilen ağların karmaşıklığını doğrudan etkileyen n_c hiper parametresinin segmentasyon performansı ve karmaşıklık üzerindeki etkisi ayrıca incelenmelidir. Bu ablasyon çalışmasında, hücre sayısı hiper parametresi değiştirilerek ağın karmaşıklığı kademeli olarak artırılmıştır. Tablo 2.8'de sunulan sonuçlar, kayan nokta işlem sayısının UNAS-Net ODE için $n_c = 2$ 'den $n_c = 5$ 'e kadar %41 ile %124 arasında değiştiğini ve Attention-UNAS-Net ODE için %46 ile %138 arasında değiştiğini göstermektedir. Ancak segmentasyon performansının da düştüğü gözlemlenmektedir.

Attention-UNAS-Net gibi karmaşık ağlar için n_c seçimi sonuçlar üzerinde daha küçük bir etkiye sahipken, UNAS-Net gibi daha az karmaşık ağlar için n_c seçimi daha büyük bir etkiye sahiptir. Ayrıca, DICE, IoU, SNS metrikleri n_c değerinden SP ve ACC metriklerine göre daha fazla etkilenmektedir. Sonuçlar, hücre sayısı ile (n_c) model karmaşıklığı arasında doğrudan bir ilişkisi olmasına rağmen, n_c ile segmentasyon performansı arasında net bir korelasyon olmadığını göstermektedir. Aslında, hücre sayısı değerinin artırılması ($n_c = 5$) sonuçlarda bir düşüşe yol açmaktadır. Bu nedenle, Tablo 2.8'de görüldüğü gibi, kodlayıcı ve kod çözücü hücre içindeki ağ yapısını optimize etmekten daha fazlası gereklidir. Aynı zamanda n_c (mikro ve makro arama uzayları) değerlerinin eş zamanlı optimizasyonu da performansı olumlu yönde etkilemektedir. Ayrıca, Tablo 2.5 ve Tablo 2.8 önerilen DE tabanlı sezgiseller aracılığıyla seçilen $n_c = 4$ değeri ile en iyi sonucun elde edildiğini ortaya koymaktadır. Bu da önerilen yöntemlerin etkinliğini göstermektedir.

Tablo 2.8: Farklı hücre sayısı değerleri için performans karşılaştırması

Performans Ölçütleri						
UNAS-Net ODE						
#Cells	FLOPs (G)	DICE	IoU	SNS	SP	ACC
2	46	65.89	51.67	60.10	98.26	94.01
3	65	73.03	60.42	70.36	97.94	95.22
4	84	79.57	66.74	81.48	97.81	96.19
5	103	72.68	58.94	70.16	98.12	95.31
Attention-UNAS-Net ODE						
2	91	74.67	61.11	75.70	97.29	95.23
3	133	73.67	60.01	72.97	97.48	95.02
4	175	79.36	66.49	78.56	97.90	96.04
5	217	74.59	61.16	76.15	97.24	95.12

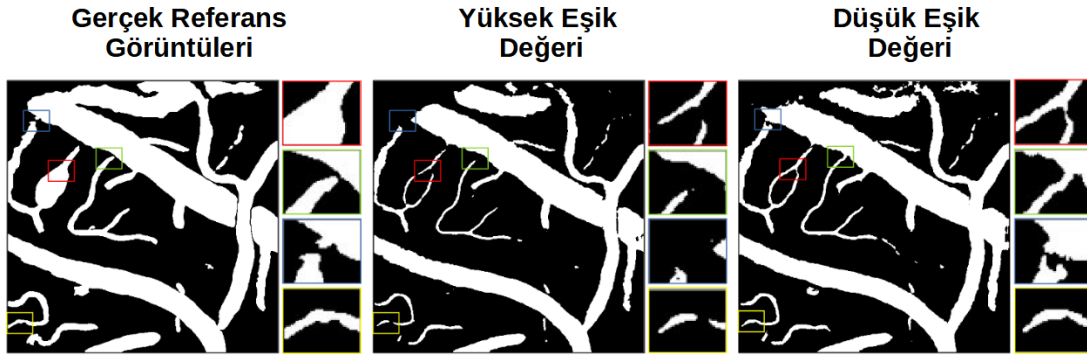
2.2.5.1.3. Eşik Değeri Kıyaslaması

Araştırmalar, eşik değerinin doğru seçilmesinin hem ince hem de kalın damarların tespitinde önemli bir etkiye sahip olduğunu göstermektedir (Wentao vd., 2022). Segmente edilmiş görüntüler oluşturmak için, ağın çıktısı olarak verilen olasılık haritalarına ($\in (0, 1)$) belirli bir eşik değeri uygulanmalıdır. Ancak yüksek eşik değerlerinin seçilmesi bu görüntülerde ince damar bilgisinin kaybolmasına neden olmaktadır. Bu nedenle, eşik değeri ne ince damarlar ne de kalın damarlar hakkındaki bilgilerin kaybolmaması için dikkatli seçilmelidir. Ablasyon

çalışmasında, yüksek ve düşük eşik olarak adlandırılan iki sabit eşik değerinin yanı sıra eşik değerini giriş görüntüsüne göre dinamik olarak hesaplayan Otsu eşik yöntemi de kullanılmaktadır. Yüksek eşik değeri, 0.5 olarak ayarlanmıştır. Öte yandan, düşük eşik değeri Wentao vd., (2022) tarafından yapılan çalışma referans alınarak 0.3 olarak ayarlanmıştır. Farklı eşik ayarları için gerçekleştirilen performans karşılaştırmalarının sonuçları Tablo 2.9'da verilmiştir. Sonuçlar, UNAS-Net ODE için düşük eşik değeri seçilmesinin SNS hariç tüm metriklerde iyileşmeye yol açtığını göstermektedir. Attention-UNAS-Net ODE için, varsayılan eşik değeri olan 0.5 ile elde edilenlerden daha iyi sonuç elde edilemediği görülmektedir. Ayrıca, Şekil 2.12'de gösterildiği gibi farklı eşik değerlerinin etkileri görselleştirilmiştir. Sonuçlar, düşük bir eşik değeri kullanmanın gerçek referans görüntüsüne benzer bir çıktı verdiğini göstermektedir. Buna karşılık, yüksek bir eşik değeri hassas damar bilgilerinin kaybına neden olmaktadır. Bununla birlikte, düşük bir eşik değerinin kullanılması, Şekil 2.12'deki yeşil dikdörtgende gösterildiği gibi damarların yanlış ayrılmasına yol açabilir. Sonuç olarak, eşik değerinin segmentasyon performansını etkilediği ve Wentao vd., (2022) tarafından önerildiği gibi çift eşik değeri ile segmentasyon sonuçlarının doğruluğunun artırabileceği görülmüştür.

Tablo 2.9: Eşik değeri karşılaştırmaları

Eşik Değeri	Performans Ölçütleri				
	UNAS – Net ODE				
	DICE	IOU	SNS	SP	ACC
0.3	80.01	67.43	81.29	97.84	96.20
0.5	79.57	66.74	81.48	97.81	96.19
Otsu	80.01	67.42	81.48	97.81	96.19
Attention-UNAS-Net ODE					
0.3	79.16	66.23	78.40	97.92	96.04
0.5	79.36	66.49	78.56	97.90	96.04
Otsu	79.19	66.27	78.56	97.90	96.04



Şekil 2.12: vesselNN veri seti üzerinde düşük ve yüksek eşik değeri analizi sonuçlarının grafiksel gösterimi. Ayrıntıları daha iyi görüntülemek için görüntülerde dört adet dikdörtgen kullanıldı

2.2.5.1.4. Genelleştirme Performansı

Derin öğrenme modellerinin yeteneklerini belirlemek için yeni görüntüler ve görevler üzerindeki performanslarını değerlendirmek çok önemlidir. Bu durum özellikle tıbbi görüntüler için önemlidir çünkü bu görüntüler büyük farklılıklar gösterebilir ve bir veri kümesinde iyi çalışan bir model yeni verilerde aynı performansı gösteremeyebilir. Yapılan çalışmada, ODE (UNAS-Net ODE ve Attention-UNAS-Net ODE) kullanılarak aranan en iyi modellerin KUVESG veri kümesi üzerindeki genelleme performansı değerlendirilmiştir. Bu modeller, iki ekstra veri kümesi üzerinde test edilmiştir: DRIVE (Staal vd., 2004) ve Optofil (Kus vd., 2022). NAS adımları bu iki ekstra veri kümesi için tekrar gerçekleştirilmemiştir. Aranan modeller (ODE kullanılarak aranan en iyi modeller) bu iki veri kümesi üzerinde eğitilir ve test kümesi üzerinde değerlendirilir. Ayrıca bu prosedür ince ayar olarak adlandırılır. Optofil veri kümesi, gri ölçekli formatta temsil edilen toplam 4914 hücre kültürü görüntüsü içerir ve DRIVE veri kümesi 40 renkli retinal kan damarı görüntüsünden oluşur. İlk olarak, aranan modeller Optofil eğitim seti üzerinde eğitilmiş ve Kus vd., (2022) tarafından açıklanan değerlendirme süreci izlenerek test seti üzerinde değerlendirilmiştir. Ancak, adil bir karşılaştırma sağlamak amacıyla Kus vd., (2022) tarafından bulunan en iyi model kullanılarak tüm ölçümler yeniden hesaplanmıştır. Ortalama sonuçlar, Tablo 2.10'da sunulmuştur. Attention-UNAS-Net + ODE, ACC, SP ve F1 ölçümleri için en iyi performansı gösterirken, SNS ve IoU için küçük bir (1.31 kat) karmaşıklık artışıyla ikinci en iyi sonuçları vermektedir. Ayrıca UNAS-Net, SNS açısından 1.58 kat daha az FLOPs ile diğer

yöntemlerden daha iyi performans göstermekte ve F1 hariç diğer ölçütler için oldukça rekabetçi sonuçlar vermektedir.

Tablo 2.10: Önerilen yöntemlerin Optofil veri kümesi üzerindeki performans sonuçları. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir. U-ODE: UNAS-Net ODE; Att-ODE: Attention-UNAS-Net ODE

Yöntem	FLOPS (G)	ACC	SNS	SP	F1	IOU
Kus vd., (2022)	4.18	98.99	78.06	99.74	82.97	72.67
U - ODE	2.64	98.78	83.04	99.40	79.93	70.15
Att - ODE	5.48	99.10	80.90	99.75	84.60	71.13

Ayrıca aranan modeller, model karmaşıklığı ve segmentasyon performansı açısından DRIVE veri kümesi üzerinde gerçekleştirilen NAS çalışmalarıyla karşılaştırılmaktadır. Aranan modeller DRIVE eğitim seti üzerinde eğitilip, test seti üzerinde değerlendirilmiştir. Yapılan çalışmada, Wentao vd., (2022) tarafından uygulanan ön-işleme adımları kullanılmıştır. Literatürde DRIVE için yapılmış az sayıda NAS araştırması bulunmaktadır. Popat vd., (2020) ve HNAS (Houreh vd., 2021) araştırmalarında farklı parametre boyutlarına sahip çeşitli modeller kullanmıştır. Sırasıyla en az ve en çok parametreye sahip olan v1 ve v2 modellerinden ikisi karşılaştırma için seçilmiştir. Ayrıca, Genetik U-Net (Wei vd., 2022), biri ikili görüş alanı olan (v1) ve diğeri olmayan (v2) iki farklı sonuç sunmaktadır. Elde edilen sonuçlar, ilgili makalelerde raporlanan sonuçlar ile karşılaştırılmıştır (Popat vd., 2020; Houreh vd., 2021; Wei vd., 2022; Rajesh vd., 2023). Sonuçlar, Tablo 2.11’de gösterilmiştir. Attention-UNAS-Net ODE tüm metriklerde (F1 hariç) en iyi sonuçları vermektedir. Ayrıca, sırasıyla 1.62 ve 15.21 kat daha az parametreye sahip olmasına rağmen tüm ölçütler için HNAS v1 ve BTU-Net’ten daha iyi performans göstermektedir. UNAS-Net ODE modeli ACC ve SP için ikinci en iyi sonuçları vermektedir. Ayrıca, ACC ve F1 için BTU-Net’e göre 2.54 kat daha az parametre ile daha iyi sonuçlar elde etmektedir ve daha az parametre ile ACC, SNS ve F1 (Genetik U-Net hariç) açısından tüm modellerden daha iyi performans göstermektedir. Sonuç olarak, aranan modeller ilgili veri kümeleri üzerinde tekrar NAS gerçekleştirilmeden bile yeni hücre segmentasyonu ve kan damarı segmentasyonu görevlerine başarılı bir şekilde genelleştirilebilir.

Tablo 2.11: DRIVE veri kümesi üzerinde farklı NAS çalışmalarının performans sonuçları. Kırmızı ve mavi renkler her bir performans ölçütü için en iyi iki değeri göstermektedir. U-ODE: UNAS-Net ODE; Att-ODE: Dikkat-UNAS-Net ODE

Yöntem	# Parametre (M)	ACC	SNS	SP	F1
Popat et al . v1	0.18	93.56	59.67	98.50	-
Popat et al . v2	8.10	95.34	75.01	98.31	-
HNAS v1	157	95.42	77.07	98.10	81.08
HNAS v2	2.90	95.46	77.44	98.09	81.29
Genetic U - Net v1	0.27	95.77	83.00	97.58	83.14
Genetic U - Net v2	0.27	97.07	83.00	98.43	83.14
BTU - Net	16.80	96.89	-	98.44	81.78
U - ODE	6.60	97.00	83.74	98.31	82.04
Att - ODE	10.32	97.08	89.71	98.95	82.03

ÜÇÜNCÜ BÖLÜM

3. MedUNAS

3.1. MedUNAS

3.1.1. Giriş

Retinal kan damarı segmentasyonu, retina içerisindeki fundus görüntülerinden kan damarlarının çıkarılmasını içeren tıbbi görüntü analizinde önemli bir görevdir. Bu işlem, diyabetik retinopati ve yaşa bağlı makula dejenerasyonu da dahil olmak üzere retinal vasküler hastalıkların teşhisi ve izlenmesi için kritik öneme sahiptir (Abramoff vd., 2010). Retinal kan damarlarının doğru segmentasyonu, klinisyenlere damar morfolojisi ve kıvrımlılığı gibi değerli bilgiler sağlayarak bu hastalıkların tespit edilmesine ve yönetilmesine yardımcı olur (Li vd., 2021). Öneme rağmen, retinal kan damarlarının karmaşık yapısı ve değişken görünümü nedeniyle retinal damar segmentasyonu zordur (Wang vd., 2019). Bu nedenle, retinal damar segmentasyonu için gürbüz ve verimli algoritmalar geliştirmek, tıbbi görüntülemeye aktif bir araştırma konusu haline gelmiştir.

Fundus görüntülerinde retinal damarların otomatik olarak bölütlenmesi, damar bölgesi ile arka planın ayırt edilmesi, damar ağaçlarının karmaşık yapısı, diğer patolojik sorunlar ve düzensiz aydınlatma gibi çeşitli nedenlerden dolayı zordur (Wei vd., 2022). Bu zorlukları diğer geleneksel yöntemlerden çok daha etkili bir şekilde ele aldığı için evrimsel sinir ağı (CNN) tabanlı yöntemler, retinal damar segmentasyonu için tercih edilen modeller haline gelmiştir (Wentao vd., 2022; Qu vd., 2023). Ayrıca yapılan çalışmalar, biyomedikal görüntü segmentasyonu için önerilen özel ESA modelleri olan U-Net ve U-şekilli ağların retinal damar segmentasyonu çalışmaları için en iyi sonuçları verdiğini göstermiştir (Ronneberger vd., 2015; Oktay vd., 2018; Wu vd., 2019). U-Net ve U-şekilli ağlar temel olarak kabul edilmesine rağmen, bazı dezavantajları vardır. Bu ağlar elle tasarlanmış mimarilerdir ve ince damar görüntüleri gibi nispeten yoğun karmaşık yapılara sahip

fundus görüntüleri için ince damarları yeterince iyi tespit edemezler. Ayrıca, U-şekilli ağlar manuel olarak belirlenmesi gereken birçok hiper parametreye sahiptir. Bu parametreler segmentasyon performansını ve ağın karmaşıklığını doğrudan etkilemektedir. Bu hiper parametrelere ek olarak, bu ağlar özellik çıkarma ve görüntüyü yeniden oluşturma için benzer tipte yığılmış bloklar kullanır. Ancak bu yapılar karmaşık vasküler yapıların tespitinde ve özellik çıkarımında başarılı olamamaktadır. Bu nedenle, daha az parametre ile karmaşık damar desenlerini çıkarmak için daha esnek yapılara ihtiyaç duyulmaktadır.

Derin öğrenme yöntemleri, özellikle U-şekilli model, retinal damar segmentasyonunda büyük başarı göstermiştir (Chen vd., 2021). Öte yandan, ele alınan görev için bazı yeni NAS çalışmaları literatürde sunulmuştur. İlerleyen paragraflarda, retinal damar segmentasyonu için son zamanlarda yayınlanan derin öğrenme yöntemleri ve NAS çalışmalarına genel bir bakış sunulmuştur.

Zhou vd. (2021), U-Net modelinin gürültülü etiketlere karşı gürbüzlüğünü artırmak için “Çalışma Grubu Öğrenimi (SGL)” olarak isimlendirilen bir yöntem sunmaktadır. Deneysel sonuçlar, SGL'nin özellikle eğitim etiketlerinin gürültülü olduğu durumlarda, herkese açık iki veri kümesinde segmentasyon performansını önemli ölçüde artırdığını göstermektedir.

Kamran vd. (2021), damar bilgilerini çıkarmak için iki üreteç (generator) ve iki çok ölçekli otomatik kodlama ayırıcı (multi-scale autoencoding discriminators) kullanan üretken düşman ağ tabanlı bir yöntem olan RV-GAN'ı sunmaktadır. RV-GAN'ı DRIVE, STARE ve CHASE_DB1 veri kümeleri üzerinde değerlendirmişler ve Mean-IOU ve SSIM ölçütleri açısından diğer mimarilerden daha iyi sonuçlar vermişlerdir.

Wentao vd. (2022) tarafından yapılan çalışmada, tıbbi görüntülerdeki kan damarlarını etkili bir şekilde segmente edebilen, ince damarlar ve düşük kontrast gibi zorlukların üstesinden gelen FR-UNet adlı yeni bir teknik sunulmaktadır. Bu teknik, görüntü çözünürlüğünü korurken ağı hem yatay hem de dikey yönde genişletmek için çok çözünürlüklü bir evrişim mekanizması içerir. Doğru piksel düzeyinde temsiller elde etmek için bir özellik toplama modülüne ve değiştirilmiş artık bloklara sahiptir. Yöntem, retinal damar ve koroner anjiyografi veri kümeleri üzerinde

değerlendirilmiş ve hassasiyeti önemli ölçüde artırırken daha az parametre ile mevcut yaklaşımlardan daha iyi performans göstermiştir.

Qu vd. (2023) tarafından TP-Net adı verilen ve bir ana yol, alt yol ve çok ölçekli özellik birleştirme modülü içeren iki yollu bir ağ önerilmiştir. Önerilen yöntem, performans ve genelleme yeteneği açısından mevcut yöntemlerden daha iyi performans göstermiş ve daha az model parametresi kullanılmıştır.

Öte yandan, retinal damar segmentasyonu için en uygun U-şekilli ağ mimarilerini aramak için meta-sezgisel yöntemler, özellikle de evrimsel algoritmalar kullanılmaktadır. Retinal damar için NAS kullanan ilk çalışma olduğu belirtilen Fan vd. (2020)'nin önerdiği belirli bir arama uzayına sahip evrimsel bir sinir mimari arama yöntemi sunulmuştur. Yöntemin performansı DRIVE, STARE ve CHASE_DB1 olmak üzere üç veri kümesi üzerinde değerlendirilmiştir. Sonuçlara göre, önerilen yöntem daha az parametre ile daha iyi segmentasyon performansı sağlamıştır.

Retinal kan damarı segmentasyonu için U-net mimarilerini otomatik olarak aramak için genetik algoritmayı kullanan ve “Hiper Sinirsel Mimari Arama (HNAS)” olarak adlandırılan başka bir yöntem Houreh vd. (2021) tarafından sunulmuştur. Yazarlar karşılaştırma için DRIVE veri setini kullanmış ve iki deney seti uygulanmıştır: U-net mimarilerinin manuel ve otomatik tasarımı. Önerilen yaklaşım, bir ConvLSTM hücresi, artık bağlantılar ve yığın normalizasyonunu seçmek için ikili bir strateji ekleyerek temel alınan U-net'in performansını artırmıştır. Ayrıca, deneyler HNAS'ın daha küçük boyutlarda, son teknoloji yöntemlere benzer doğrulukta ve aşırı uyum olmadan rekabetçi mimariler bulabildiğini göstermektedir.

Wei ve diğerleri ise (Wei vd., 2022) retinal kan damarlarını bölütlemek için U şeklinde bir CNN oluşturan Genetik U-Net adlı yeni bir sinir mimari arama yöntemi önermiştir. Bu teknik, segmentasyon doğruluğunu artırırken mimariyle ilgili parametrelerin sayısını azaltmıştır. Önerilen yöntem, geliştirilmiş bir genetik algoritma ve daha küçük ancak esnek bir arama uzayı kullanarak daha yüksek performanslı yapıları tanımlamıştır. Sonuçlar, Genetik U-Net ile oluşturulan mimarilerin, önemli ölçüde daha az parametre ile diğer son teknoloji modellerden daha iyi performans gösterdiğini göstermektedir.

Evrimsel algoritmaların yanı sıra, İkili Öğretme-Öğrenme Tabanlı Optimizasyon (BTLBO -Rajesh vd., 2023) yöntemi de arama algoritması olarak kullanılmaktadır. Bu çalışmada yazarlar, evrimsel adımları hızlandırmak için esnek bir arama uzayı, bir dikkat mekanizması ve bir önbellek sistemi kullanarak U şeklindeki mimariyi otomatik olarak optimize eden BTLBO evrimsel modelini (BTU-Net) sunmaktadır. BTU-Net, arama yeteneğinin gösterilmesi için herkese açık retinal damar segmentasyonu veri kümeleri üzerinde test edilmiştir. Sonuçlar, önerilen yöntemin son teknoloji modellere göre üstün performans gösterdiğini ortaya koymuştur.

Yapılan tez çalışmasında ilk olarak, U-şekilli ağlar için yeni bir sinir mimari arama yöntemi (Neural Architecture Search - NAS) olan MedUNAS önerilmiştir. Ayrıca, Dong vd. (2022) ve Kus vd. (2022) tarafından sunulan arama uzayına dayanan yeni bir hücre tabanlı arama uzayı önerilmektedir. En iyi ağ yapısını aramak için bu arama uzayı üzerinde “Karşıtlık Temelli Diferansiyel Gelişim (Opposition Based Differential Evolution - ODE)” ve “Genetik Algoritma (GA)” yöntemleri uygulanmıştır. Retinal damar segmentasyonu için Dong vd. (2022) tarafından önerilen GA ve ayrık kodlama stratejisi bu çalışmaya uyarlanmış ve ayrıca kodlama stratejisi ODE kullanılarak sürekli arama uzayı için de genişletilmiştir. Dong vd. (2022) tarafından sunulan ayrık kodlamadan farklı olarak, karmaşıklığı azaltmak için hücre yapısı değiştirilmiş ve yerel arama işlemi çalışmaya dahil edilmemiştir. Öte yandan, NAS alanında yapılan araştırmalar sürekli arama uzayı tabanlı çalışmaların daha iyi sonuçlar verdiğini göstermektedir (Awad vd., 2020; Kus vd., 2022). Bu nedenle, önerilen kodlama şeması sürekli arama uzayı üzerinde çalışacak şekilde değiştirilmiş, ayrık ve sürekli kodlama stratejilerini karşılaştırmak için ODE kullanılmıştır. Bilinen kadarıyla yapılan çalışma, retinal damar segmentasyonu problemi için ODE uygulayan ilk NAS çalışmasıdır. Önerilen yöntemler DRIVE, CHASE_DB1, DCA1 ve CHUAC olmak üzere herkese açık olan dört adet retinal damar segmentasyonu veri kümesi üzerinde değerlendirilmiştir. Sonuçlar, segmentasyon performansı ve model karmaşıklığı açısından son teknoloji yöntemler ve sinir mimarisi arama çalışmaları ile karşılaştırılmıştır. Deneysel sonuçlara göre, önerilen yöntemler son teknoloji çalışmalara göre en az karmaşık modelleri (CHASE_DB1 hariç) üretmekte ve yine de oldukça rekabetçi sonuçlar elde

etmektedir. Ayrıca, kayıp fonksiyonlarının etkisini değerlendirmek için ablasyon çalışmaları gerçekleştirilmiş ve modelin farklı alanlar için genelleştirme performansı ve çapraz veri kümelerindeki performansı incelenmiştir.

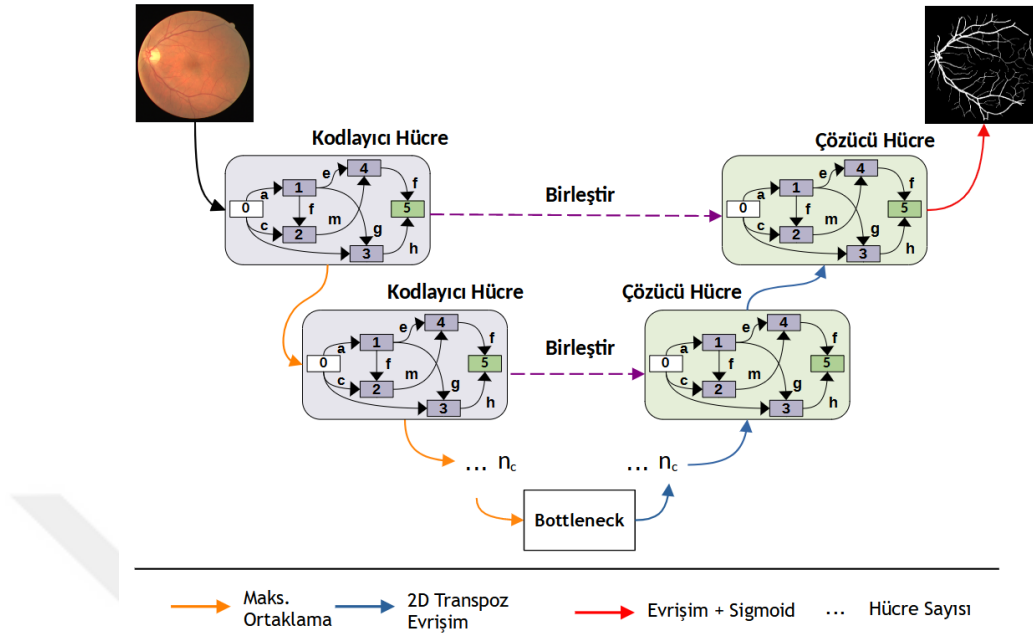
Ayrıca, hücre şeklindeki ağlar için farklı kodlama stratejilerinin performansı retinal kan damar segmentasyonu veri kümeleri üzerinde karşılaştırılmıştır. Farklı hücre yapılarının arama uzayında araştırılması için ODE yöntemi kullanılmıştır. MedUNAS ve UNAS-Net (Kus vd., 2022) yöntemlerinde kullanılan kodlama yöntemleri karşılaştırıldığında UNAS-Net çalışmasında önerilen yöntemin en az %50 daha az parametre ile rekabetçi sonuçlar elde ettiği gözlemlenmiştir.

3.1.2. Yöntem

MedUNAS, retinal damar segmentasyonu için en uygun U-Net mimarisini araştırmak üzere önerilen evrimsel bir NAS yöntemidir. MedUNAS, hücre tabanlı kodlayıcı ve kod çözücü hücrelere sahip U-şeklinde bir mimaridir. Bu mimari, hücrenin yapısını belirleyen hücre tabanlı mikro arama uzayını ve toplam hücre sayısını ve ilk hücredeki özellik haritalarının sayısını belirleyen makro arama uzayını içerir. Ayrıca, MedUNAS'ta genetik algoritma ve karşıtlık tabanlı diferansiyel algoritma kullanılmıştır. Aşağıdaki bölümler, arama uzayı, kodlama şeması ve arama yöntemi de dahil olmak üzere MedUNAS hakkında derinlemesine bilgi vermektedir.

3.1.2.1. Arama Uzayı

MedUNAS, sırasıyla aşağı örnekleme ve yukarı örnekleme yapan kodlayıcı ve kod çözücü hücrelerden oluşur. Kodlayıcı hücreler evrişim işlemleri yoluyla özellikleri çıkarır ve maksimum havuzlama kullanarak görüntü boyutunu azaltırken, kod çözücü hücreler çıkarılan özellikleri kullanarak görüntüyü yeniden yapılandırır ve transpoze evrişim yoluyla görüntü boyutunu artırır. Son olarak, segmente edilmiş görüntüyü oluşturmak için bir 2D evrişim ve sigmoid fonksiyonu uygulanır. MedUNAS'ın genel mimarisi Şekil 3.1'de gösterilmiştir.



Şekil 3.1: MedUNAS genel yapısı

Yapılan çalışmada, yönlendirilmiş döngü içermeyen çizgeler (DAG) şeklinde temsil edilen kodlayıcı ve kod çözücü hücrelerin yapısını tanımlamak için hücre tabanlı mikro arama yöntemi kullanılmıştır. Hücre tabanlı mikro-arama yöntemi için Dong vd. (2022) tarafından sunulan yöntem temel alınmıştır; burada kenarlar işlemleri, düğümler ise farklı işlemlerden elde edilen çıktıları göstermektedir. Bu çalışmada, her hücre 2 girdi düğümü, 4 ara düğüm ve bir çıktı düğümü içermektedir. Her ara düğüm kendisinden önceki düğümlerden iki bilgi akışı alır, bilgiyi işler ve kendisinden sonraki düğümlere ve çıktı düğümüne gönderir. Yapılan çalışmada ise, Dong vd. (2022) tarafından önerilen arama uzayının aksine, her hücrenin yalnızca bir giriş düğümü vardır ve tüm ara düğümlerin bilgi akışlarını çıkış düğümüne göndermesi için herhangi bir kısıtlama yoktur. Bir hücrenin gösterimi Şekil 3.2'de verilmiştir. Her hücre bir dizi düğüm (N), kenar (E) ve işlemden (O) oluşur. Bu hücreler bir girdi, bir çıktı ve farklı işlemlerden elde edilen çıktıları toplama işlemi ile birleştiren 4 ara düğüme ($|N| = 4$) sahiptir. Her ara düğüm ($n \in N \setminus \{n_1\}$) önceki düğümlerden gelebilecek 2 iç kenara (in-edge) sahiptir. Hücre yapısında da gösterilen iki farklı türde kenar vardır: Şekil 3.2'de düz kenarlarla gösterilen sabit kenarlar ve olası kenarlar (Şekil 3.2'de kesikli çizgiler). Bu kenar türleri kenarlar kümesini (E) oluşturur ve her kenar ($e \in E$) bir ara düğüme uygulanan bir işlemi temsil eder. Uygulanacak işlemler bir işlemler kümesinden (O) seçilir. Her kenar için

seçilebilecek 15 olası işlem Tablo 3.1'de listelenmiştir. İlk 14 olası işlem Liu vd. (2018); Kus vd., (2022); Kus vd. (2023) çalışmalarında önerilirken, sonuncusu (o) 2 iç kenar kısıtlamasını karşılarken karmaşıklığı azaltmak için kullanılır.

Şekil 3.2: Hücre tabanlı mikro arama alanının bir özeti ve genel kodlayıcı ve kod çözücü hücrelerin mimarisi.

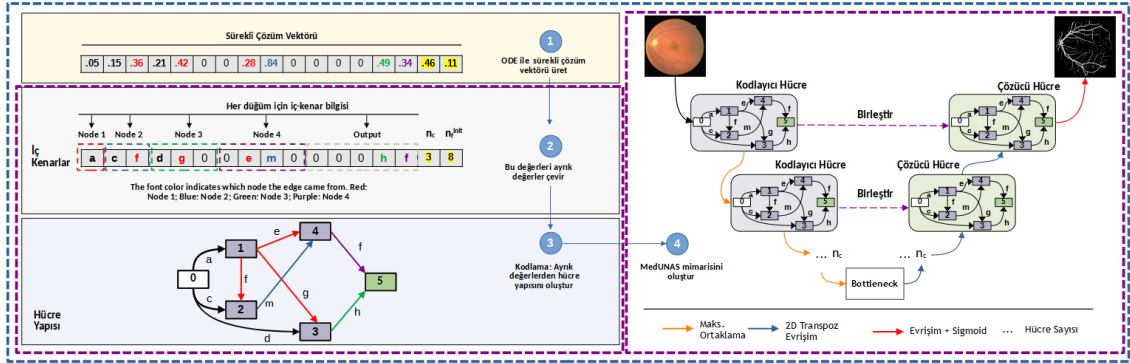
Operasyon	Operasyon Adı
a	1×1 2B Evrişim
b	3×3 2B Evrişim
c	5×5 2B Evrişim
d	7×7 2B Evrişim
e	3 ×3 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim
f	5 ×5 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim
g	7×7 2B Derinlemesine Ayrılabilir (Depthwise Separable) Evrişim
h	3×3 2B Açılan (Dilated) Evrişim
i	5×5 2B Açılan (Dilated) Evrişim
j	7×7 2B Açılan (Dilated) Evrişim
k	3×3 2B Asimetrik Evrişim
l	5×5 2B Asimetrik Evrişim
m	7×7 2B Asimetrik Evrişim
n	Atlama bağlantısı (Skip-Connection)
o	Bağlantı yok

Ayrıca, MedUNAS mimarisi iki ek hiper parametreye sahiptir: Hücre sayısı ve makro arama uzayını tanımlayan ilk hücredeki özellik haritalarının sayısı. Mimaride eşit sayıda kodlayıcı ve kod çözücü hücre vardır ve toplam hücre sayısı (N_c) kodlayıcı, kod çözücü ve darboğaz hücrelerinin sayısı toplanarak hesaplanır ($N_c = 2n_c + 1$). Kodlayıcı ve kod çözücü hücrelerin sayısı ($n_c \in \{2, 3, 4, 5\}$) bir karar değişkenidir. Makro arama uzayındaki ikinci hiper parametre olan ilk hücredeki özellik haritalarının sayısı (n_f^{init}) ağıın performansını etkiler ve her maksimum havuzlama işleminden sonra özellik haritalarının sayısı iki katına çıkarılır. Bu hiper-parametre ($n_f^{init} \in \{8, 16, 32\}$) ilk kodlayıcı hücrenin işlemlerinin sonunda kaç özellik haritası üretileceğini belirtir.

3.1.2.2. Kodlama Şeması

Yapılan çalışmada, genetik algoritma (GA) için Dong vd. (2022) tarafından önerilen kodlama şeması kullanıldı ve bu kodlama şeması karşıt tabanlı diferansiyel gelişim algoritması (ODE) için çalışacak şekilde değiştirildi. GA için ayrık değerli bir vektör ve ODE için 17 değişkenli gerçek değerli bir vektör ($\in [0, 1]$) aday bir ağı göstermektedir. Bu vektördeki ilk 15 değişken, her bir düğüm arasındaki akışı ve yönlü döngü içermeyen çizgedeki her bir kenar için seçilen işlemleri temsil eder. Son iki değişken ise sırasıyla kodlayıcı/kod çözücü hücre sayısını (n_c) ve ilk hücredeki özellik haritası sayısını (n_f^{init}) göstermektedir.

Belirli bir hücre yapısı için kodlama mekanizması hem GA hem de ODE için Şekil 3.3'te gösterilmiştir. İlk olarak, hücre yapısı için karşılık gelen ayrık değişkenlerle bir DAG oluşturulur. Ardından, MedUNAS mimarisi bu hücre yapısı ve iki hiper-parametre ile inşa edilir. Bu adımların yanı sıra, ODE'de aday çözümdeki sürekli değerler ayrık değişkenlere dönüştürülür.



Şekil 3.3: MedUNAS mimarisi için kodlama adımlarına genel bakış. Mor kesikli çizgi GA adımlarını, mavi kesikli çizgi ise ODE adımlarını göstermektedir. Adım 1 yalnızca ODE için gerçekleştirilir.

Şekil 3.3'te ayrıca 4 ara düğüm içeren örnek bir hücre yapısını göstermektedir. Bu yapıda, *Node 1*, *Node 0*'dan (giriş düğümü) bilgi alır ve *a* işlemini uygular (1×1 Evrişim, Tablo 3.1'e bakınız). Daha sonra, bu işlemin çıktısı *Node 1*'de saklanır. *Node 2*, *Node 0* ve *Node 1*'den gelebilecek yalnızca iki olası giriş kenarına sahiptir. Bu nedenle, *Node 0* ve *Node 1*'den bilgi alır. Daha sonra bu düğümlere sırasıyla *c* ve *f* işlemleri uygulanır. Bu işlemlerden sonra çıktılar *Node 2*'de toplama işlemi ile birleştirilir. *Node 3* düğümüne gelebilecek üç olası in-edge vardır: *Node 0*, *Node 1* ve *Node 2*. Bu adımda, iki düğüm rastgele seçilir. Şekil 3.3'te gösterildiği gibi, *Node 3*, *Node 0* ve *Node 1*'den bilgi alır, ardından bu düğümlere sırasıyla *d* ve *g* işlemlerini uygular. Benzer şekilde, bu işlemlerin çıktıları *Node 3*'te birleştirilir. Aynı adımlar *Node 4* ve *Node 5* için de gerçekleştirilir.

Dong vd. (2022) tarafından yapılan çalışmada yazarlar, yalnızca sinir düğümü bağlantılarını temsil eden ve yaygın olarak kullanılan ikili dizi ("01") kodlaması yerine, işlem ve bağlantı bilgilerini temsil etmek için hem alfabe hem de "0" dizelerini kullanan bir çözüm kodlaması önermektedir. Bu çözüm kodlama yöntemi GA için ele alınmış ve ODE için modifiye edilmiştir. Bu çözüm kodlamasında kenarlar işlemleri, düğümler ise farklı işlemlerden elde edilen çıktıları göstermektedir. Şekil 3.3'te "a-cf-dg0-0em0-000hf" kodlaması kullanılarak oluşturulmuş örnek bir hücre yapısını göstermektedir. Şekil 3.3'te gösterildiği gibi, çözüm kodlamasının ilk biti *Node 1*'e gelen kenar-içi bilgisini ve bu kenar-içi için seçilen işlemi belirtir. Benzer şekilde, sonraki iki bit *Node 2*'ye gelen kenar içi bilgileri ve seçilen işlemleri belirtir ve bu böyle devam eder. Son olarak, son beş bit

Node 5 için kenar-içi ve seçilen operasyon bilgilerini gösterir. Örneğin, son beş bit (“000hf”) *Node 5* için beş olası kenar-içi bağlantısını (*Node 0*, *Node 1*, *Node 2*, *Node 3*, *Node 4*) ve bu olası kenar-içi bağlantılar için seçilen işlemleri gösterir. Son beş bite göre, aşağıdaki düğümler arasında hiçbir bağlantı yoktur: *Node 0* – *Node 5*; *Node 1* – *Node 5*; *Node 2* – *Node 5* (Şekil 3.3'e bakınız). Ayrıca, *Node 5*, *Node 3*'ten bilgi alır ve *h* işlemini *Node 3*'e uygular. Benzer şekilde, *Node 5*, *Node 4*'ten bilgi alır ve *Node 4* üzerinde *f* işlemini gerçekleştirir.

Dong vd. (2022) tarafından önerilen çözüm kodlaması ODE için kullanılamaz. ODE, mutasyon ve çaprazlama işlemlerini gerçekleştirmek için gerçek değerli bir vektör kullanır. Bu nedenle, çözüm kodlama adımlarına fazladan iki adım ekledik: (1) gerçek değerli bir çözüm vektörü oluşturmak ($\in [0, 1]$) ve (2) bu vektör üzerinde ayrık bir çözüm vektörüne eşleme (mapping) yapmak (Şekil 3.3'e bakınız). Bu çalışmada, Awad vd. (2020); Kus vd. (2022) çalışmalarında önerilen eşleme şeması kullanılmıştır. Bu adımlar aşağıdaki gibi özetlenebilir:

1. İlk olarak, sıfır değerlerle doldurulmuş 17 eleman içeren bir çözüm vektörü oluşturulur. İlk 15 eleman her bir olası kenar için bağlantıları ve işlemleri göstermektedir. Bu elemanlar, maksimum 2 kenar içi kısıtlaması ve her düğüm için olası bağlantı sayısı dikkate alınarak rastgele üretilen gerçek değerlerle değiştirilir.
2. Her düğüm için olası iç kenar sayısı Şekil 3.3'te (Adım 2) farklı renkli kesikli çizgilerle gösterilmiştir. Örneğin, kırmızı kesikli çizgi *Node 1* için olası iç-kenarları gösterirken, mor kesikli çizgi *Node 4* için 4 olası iç-kenarı göstermektedir. *Node 1*'in bir olası giriş kenarı vardır. Bu nedenle, yalnızca bir gerçek değerli sayı rastgele üretilir (Şekil 3.3'te 0.05). Bu iç-kenar için on beş olası işlem olduğundan, 0 ile 1 arasındaki değerler, her biri belirli bir işleme karşılık gelen on beş eşdeğer bölmeye ayrılır. İlk bölme $[0, 0.067]$ Tablo 3.1'de gösterildiği gibi “*a* : 1×1 2B Evrişim” işlemini gösterirken, ikinci bölme $[0.067, 0.133]$ “*b* : 3×3 2B Evrişim” ve üçüncü bölme $[0.133, 0.200]$ “*c* : 5×5 2B Evrişim” işlemine karşılık gelir ve bu böyle devam eder. 0.05 değeri ilk bölmedeki işleme karşılık geldiğinden *a* işlemi ilk eleman için seçilmiştir. Benzer şekilde, 7 ile 10 arasındaki elemanlar *Node 4* için olası iç-

kenarları gösterir. *Node 4*'ün dört olası iç kenarı vardır ve bunlardan ikisi rastgele seçilmelidir (iki iç kenar kısıtı). Bu nedenle, iki gerçek değerli sayı rastgele üretilir (Şekil 3.3'te 0.28 ve 0.84). Ardından, bu iki iç-kenar için sırasıyla ilgili işlemler (e ve m) seçilir.

3. Son iki eleman; n_c ve n_f^{init} karar değişkenlerini belirtir. n_c için dört değer ve n_f^{init} için üç değer vardır, bu nedenle $[0, 1]$ aralığı sırasıyla eşit büyüklükte dört ve üç bölüme bölünür. Ardından, Şekil 3.3'te gösterildiği gibi (sarı vurgulu) bu karar değişkenleri için karşılık gelen değerler seçilir.
4. İlk üç adımın bir sonucu olarak MedUNAS mimarisi oluşturulur.

3.1.2.3. Arama Yöntemleri

Yapılan çalışmada, iki farklı arama yöntemi karşılaştırılmıştır: “Karşıtlık Temelli Diferansiyel Gelişim” ve “Genetik Algoritma”. Karşıtlık Temelli Diferansiyel Gelişim yöntemi Bölüm 1.5.1’de detaylı olarak açıklanmıştır. Bu bölümde, Genetik Algoritma yönteminin MedUNAS ile nasıl kullanıldığı anlatılmaktadır.

3.1.2.3.1. Genetik Algoritma

Genetik algoritma, doğal seçim sürecinden esinlenen bir optimizasyon algoritması türüdür. Bir genetik algoritmada yer alan adımlar aşağıdaki gibidir:

(1) Başlatma: Başlangıç popülasyonu Bölüm 3.1.2.2’de açıklanan kodlama yöntemi kullanılarak rastgele oluşturulur. Başlangıç popülasyonunda her bir mimari için hücre yapısını ve geriye kalan iki hiper parametreyi temsil eden 20 birey vardır.

(2) Uygunluk Değerlendirmesi: Popülasyondaki her birey, segmentasyon performansına göre değerlendirilir. Her bir bireyi değerlendirmek için “dice score” kullanılır.

(3) Seçim: Popülasyondaki en uygun bireyler bir sonraki neslin ebeveynleri olmak üzere seçilir. Ebeveyn seçimi olarak Rulet Tekerleği (Roulette Wheel) seçimi kullanılır.

(4) Çaprazlama: Dong vd. (2022) tarafından sunulan tek tip (uniform) çaprazlama işlemi kullanılmıştır. Rulet tekerleği seçimi kullanılarak iki ebeveyn seçilir: $P1$ ve $P2$. Aday çözümde, her biri bir ara düğümün bağlantılarını ve

işlemlerini gösteren 5 blok (Şekil 3.3'te kırmızı, mavi, yeşil, mor ve açık yeşil kesikli çizgiler) bulunduğundan, tek tip çaprazlama bu bloklar üzerinde gerçekleştirilir. Her blok, iki yeni çocuk ($C1$ ve $C2$) oluşturmak için önceden tanımlanmış bir α olasılığına göre değiştirilir. Örneğin: İki ebeveyn çözüm kodlamamız olduğunu varsayalım: “a-cf-dg0-0em0-000hf” ve “b-ed-d0a-e00h-0m0n0”. 5 blok için çaprazlama olasılıkları 0.3, 0.6, 0.4, 0.2 ve 0.7 olsun. 2 ve 5 numaralı blokların çaprazlama olasılıkları $\alpha = 0.5$ değerinden büyük olduğu için bu bloklar yer değiştirilir ve aşağıdaki çocuklar oluşturulur: “a-**ed**-dg0-0em0-**0m0n0**” ve “b-**cf**-d0a-e00h-**000hf**”.

(5) Mutasyon: Çaprazlamadan sonra, $C1$ ve $C2$ Dong vd. (2022) tarafından önerilen mutasyon işlemleri kullanılarak mutasyona uğratılır. Çaprazlamada olduğu gibi, mutasyon işlemleri önceden tanımlanmış bir olasılıkla ($\beta = 0.5$) ara düğümleri gösteren bloklara uygulanır. Yapılan çalışmada, mutasyon gerçekleştirmek üzere rastgele seçilen iki mutasyon operatörü kullanılmaktadır. İlk operatör, iç kenarları değiştirmeden işlemi değiştirir. İkinci operatör ise seçilen operatörü sabit tutarken daha önce bağlantısı olmayan farklı bir ara düğüm ile iç-kenar bağlantısını değiştirir. Örneğin, aşağıdaki olasılıklarla “a-ed-dg0-0em0-0m0n0” kodlamasına sahibiz: 0.2, 0.4, 0.8. 0.3, 0.4 ve ilk mutasyon işlemi seçilmiş olsun. Bu olasılıklar göz önüne alındığında, mutasyon sadece üçüncü bölüme uygulanır ve rastgele seçilen işlem başka bir işlemle değiştirilir: “a-ed-dg0-0em0-0m0n0” $\rightarrow$ “a-ed-**di**0-0em0-0m0n0”. Benzer şekilde, ikinci mutasyon işlemi üçüncü bölüme uygulanırsa, sonuç olarak aşağıdaki kodlama oluşturulur: “a-ed-dg0-0em0-0m0n0” $\rightarrow$ “a-ed-**d**0g-0em0-0m0n0”. Uygulanan mutasyon operatörü tipi rastgele seçilir.

(6) Değiştirme: Değiştirme adımında elitizm ve nesilsel değiştirmeyi uygulanmıştır.

3.1.3. Deneysel Çalışmalar

3.1.3.1. Veri Kümeleri

Yapılan çalışmada, farklı yöntemlerin karşılaştırılması için herkese açık bir şekilde sunulan 4 adet retinal kan damar segmentasyonu veri kümesi kullanılmıştır: DRIVE, CHASE_DB1, DCA1 ve CHUAC. Bu veri kümeleri Bölüm 1.6’da detaylı olarak açıklanmıştır.

3.1.3.2. Performans Ölçütleri

Bu çalışmada, farklı yöntemlerin segmentasyon başarısı beş farklı performans ölçütü kullanılarak değerlendirilmiştir: F1 puanı (F1), doğruluk (ACC), özgüllük (SPE), birleşimlerin keşimi (IOU) ve duyarlılık (SEN) (Wentao vd., 2022; Qu vd., 2023). Bu ölçütlerin hesaplanması için karmaşıklık matrisini oluşturan şu değerler kullanılmıştır: Doğru Pozitif (TP), Doğru Negatif (TN), Yanlış Pozitif (FP) ve Yanlış Negatif (FN). Doğru Pozitif (TP): gerçek referans görüntüsünde damar olarak işaretlenen ve tahmin edici tarafından da damar olarak tahmin edilen piksellerin sayısını; Doğru Negatif (TN): gerçek referans görüntüsünde damar olmadığı belirtilen ve tahmin edici tarafından da damar yok olarak tahmin edilen piksellerin sayısını; Yanlış Pozitif (FP): gerçek referans görüntüsünde damar yok olarak işaretlenen ancak tahmin edici tarafından damar var olarak tahmin edilen piksellerin sayısını ve son olarak Yanlış Negatif (FN): tahmin edici tarafından damar yok olarak sınıflandırılan ancak gerçek referans görüntüsünde damar olan piksellerin sayısını ifade etmek için kullanılır. Çalışmada kullanılan ölçütlerin her biri önerilen yöntemlerin performansını farklı kriterlere göre değerlendirmek için seçilmiştir. Bu tez çalışmasında, eşik değeri 0.5 seçilmiş ve seçilen eşik değeri kullanılarak segmente edilmiş ikili görüntüler oluşturulmuştur.

3.1.3.3. Geliştirme Detayları

Derin sinir ağlarının performansı, ağın uzun eğitim dönemleri boyunca tüm eğitim seti üzerinde eğitilmesi ve öğrenme sürecinde kullanılmayan bir test seti üzerinde değerlendirilmesi ile ölçülür. Ancak NAS araştırmalarında çok sayıda potansiyel ağın GPU üzerinde değerlendirilmesi gerekmektedir. Bu nedenle, hepsini uzun süreler boyunca tüm eğitim verileriyle eğitmek pratik değildir. Bu sorunun üstesinden gelmek için iki farklı değerlendirme aşaması kullanılmıştır: ön değerlendirme ve son değerlendirme.

Ön değerlendirme aşamasında, evrimsel algoritmalar tarafından üretilen ağların değerlendirilmesi gerçekleştirilmiştir. Ağlar eğitim verilerinin yarısı kullanılarak eğitilir ve eğitime dahil edilmeyen eğitim verilerinin %10'undan oluşan bir doğrulama seti üzerinde değerlendirilir. Değerlendirme sürecinde adaleti sağlamak için görüntüler rastgele seçilir ve üretilen her ağı değerlendirmek için aynı

setler kullanılır. Ayrıca, eğitim süresini verimli kullanmak için “*EarlyStopping*” kullanılmıştır. Maksimum eğitim dönemi 120 ve durdurma koşulunu belirleyen “*sabır (patience)*” parametresi 10 (Baldeon vd., 2021) olarak seçilmiştir. “*EarlyStopping*” sürecinde, en iyi doğrulama kaybı 10 ardışık adım boyunca değişmezse eğitim bu dönemde durdurulur. Aday ağın kalitesi, “*EarlyStopping*” işleminde elde edilen en iyi “*dice score*” kullanılarak belirlenir ve bu daha sonra eğitilen ağın uygunluk değeri olarak kullanılır. GPU üzerinde 500 ağ mimarisi eğitildikten sonra evrimsel algoritma sonlandırılır.

Son değerlendirme aşamasında, evrimsel algoritma tarafından elde edilen en iyi beş ağın gerçek performansı değerlendirilmiştir. Bu ağlar 200 eğitim dönemi boyunca tüm eğitim veri kümesi kullanılarak eğitilir ve eğitime dahil edilmeyen test kümesi üzerinde değerlendirilir. Ağlar, adil bir değerlendirme sağlamak için ön değerlendirme sırasında öğrenilen ağırlıklarla değil rastgele ağırlıklarla başlatılır. Test sonuçları ağın gerçek performansı olarak raporlanır.

Hem ön değerlendirme hem de son değerlendirme eğitim aşamalarında Wentao vd. (2022) tarafından sunulduğu gibi aşağıdaki ön işleme adımları kullanılmıştır:

- Renkli görüntüler gri ölçekli görüntülere dönüştürülür ve normalize edilir.
- Eğitim verisi miktarını artırmak amacıyla görüntülerden yamalar (patch) oluşturmak için 48 adımlı (*stride*) 48×48 boyutunda kayan pencereler kullanılmıştır (üst üste binen yamalar eğitim verisine dahil edilmemiştir).
- Eğitim görüntüleri, rastgele şekilde yatay ve dikey olarak çevrilerek ve [90, 180, 270] derece döndürülerek aşırı uyumu azaltmak için artırılır.

Eğitim adımlarından farklı olarak, son değerlendirme test aşamasında oluşturulan yamalar yerine test setindeki orijinal görüntüler kullanılır.

3.1.4. Sonuçlar ve Tartışma

Deneyler üç bölümde gerçekleştirilmiştir: Birinci ve ikinci bölüm MedUNAS'ın sırasıyla NAS çalışmaları ve literatürde sunulan son teknoloji modellerle karşılaştırılmasıdır. Son bölüm ise MedUNAS'ın incelenmesi için yapılan ablasyon çalışmalarını içermektedir.

3.1.4.1. Literatürdeki NAS Çalışmaları ile Karşılaştırma

Bu bölümde, MedUNAS yöntemi literatürde sunulan NAS çalışmalarıyla herkese açık olarak paylaşılan DRIVE ve CHASE_DB1 veri kümeleri üzerinde karşılaştırılmıştır. Ayrıca MedUNAS'ta arama yöntemi olarak kullanılan GA ve ODE'nin performansı da araştırılmıştır. Literatürde retinal damar segmentasyonu problemi için yapılmış sadece birkaç NAS çalışması bulunmaktadır. Bu çalışmada, son zamanlarda yapılan çalışmalar (Popat vd., 2020; Houreh vd., 2021; Wei vd., 2022, Rajesh vd., 2023) dahil edilmiş ve karşılaştırma için ilgili makalelerde raporlanan sonuçlar kullanılmıştır. Popat vd. (2020) ve HNAS (Houreh vd., 2021) çalışmalarında farklı modeller için sonuçlar rapor edilmiştir. Karşılaştırma için en az ve en çok parametreye sahip modeller seçilmiş ve bunlar sırasıyla v1 ve v2 olarak gösterilmiştir. Benzer şekilde, Genetik U-Net'te ikili görüş alanı olan (v1) ve olmayan (v2) iki farklı sonuç rapor edilmiştir.

Tablo 3.2, DRIVE veri seti için farklı NAS yaklaşımlarının parametre sayısı, ACC, SEN, SPE ve F1 ölçütlerine göre elde ettiği sonuçları göstermektedir. Tablo 3.2'te gösterilen kırmızı ve mavi renkler sırasıyla her bir ölçüt için en iyi iki değeri temsil etmektedir. Sonuçlar incelendiğinde, MedUNAS GA'nın ACC, SEN ve SPE ölçütleri için diğer çalışmalardan daha iyi performans gösterdiği gözlemlenebilir. Ayrıca, MedUNAS ODE; SEN ve F1 ölçütleri için ikinci en iyi performansı elde etmektedir. Genetik U-Net modelleri hariç, MedUNAS ODE ve GA yöntemleri ACC ve SEN açısından diğerlerinden büyük bir farkla daha iyidir. Ayrıca, MedUNAS ODE ve GA, BTU-Net, Popat vd. ve HNAS gibi yöntemlerden daha az parametreye sahip olmalarına rağmen tüm ölçütlerde (SPE hariç) daha iyi sonuçlar elde etmektedir. Genel olarak, bu sonuçlar MedUNAS tarafından keşfedilen mimarilerin damarları tespit etme konusunda daha yetenekli olduğunu göstermektedir.

Tablo 3.2: Önerilen yöntemlerin farklı NAS çalışmaları ile DRIVE veri kümesi üzerinde karşılaştırılması. Kırmızı ve mavi renkler sırasıyla her bir ölçüt için elde edilen en iyi iki sonucu göstermektedir.

Yöntem	Param (M)	ACC	SEN	SPE	F1
Popat vd. v1	0.18	93.56	59.67	98.50	-
Popat vd. v2	8.10	95.34	75.01	98.31	-
HNAS v1	157	95.42	77.07	98.10	81.08
HNAS v2	2.90	95.46	77.44	98.09	81.29
Genetic U-Net v1	0.27	95.77	83.00	97.58	83.14
Genetic U-Net v2	0.27	97.07	83.00	98.43	83.14
BTU-Net	16.80	96.89	-	98.44	81.78
MedUNAS GA	2.32	97.11	84.54	98.64	82.06
MedUNAS ODE	1.27	97.06	83.41	98.36	82.18

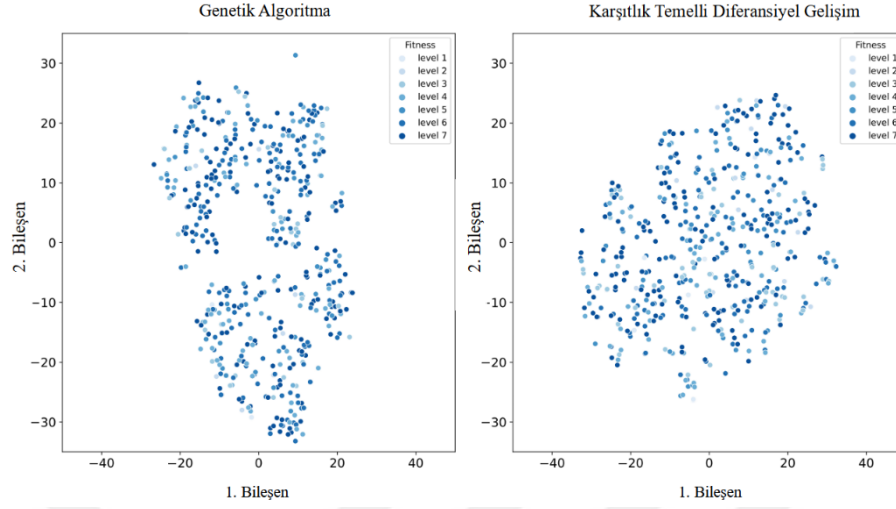
Ayrıca MedUNAS'ın performansı CHASE_DB1 veri seti için diğer NAS çalışmalarıyla karşılaştırılmıştır. Sonuçlar Tablo 3.3'te verilmiştir. MedUNAS ODE ve MedUNAS GA tüm ölçütler açısından rekabetçi sonuçlar vermektedir. BTU-Net'ten daha az parametreye sahip mimariler üretebilmektedir. Genetik U-Net, iki veri kümesi için en az sayıda parametreye sahip modelleri üretir. Bunun nedeni, özellik füzyonu için eleman-bazlı (element-wise) toplama kullanmaları, her bir ağı 130 eğitim dönemi için eğitmeleri ve özelleşmiş bir çaprazlama operatörü kullanmalarıdır. Buna karşılık, önerilen yaklaşımda özellik füzyonu için birleştirme (concatenation) kullanır, Erken Durdurma kullanılarak her ağı daha az eğitim dönemi boyunca eğitilir ve herhangi bir probleme özgü operatör kullanılmaz. Bildiğimiz kadarıyla, DCA1 ve CHUAC veri kümeleri için NAS çalışması bulunmamaktadır. Bu nedenle, NAS karşılaştırması için bu veri kümeleri deneysel çalışmaya dahil edilmemiştir.

Tablo 3.3: Önerilen yöntemlerin farklı NAS çalışmaları ile CHASE_DB1 üzerinde karşılaştırılması. Kırmızı ve mavi renkler sırasıyla her bir ölçüt için elde edilen en iyi iki sonucu göstermektedir.

Yöntem	Param (M)	ACC	SEN	SPE	F1
BTU-Net	16.80	97.87	-	98.64	81.09
Genetic U-Net v1	0.27	96.67	84.63	98.45	82.23
Genetic U-Net v2	0.27	97.69	84.63	98.57	82.23
MedUNAS GA	8.71	97.48	86.52	98.44	79.50
MedUNAS ODE	6.01	97.56	84.50	98.60	80.22

MedUNAS için GA ve ODE olarak adlandırılan arama yöntemlerinin etkisi, segmentasyon performansı ve model karmaşıklığı açısından herkese açık dört retinal kan damarı segmentasyonu veri kümesi üzerinde incelenmiştir. İlgili sonuçlar Tablo 3.4 ve Tablo 3.5'te sunulmuştur (son iki satıra bakınız). Sonuçlar, ODE'nin DRIVE üzerinde daha az parametreyle (1.27 M) F1 ve IOU için daha iyi sonuçlar elde ettiğini, GA'nın ise kalan ölçütler açısından daha iyi olduğunu göstermektedir. Ayrıca, ODE, CHASE_DB1 üzerinde daha az karmaşıklığa sahipken, tüm ölçütler için (SEN hariç) GA'dan daha iyi performans göstermektedir. DCA1'de ODE, SEN, F1 ve IOU için GA'dan daha iyi sonuçlar vermektedir. Ayrıca, ODE bu veri kümesi için 2.02 kat daha az eğitilebilir parametreye sahip bir model üretir. CHUAC için ODE, 1.50 kat daha az parametre ile tüm ölçütlerde (SEN ve SPE hariç) GA'dan daha iyi performans göstermektedir. Genel olarak, ODE'nin daha düşük karmaşıklığa sahip modeller üretirken tüm veri kümelerinde F1 ve IOU ölçütleri için daha iyi sonuçlar verdiği gözlemlenmiştir.

Bu çalışmada, GA ve ODE'nin arama performansları da incelenmiştir. GA ve ODE arama süreçlerinde elde edilen çözüm vektörleri, t-dağılımlı Stokastik Komşu Gömme (t-SNE) (Siems vd., 2020)'ye göre iki boyutlu olarak görselleştirilmiştir. Sonuçlar Şekil 3.4'de gösterilmiştir. GA tarafından üretilen çözümlerin belirli alanlarda kümелendiği görülmektedir. Öte yandan, ODE tarafından üretilen çözümlerin arama uzayında daha homojen bir şekilde dağıldığı ve daha geniş bir aralıkta çözümler üretildiği görülmektedir. Her iki yöntem de arama uzayını verimli bir şekilde keşfedebilmekte ve güçlü bir performans göstermektedir. Ayrıca, ODE'nin iyi uygunluk değerlerine sahip çeşitli çözümler üretmede daha iyi olduğu görülmektedir. Ayrıca, ~8M'den daha düşük parametrelere sahip küçük mimarilerin hem GA hem de ODE için en iyi segmentasyon hatalarını verdiği, ancak segmentasyon performansının genellikle parametre sayısı arttıkça kötüleştiği gözlemlenmiştir. Bu durum aşırı uyumdan kaynaklanmaktadır ve daha düşük parametrelerle iyi segmentasyon performansı veren ağlar bulmanın önemini göstermektedir.



Şekil 3.4: Genetik Algoritma ve Karşıtlık Temelli Diferansiyel Gelişim kullanılarak oluşturulan mimarilerin t-SNE izdüşümü. Seviyeler doğrulama doğruluk değerlerini göstermektedir. Daha yüksek seviyeler daha yüksek doğruluğa sahip çözümü temsil etmektedir.

3.1.4.2. Literatürdeki Son Teknoloji Çalışmalar ile Karşılaştırma

Bu bölümde, MedUNAS'ın performansı dört adet herkese açık retinal damar segmentasyon veri kümesi (DRIVE, CHASE_DB1, DCA1, CHUAC) üzerinde literatürde sunulan son teknoloji modellerle karşılaştırılmıştır. Sonuç tablolarında, algoritmaların performansı beş performans ölçütüne ve model karmaşıklığına göre sunulmuştur. Adil bir karşılaştırma gerçekleştirilmesi amacıyla son teknoloji modellerin sonuçları doğrudan Wentao vd. (2022) tarafından yapılan çalışmadan alınmıştır.

Tablo 3.4, DRIVE ve CHASE_DB1 veri kümeleri için önerilen yöntemlerin ve son teknoloji modellerin sonuçlarını sunmaktadır. Sonuçlar, önerilen yöntemlerin DRIVE veri kümesi için en düşük karmaşıklığa sahip modelleri ürettiğini ve son teknoloji modellere kıyasla oldukça rekabetçi segmentasyon performansı elde ettiğini ortaya koymaktadır. MedUNAS GA, DRIVE için son teknoloji modellere kıyasla ikinci en iyi ACC, SEN ve SPE ölçümlerini elde ederken, U-şeklindeki ağlara kıyasla en iyi ACC, SEN ve SPE sonuçlarını vermektedir. SGL, RV-GAN ve TP-Net hem DRIVE hem de CHASE_DB1 için en iyi sonuçları vermektedir. RV-GAN sırasıyla DRIVE'da en iyi ACC, SPE ve F1 değerlerini, CHASE_DB1'de ise en iyi F1 değerini elde etmektedir. Ancak RV-GAN, DRIVE ve CHASE_DB1 için önerilen

yöntemlerden sırasıyla 11.6 ve 2.46 kat daha fazla parametreye sahiptir. Buna ek olarak, MedUNAS ODE ve GA, CHASE_DB1 üzerinde ACC, SEN ve SPE açısından RV-GAN'dan daha iyi performans göstererek daha az karmaşıklığa sahip bir modelin ince damarları daha doğru bir şekilde tespit edebileceğini ve damarlı ve damarsız bölgeleri ayırt edebileceğini göstermektedir (Wu vd., 2021). SGL'den 12 kat daha az parametreye sahip olmasına rağmen MedUNAS ODE, ACC, SEN, SPE ve F1 ölçütleri açısından DRIVE üzerinde oldukça rekabetçi sonuçlar elde ederken, 6 kat daha az parametreye sahip MedUNAS GA, ACC, SEN ve SPE ölçütleri açısından DRIVE üzerinde SGL'den daha iyi sonuçlar vermektedir. Ayrıca, önerilen yöntemler DRIVE ve CHASE_DB1 üzerinde TP-Net'ten daha iyi ACC ve SPE ölçütleri sağlamaktadır. Elde edilen sonuçlara göre, önerilen yöntemlerin bu veri kümeleri için daha az karmaşıklık ile rekabetçi sonuçlar elde ettiği görülmektedir.

Tablo 3.4: DRIVE ve CHASE_DB1 üzerinde önerilen yöntemlerin son teknoloji yöntemlerle karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir. Att-U: Attention U-Net. **: İlgili yöntemin U-şekilli ağ olduğunu ifade etmektedir.

Yöntemler	Yıl	Param (M)	DRIVE					CHASE_DB1				
			ACC	SEN	SPE	F1	IOU	ACC	SEN	SPE	F1	IOU
U-Net **	2015	7.76	96.78	80.57	98.33	81.41	68.64	97.43	76.50	98.84	78.98	65.26
U-Net++**	2018	9.05	96.79	78.91	98.50	81.14	68.27	97.39	83.57	98.32	80.15	66.88
Att-U**	2018	8.73	96.62	79.06	98.31	80.39	67.21	97.30	83.84	98.20	79.64	66.17
HRNet	2019	9.64	97.04	80.40	98.64	82.65	70.43	97.58	84.43	98.47	81.48	68.75
CS-Net**	2019	8.40	96.32	81.70	98.54	80.39	70.17	97.42	84.00	98.32	80.42	67.25
AG-Net**	2019	-	96.92	81.00	98.48	-	69.65	97.43	81.86	98.48	-	66.69
RVSegNet**	2020	5.20	96.81	81.07	98.45	-	-	97.26	80.69	98.36	-	-
SCS-Net**	2021	-	96.97	82.89	98.38	-	-	97.44	83.65	98.39	-	-
SGL**	2021	15.53	97.05	83.80	98.34	83.16	-	97.71	86.90	98.43	82.71	-
VSSC Net	2021	8.05	96.27	78.27	98.21	-	-	96.33	72.33	98.65	-	-
RV-GAN	2021	14.81	97.90	79.27	99.69	86.90	-	96.97	81.99	98.06	89.57	-
FR-UNet**	2022	5.72	97.05	83.56	98.37	83.16	71.20	97.48	87.98	98.14	81.51	68.82
TP-Net	2023	2.67	96.29	87.49	97.58	85.69	-	97.30	86.00	98.41	85.18	-
MedUNAS GA**	2023	2.32-8.71	97.11	84.54	98.64	82.06	69.59	97.48	86.52	98.44	79.50	66.03
MedUNAS ODE**	2023	1.27 -6.01	97.06	83.41	98.36	82.18	69.77	97.56	84.50	98.60	80.22	67.01

Tablo 3.5: DCA1 ve CHUAC üzerinde önerilen yöntemlerin son teknoloji yöntemlerle karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir. Att-U: Attention U-Net. **: İlgili yöntemin U-şekilli ağ olduğunu ifade etmektedir

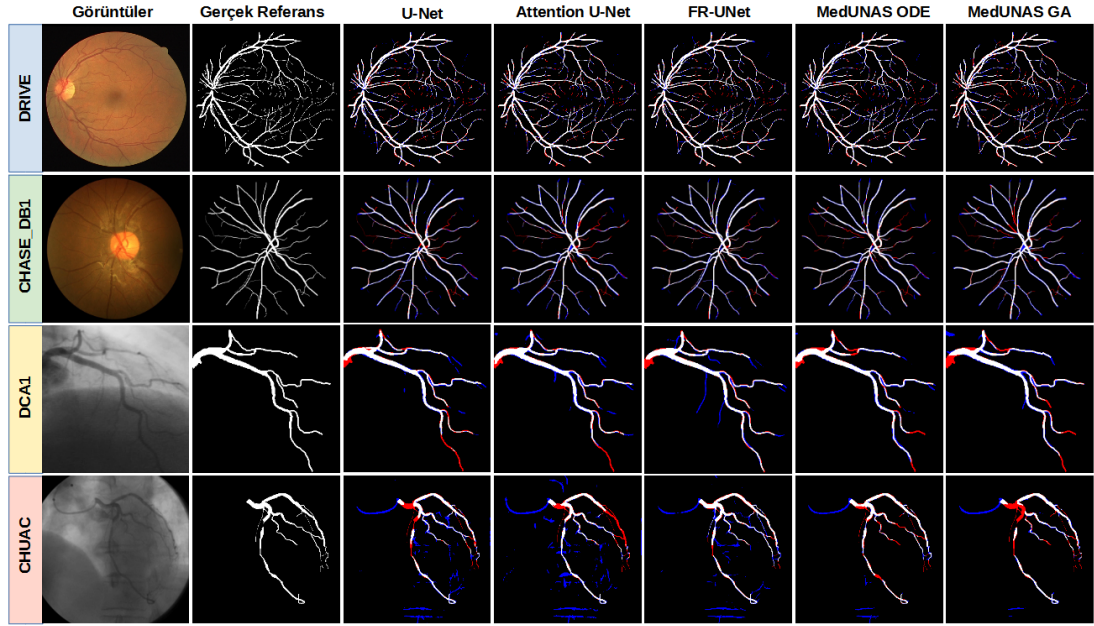
Yöntemler	Yıl	Param (M)	DCA1					CHUAC				
			ACC	SEN	SPE	F1	IOU	ACC	SEN	SPE	F1	IOU
U-Net **	2015	7.76	97.58	78.16	98.66	77.35	63.07	97.84	58.81	99.40	67.68	51.15
U-Net++**	2018	9.05	97.61	79.54	98.62	77.86	63.75	98.12	66.87	99.37	73.23	57.77
Att-U**	2018	8.73	97.55	79.86	98.53	77.48	63.24	98.00	65.26	99.13	71.54	55.69
HRNet	2019	9.64	97.77	80.07	98.76	79.19	65.54	98.11	74.56	99.06	75.26	60.33
CS-Net**	2019	8.40	97.63	78.95	98.67	77.90	63.80	97.96	67.35	99.18	71.71	55.89
VSSC Net	2021	8.05	97.88	82.48	98.75	80.22	67.08	98.03	81.71	98.68	76.01	61.51
FR-UNet**	2022	5.72	97.00	77.28	98.09	-	-	97.21	78.92	97.97	-	-
MedUNAS GA**	2023	2.32-8.71	98.00	80.89	99.05	78.20	64.33	97.92	78.65	99.16	71.96	56.31
MedUNAS ODE**	2023	1.27-6.01	97.93	84.12	98.89	78.22	64.37	98.07	78.29	99.12	74.56	59.54

DCA1 ve CHUAC veri kümeleri için elde edilen segmentasyon sonuçları Tablo 3.5'de sunulmuştur. Önerilen yöntemler, mevcut yaklaşımlara kıyasla en az karmaşıklığa sahip modelleri üretebilmektedir. MedUNAS GA, FR-UNet'ten 5.66 kat daha az parametre ile DCA1 üzerinde ACC ve SPE ölçütleri için üstün sonuçlar elde etmekte ve U-şekilli ağlara kıyasla ikinci en iyi F1 ve IOU değerlerini vermektedir. Ayrıca MedUNAS ODE, 11.44 kat daha az parametre ile DCA1 üzerinde rekabetçi bir performans göstermektedir. Bu veri kümesinde SEN için en iyi sonucu ve diğer ölçütler için U-şekilli ağlara kıyasla ikinci en iyi sonucu elde eder. CHUAC için, MedUNAS ODE modeli ACC, F1 ve IOU açısından U-şekilli ağlara kıyasla ikinci en iyi sonuçları elde etmektedir. Ayrıca, bu model CHUAC üzerinde SEN, F1 ve IOU metriklerine göre en iyi performansı gösteren FR-UNet'ten 5.34 kat daha az karmaşıktır. FR-UNet daha iyi performans göstermesine rağmen, bu beklenen bir durumdur çünkü eğitim için 160bin, 492bin, 193bin, 124bin yama görüntüsü kullanırken, önerilen yöntemler DRIVE, CHASE_DB1, DCA1 ve CHUAC veri kümeleri için sırasıyla yalnızca 3bin, 8bin, 4bin, 2bin yama görüntüsü ile eğitilmiştir. Son teknoloji yöntemlerle karşılaştırıldığında, bu çalışmada önerilen yöntemler kullanılarak üretilen modeller üç veri kümesi için en az karmaşık ağlardır, bu da bazı metrikler için daha az karmaşık ağlarla rekabetçi ve daha iyi sonuçlar elde edilebileceğini göstermektedir.

Şekil 3.5 farklı veri kümeleri için segmentasyon sonuçlarını görsel olarak karşılaştırmaktadır. Karşılaştırma için U-Net, Attention U-Net, FR-UNet (Wentao vd., 2022) (önceden eğitilmiş ağırlıklar mevcut), MedUNAS ODE ve MedUNAS GA

dikkate alınmıştır. Şekilde Yanlış Negatif (FN) ve Yanlış Pozitif (FP) hatalar gösterilmektedir; burada FN, gerçek referans görüntüsünde var olan ancak ağ tarafından tespit edilmeyen damarları, FP ise ağ tarafından tespit edilen ancak gerçek referans görüntüsünde var olmayan damar bölgelerini ifade etmektedir. İyi bir segmentasyon modelinin görüntülerde daha az FP ve FN pikseline sahip olması beklenir. Sonuçlar, tüm segmentasyon modellerinin DRIVE ve CHASE_DB1 veri kümelerinde benzer şekilde iyi performans gösterdiğini ortaya koymaktadır. Bununla birlikte, FR-UNet ve MedUNAS ODE modelleri bu veri kümeleri için ince damarları daha iyi tespit etmektedir. Aynı zamanda, tüm modeller çok ince damarları tespit etmekte başarısız olmaktadır. DCA1 ve CHUAC veri kümeleri için MedUNAS ODE ve GA, oluşturulan görüntülerde daha az yanlış segmentasyon bölgesi ile daha iyi FP hataları verir. U-Net ve Attention U-Net modelleri damar bölgelerini doğru bir şekilde segmente edememekte, bu da daha büyük bir FN ve FP hatasına neden olmaktadır. Bu nedenle, optimize edilmiş mimariler, tüm veri kümeleri için damar bölgelerini çıkarmada U-Net ve Attention U-Net'ten daha etkilidir.

Ayrıca özellikle ince damar segmentasyon performansını değerlendirmek için karşılaştırmalar gerçekleştirilmiş ve ince damarları çıkarmak için Yan vd., (2019) tarafından sunulan süreç takip edilmiştir. U-Net, Attention U-Net, FR-UNet ve MedUNAS DRIVE ve CHASE_DB1 veri kümeleri üzerinde karşılaştırılmıştır. Çünkü bu veri kümeleri diğer veri kümelerinden daha fazla ince damar içermektedir. MedUNAS ODE'nin DRIVE veri seti üzerinde daha az parametreyle diğer yöntemlerden daha iyi sonuçlar elde ettiği (SPE hariç) Tablo 3.6'da gösterilmektedir. Bununla birlikte, CHASE_DB1 veri kümesinde FR-UNet diğer yöntemlere kıyasla en iyi segmentasyon sonuçlarını (SPE hariç) vermiştir. Bu performans farkının CHASE_DB1'in DRIVE'a göre daha az eğitim örneğine ve daha az ince damar yapısına sahip olmasından kaynaklandığı düşünülmektedir.



Şekil 3.5: Çeşitli medikal görüntü segmentasyon modellerinin görsel karşılaştırması. Beyaz pikseller gerçek pozitif pikselleri, kırmızı pikseller yanlış negatif pikselleri ve mavi pikseller yanlış pozitif pikselleri göstermektedir.

Tablo 3.6: DRIVE ve CHASE_DB1 veri kümeleri için ince damar segmentasyonu kıyaslaması

Yöntem	DRIVE				
	F1	ACC	SEN	SPE	IOU
U-Net	78.10	71.31	65.30	96.42	64.73
Attention U-Net	77.87	71.05	64.95	96.49	64.39
MedUNAS ODE	79.99	73.32	67.83	96.30	67.21
MedUNAS GA	78.07	71.27	65.26	96.35	64.67
FR - UNet	77.98	71.11	65.05	96.49	64.49
Yöntem	CHASE_DB1				
	ACC	SEN	SPE	F1	IOU
U-Net	82.81	82.24	71.58	98.44	70.85
Attention U-Net	80.53	80.36	68.49	98.50	67.81
MedUNAS ODE	77.87	78.15	64.57	98.58	63.98
MedUNAS GA	80.90	80.55	68.66	98.48	67.98
FR-UNet	83.89	83.20	73.17	98.43	72.42

3.1.4.3. Ablasyon Çalışmaları

Bu bölümde, MedUNAS ODE üzerinde yapılan ablasyon çalışmaları anlatılmaktadır. MedUNAS ODE, MedUNAS GA'ya göre daha düşük parametrelerle daha iyi sonuçlar verdiği için (bkz. Tablo 3.5 ve Tablo 3.6), tüm ablasyon

çalışmaları bu model için gerçekleştirilmiştir. Kayıp fonksiyonunun etkisi incelenmiş ve modelin hücre ve beyin damarı segmentasyonu gibi farklı segmentasyon görevleri üzerindeki genelleme performansı değerlendirilmiştir. Ayrıca, modelin ince ayarlı (fine-tuning) ve ince ayarsız gürbüzlüğü farklı retinal kan damarı veri kümeleri üzerinde test edilmiştir. Tüm ablasyon çalışmaları için final değerlendirme aşamasında açıklanan aynı eğitim ve değerlendirme adımları gerçekleştirilmiştir. İlk olarak, modeller tüm eğitim örnekleri kullanılarak 200 eğitim dönemi boyunca eğitilir. Daha sonra gerçek performansın raporlanması için bu modeller test setindeki tüm görüntüler üzerinde değerlendirilmiştir.

3.1.4.3.1. Kayıp Fonksiyonları

Bu bölümde, farklı kayıp fonksiyonlarının segmentasyon sonucu üzerindeki etkisi incelenmiştir. En iyi kayıp fonksiyonunu belirlemek için MedUNAS ODE'nin DRIVE veri kümesi üzerindeki performansı çeşitli kayıp fonksiyonları kullanılarak değerlendirilmiştir: Dice, İkili Çapraz Entropi (BCE), Focal, Jaccard, Dice + BCE ve Dice + BCE + Jaccard (Wentao vd., 2022). Tablo 3.7, MedUNAS ODE için farklı kayıp fonksiyonlarının karşılaştırma sonuçlarını göstermektedir. Mavi ok işareti ($\uparrow$), Tablo 3.4'te DRIVE veri kümesi için gösterilen MedUNAS ODE sonucunun ilgili kayıp fonksiyonu kullanılarak iyileştirildiğini gösterir. Tablo 3.7'de gösterildiği gibi, Dice ACC ve SEN ölçümleri için en iyi sonuçları verirken, Jaccard F1 ve IOU için en iyi sonuçları üretmektedir. Ayrıca, Focal en iyi SPE sonuçlarını vermektedir. Sonuç olarak, kayıp fonksiyonunun doğru seçilmesinin segmentasyon performansını artırdığı ve bu fonksiyonun ayarlanmasının önemli olduğu görülmektedir.

Tablo 3.7: MedUNAS ODE için farklı kayıp fonksiyonlarının karşılaştırılması

	Performans Ölçütleri				
	MedUNAS ODE				
Kayıp Fonksiyonu	ACC	SEN	SPE	F1	IOU
Dice	97.11$\uparrow$	83.89$\uparrow$	98.63 $\uparrow$	82.05	69.60
BCE	97.07	76.67	99.23 $\uparrow$	81.09	68.27
Focal	96.84	71.45	99.58$\uparrow$	78.22	64.35
Jaccard	97.10 $\uparrow$	83.59 $\uparrow$	98.53 $\uparrow$	82.30$\uparrow$	69.96$\uparrow$
Dice + BCE	97.10 $\uparrow$	82.09	98.70 $\uparrow$	82.04	69.57
Dice + BCE + Jaccard	97.06	83.41	98.36	82.18	69.77

3.1.4.3.2. İnce Ayar Olmadan Çapraz-Eğitim Değerlendirmesi

Medikal görüntü segmentasyonunda çapraz eğitim değerlendirme, medikal görüntüleri segmente etmek için makine öğrenimi algoritmalarının etkinliğini değerlendirmeye yardımcı olan kritik bir süreçtir. Ayrıca bu yaklaşım, modelin genelleştirme performansını ve görülmeyen veri kümeleri üzerinde medikal görüntülerin segmentasyon yeteneğini değerlendirmeye yardımcı olur. Çapraz eğitim deneylerinde, DRIVE veri kümesi için elde edilen en iyi MedUNAS ODE modeli kullanılmıştır (bkz. Tablo 3.4). Bu model DRIVE eğitim seti üzerinde eğitilmiş ve yeniden eğitim yapılmadan DCA1, CHUAC ve CHASE_DB1 test setleri üzerinde değerlendirilmiştir. Sonuçlar Tablo 3.8'de verilmiştir.

Tablo 3.8: MedUNAS ODE için ince ayar olmadan çapraz-eğitim değerlendirme sonuçları

Test veri kümesi	ACC	SEN	SPE	FI	IOU
DCA1	94.89	60.47	97.25	46.16	30.81
CHUAC	96.08	49.51	98.49	43.81	28.26
CHASE_DB1	94.77	31.95	99.26	41.56	26.36

Tablo 3.8'de belirtildiği gibi, tüm ölçütler için daha düşük sonuçlar elde edilmiştir (CHASE_DB1 için SPE hariç). ACC ve SPE ölçütleri için küçük düşüşler meydana gelirken, diğer ölçütler için önemli azalmalar gözlenmektedir. Bu sonuçlar, eğitim ve test için kullanılan veri kümeleri arasındaki yapısal farklılıklardan da kaynaklanıyor olabilir. Örneğin, DRIVE'da çok sayıda ince damar bulunurken diğer veri kümelerinde ince damarlar için daha az etiketli örnek bulunmaktadır (Yan vd., 2019). Veri kümeleri arasında görüntüleme tekniği açısından da farklılıklar vardır. DRIVE görüntüleri Dijital Non-Mydriatic Retinal Kamera kullanılarak elde edilirken, CHUAC ve DCA1 görüntüleri koroner anjiyografi görüntüleridir. Sonuç olarak, modelin ince ayar yapılmadan çapraz eğitim için yeterli olmadığı görülmüştür.

3.1.4.3.3. İnce Ayar ile Çapraz-Eğitim Değerlendirmesi

Bu bölümde, ince ayarın çapraz eğitim üzerindeki etkisi araştırılmıştır. DRIVE üzerinde ODE kullanılarak aranan model (bkz. Tablo 3.4), ODE-DRIVE olarak isimlendirildi, diğer test veri kümeleri üzerinde değerlendirilmiştir. Ancak, 3.1.4.3.2 alt bölümünün aksine, test veri kümelerinin eğitim görüntüleri kullanılarak

seçilen modele ince ayar yapılmıştır. Ayrıca, bu bölümde ön eğitimin model üzerindeki etkisi de araştırılmaktadır. Ön eğitim, verilen problem için ilgili özellikleri hızlı bir şekilde öğrenmek için iyi bir başlangıç ağırlıkları seti sağlayabilir. Sonuçlar Tablo 3.9'da gösterilmektedir. Mavi ok işareti (↑), Tablo 3.4'te bildirilen sonucun iyileştirildiğini gösterir. ODE-DRIVE (ön eğitim olmadan) SEN, F1 ve IOU ölçütleri açısından DCA1 veri kümesi için Tablo 3.5'de raporlanandan daha iyi sonuçlar vermektedir. DCA1 için, ön eğitim yalnızca modelin SEN performansını artırır. Ayrıca, ODE-DRIVE modeli, ön eğitim olmadan, önemli ölçüde daha az parametreye sahip olmasına rağmen (4.73 kat daha az) tüm ölçütler için (SPE hariç) CHASE_DB1 veri kümesinden daha iyi sonuçlar elde etmektedir. Ayrıca, bu model SEN ölçütünde 4.13 puanlık bir iyileşme göstermektedir. Benzer şekilde, ön eğitim sadece CHASE_DB1 için SEN ölçütünde bir iyileşmeye yol açmaktadır. Bununla birlikte, ODE-DRIVE CHUAC veri kümesi için ön eğitim olmadan performansı iyileştirmezken, ön eğitim yalnızca SEN ölçütünü iyileştirmiştir. Elde edilen bulgular, her bir veri kümesi için NAS gerçekleştirmeden de rekabetçi sonuçlar elde edilebileceğini göstermektedir. Ayrıca, DRIVE veri kümesinde eğitilmiş ağırlıklarla başlamanın, muhtemelen veri kümeleri ve görüntüleme teknikleri arasındaki yapısal farklılıklar nedeniyle beklenen düzeyde iyileşmeye yol açmadığını göstermiştir. Yapılan analiz ayrıca ince ayar uygulanarak özellikle F1, SEN ve IOU ölçütleri açısından çapraz eğitim sonuçlarının önemli ölçüde iyileştirebileceğini ortaya koymaktadır.

Tablo 3.9: MedUNAS ODE için ince ayar ve ön-eğitim etkisinin çapraz-eğitim kümelerinde değerlendirilmesi

Test veri kümesi	ön-eğitim ile				
	ACC	SEN	SPE	FI	IOU
DCA1	97.89	84.31↑	98.72	78.20	64.33
CHUAC	93.88	84.52↑	94.56	49.11	32.84
CHASE DB1	97.38	88.08↑	98.07	79.71	66.29
Test veri kümesi	ön-eğitim olmadan				
	ACC	SEN	SPE	FI	IOU
DCA1	97.87	85.49↑	98.64	78.43↑	64.65↑
CHUAC	97.95	75.57	98.97	73.77	58.56
CHASE.DB1	97.57↑	88.63↑	98.42	80.50↑	67.40↑

3.1.4.3.4. Genelleştirme Performansı

Genelleştirme performansı değerlendirmesi, derin öğrenme modellerinin yeni görüntüler ve görevler üzerindeki kapasitesini değerlendirmek için gereklidir. Bu değerlendirme önemlidir çünkü medikal görüntüler büyük ölçüde değişiklik gösterebilir ve belirli bir veri kümesinde aranan bir model yeni verilerde iyi performans göstermeyebilir. Bu çalışmada, modelin (DRIVE üzerinde ODE ile aranan – ODE DRIVE) genelleştirme performansı iki ek veri kümesi üzerinde değerlendirilmiştir: Hücre segmentasyonu için Optofil (Kus vd., 2022) ve beyin damar segmentasyonu için KUVESG (Özkan vd., 2022). Optofil veri kümesi 4914 adet gri ölçekli hücre kültürü görüntüsü, KUVESG veri kümesi ise 335 adet fare korteksi görüntüsü içermektedir. Sonuçlar Tablo 3.10'da gösterilmiştir. ODE DRIVE modeli Optofil veri kümesiyle eğitilmiş ve Kus vd. (2022) tarafından gerçekleştirilen çalışmada olduğu gibi test kümesi üzerinde değerlendirilmiştir. Kus vd. (2022) tarafından gerçekleştirilen çalışmada bildirilen sonuçlar kullanılmamış ve adil bir karşılaştırma için Kus vd. (2022) tarafından gerçekleştirilen çalışmada bulunan en iyi model kullanılarak tüm ölçütler yeniden hesaplanmıştır. Sonuçlar, aranan modelin SPE hariç tüm ölçütler için daha iyi sonuçlar verdiğini göstermektedir. Ayrıca, aranan model KUVESG veri kümesi üzerinde eğitilmiş ve Kus vd. (2023)'de olduğu gibi test kümesi üzerinde değerlendirilmiştir. Aranan model, ACC ve SPE açısından Kus vd. (2023)'de bildirilen benzer sonuçlara ulaşırken, diğer ölçütler için daha düşük sonuçlar üretmektedir. Daha kalın damar görüntüleri içeren KUVESG veri setindeki yapısal farklılıklar bu sonuçları açıklayabilir. Ek olarak, ODE-DRIVE Kus vd. (2023) tarafından yapılan çalışmaya göre önemli ölçüde daha az parametreye (4.72 kat daha az) sahiptir. Sonuç olarak, ODE DRIVE hücre segmentasyonu görevleri için güçlü bir genelleme yeteneğine sahiptir ve beyin damarı segmentasyonu için rekabetçi sonuçlar vermiştir.

Tablo 3.10: MedUNAS ODE için ince ayar ile hücre ve beyin damar görüntüleri üzerinde genelleştirme performansının değerlendirilmesi.

Yöntemler	Optofil				
	ACC	SEN	SPE	F1	IOU
MedUNAS ODE	99.07	87.80	99.70	85.33	76.04
Kus vd ., (2023a)	99.03	80.28	99.80	83.88	73.81
Yöntemler	KUVESG				
	ACC	SEN	SPE	FI	IOU
MedUNAS ODE	94.72	69.01	98.12	71.51	58.24
Kus vd ., (2023b)	96.35	83.53	98.18	80.25	67.65

3.2. FARKLI KODLAMA YÖNTEMLERİNİN KARŞILAŞTIRILMASI

Bu bölümde, çözümleri oluşturan hücre yapılarının kodlanması için kullanılan farklı kodlama şemalarının retinal damar segmentasyonu kümeleri üzerinde karşılaştırılması gerçekleştirilmiştir. UNAS-Net (Bölüm 2’de detaylı olarak anlatılmıştır) ve MedUNAS (Bölüm 3’te detaylı olarak anlatılmıştır) çalışmalarında kullanılan kodlama şemaları bu bölümde karşılaştırılmıştır. Her iki çalışma için de ODE yöntemi çözümlerin oluşturulması için kullanılmıştır. Çözümlerin temsil edilmesi için sürekli uzay/gerçek değerlerden oluşan vektörler kullanılmıştır. Bu çözüm vektörleri kullanılarak, farklı kodlama şemaları ile hücre yapıları oluşturulmuş ve iki adet herkese açık retinal kan damar segmentasyonu veri kümesi üzerinde karşılaştırılmıştır. Farklı kodlama şemaları DRIVE ve CHASE_DB1 veri kümeleri için literatürdeki NAS çalışmaları ile karşılaştırılmış ve sonuçlar Tablo 3.11’de gösterilmiştir. Elde edilen sonuçlara bakıldığında UNAS-Net çalışmasında önerilen kodlama şeması kullanılarak MedUNAS çalışmasında önerilen kodlama şemasına göre en az %50 daha az karmaşıklığa sahip modeller üretildiği görülmüştür. MedUNAS, DRIVE veri kümesi için en iyi SEN performansını elde ederken, ACC ve F1 ölçütleri için en iyi ikinci sonuçları elde etmiştir. Aynı veri kümesi için UNAS-Net ise MedUNAS’a göre 4.09 kat daha az parametreye sahip model ile tüm ölçütler açısından rekabetçi sonuçlar elde etmiştir. MedUNAS ve UNAS-Net yöntemlerinin her ikisinin de 54.19 kata kadar daha az parametre ile DRIVE veri kümesi üzerinde HNAS ve BTU-Net çalışmalarından ACC, SEN ve F1 ölçütleri açısından daha iyi performans gösterdiği görülmüştür. Benzer sonuçların CHASE_DB1 veri kümesi için de elde edildiği ve önerilen kodlama şemalarının

Genetic U-Net hariç daha az parametre ile rekabetçi sonuçlar elde edebildiği gösterilmiştir.

Tablo 3.11: MedUNAS ve UNAS-Net çalışmalarında önerilen kodlama şemalarının DRIVE ve CHASE_DB1 veri kümeleri için literatürdeki NAS çalışmaları ile karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir.

Yöntemler	Param (M)	DRIVE				
		ACC	SEN	SPE	F1	IOU
HNAS	2.90	95.46	77.44	98.09	81.29	-
Genetic U-Net	0.27	97.07	83.00	98.43	83.14	-
BTU-Net	16.80	96.89	-	98.44	81.78	-
Popat vd.	8.10	95.34	75.01	98.31	-	-
UNAS-Net	0.31	97.02	82.00	98.42	81.80	69.23
MedUNAS	1.27	97.06	83.41	98.36	82.18	69.77
Yöntemler	Param (M)	CHASE_DB1				
		ACC	SEN	SPE	F1	IOU
Genetic U-Net	0.27	97.69	84.63	98.57	82.23	-
BTU-Net	16.80	97.87	-	98.64	81.09	-
UNAS-Net	3.41	97.55	87.50	98.34	80.52	67.43
MedUNAS	6.01	97.56	84.50	98.60	80.22	66.51

MedUNAS ve UNAS-Net çalışmalarında önerilen kodlama şemaları U-şekilli ağları kullanan son-teknoloji çalışmalar ile de karşılaştırılmış ve sonuçlar Tablo 3.12’de verilmiştir. Elde edilen sonuçlara bakıldığında MedUNAS ve UNAS-Net yöntemlerinin (MedUNAS – CHASE_DB1 hariç) tüm yöntemler arasında en az karmaşıklığa sahip modelleri ürettiği görülmüştür. DRIVE veri kümesi için MedUNAS’ın ACC açısından en iyi performansı gösterdiği ve F1 açısından ise en iyi ikinci performansa sahip olduğu görülmüştür. DRIVE veri kümesi için MedUNAS, FR-UNet çalışmasından 4.5 kat daha az parametre sayısına sahip bir model üretmesine rağmen bu yöntem ile tüm ölçütler açısından yüksek rekabetçi sonuçlar elde etmiştir. FR-UNet modelinin DRIVE veri kümesi için en iyi F1 ve IOU skorlarını, ACC ve SEN için ise en iyi ikinci skorları elde ettiği dikkate alınmalıdır. CHASE_DB1 veri kümesi için ise MedUNAS’ın ACC ve SPE ölçütleri açısından, UNAS-Net’in ise SEN ve IOU ölçütleri açısından en iyi ikinci sonuçları elde ettiği görülmüştür. UNAS-Net yöntemi CHASE_DB1 veri kümesi için en az karmaşıklığa sahip modeli üretmiştir. Aynı zamanda önerilen yöntemler, temel alınan U-Net modelinden 25 kata kadar daha az karmaşık model ile tüm ölçütler açısından daha iyi sonuçlar elde etmiştir. Bu sonuçlardan yola çıkarak, önerilen yöntemler ile temel

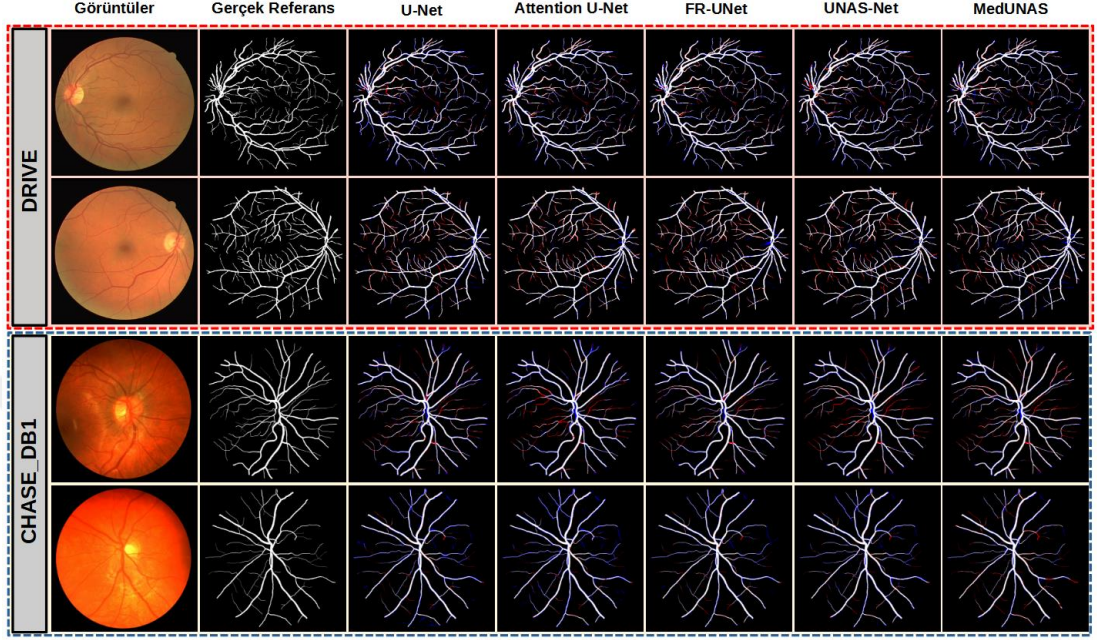
alınan U-Net modelinin hem model karmaşıklığı hem de segmentasyon performansı açısından iyileştirildiği görülmektedir.

Tablo 3.12: MedUNAS ve UNAS-Net çalışmalarında önerilen kodlama şemalarının DRIVE ve CHASE_DB1 veri kümeleri için literatürdeki son-teknoloji çalışmalar ile karşılaştırılması. Kırmızı ve mavi renkler her bir ölçüt için en iyi iki değeri göstermektedir.

Yöntemler	Yıl	Param (M)	DRIVE					CHASE_DB1				
			ACC	SEN	SPE	F1	IOU	ACC	SEN	SPE	F1	IOU
U-Net	2015	7.76	96.78	80.57	98.33	81.41	68.64	97.43	76.50	98.84	78.98	65.26
U-Net++	2018	9.05	96.79	78.91	98.50	81.14	68.27	97.39	83.57	98.32	80.15	66.88
Attention U-Net	2018	8.73	96.62	79.06	98.31	80.39	67.21	97.30	83.84	98.20	79.64	66.17
CS-Net	2019	8.40	96.32	81.70	98.54	80.39	70.17	97.42	84.00	98.32	80.42	67.25
AG-Net	2019	-	96.92	81.00	98.48	-	69.65	97.43	81.86	98.48	-	66.69
RVSegNet	2020	5.20	96.81	81.07	98.45	-	-	97.26	80.69	98.36	-	-
SCS-Net	2021	-	96.97	82.89	98.38	-	-	97.44	83.65	98.39	-	-
SGL	2021	15.53	97.05	83.80	98.34	83.16	-	97.71	86.90	98.43	82.71	-
FR-UNet	2022	5.72	97.05	83.56	98.37	83.16	71.20	97.48	87.98	98.14	81.51	68.82
UNAS-Net	2023	0.31 -3.41	97.02	82.00	98.42	81.80	69.23	97.55	87.50	98.34	80.52	67.43
MedUNAS	2023	1.27 -6.01	97.06	83.41	98.36	82.18	69.77	97.56	84.50	98.60	80.22	66.51

UNAS-Net ve MedUNAS yöntemleri tarafından elde edilen sonuçlar segmentasyon performansının görselleştirilmesi amacıyla Şekil 3.6'da sunulmuştur. Önerilen yöntemler, U-Net, Attention U-Net ve FR-UNet yöntemleri ile görsel olarak karşılaştırılmıştır. FR-UNet çalışmasında herkese açık bir şekilde kaynak kod dosyaları yayınlanmış ve aynı zamanda DRIVE ve CHASE_DB1 için eğitilmiş olan modelin ağırlıkları paylaşılmıştır. Bu nedenle, FR-UNet yöntemi karşılaştırmaya dahil edilmiştir. Benzer şekilde, U-Net ve Attention U-Net yöntemleri birçok çalışma tarafından temel alınan yöntemler olduğu ve ana makaleler referans alınarak tekrar kodlanabildiği için seçilmiştir. Bu yöntemler, Yanlış Negatif (FN) (gerçek referans görüntüsünde var olan ancak ağ tarafından tespit edilmeyen damarlar) ve Yanlış Pozitif (FP) (ağ tarafından tespit edilen ancak gerçek referans görüntüsünde var olmayan damar bölgeleri) pikseller üzerinden kıyaslanmıştır. İyi bir segmentasyon modelinin görüntülerde daha az FP ve FN pikseline sahip olması beklenir. Sonuçlar incelendiğinde, karşılaştırılan yöntemlerin DRIVE üzerinde benzer sonuçlar elde ettiği görülmüştür. Diğer yöntemlerden farklı olarak ise FR-UNet ve önerilen yöntemlerin (UNAS-Net ve MedUNAS) segmente edilmiş görüntüler üzerinde daha az FP ve FN piksele sahip olduğu görülmektedir. CHASE_DB1 veri kümesi için de DRIVE veri kümesine benzer sonuçların elde edildiği görülmüş, UNAS-Net ve

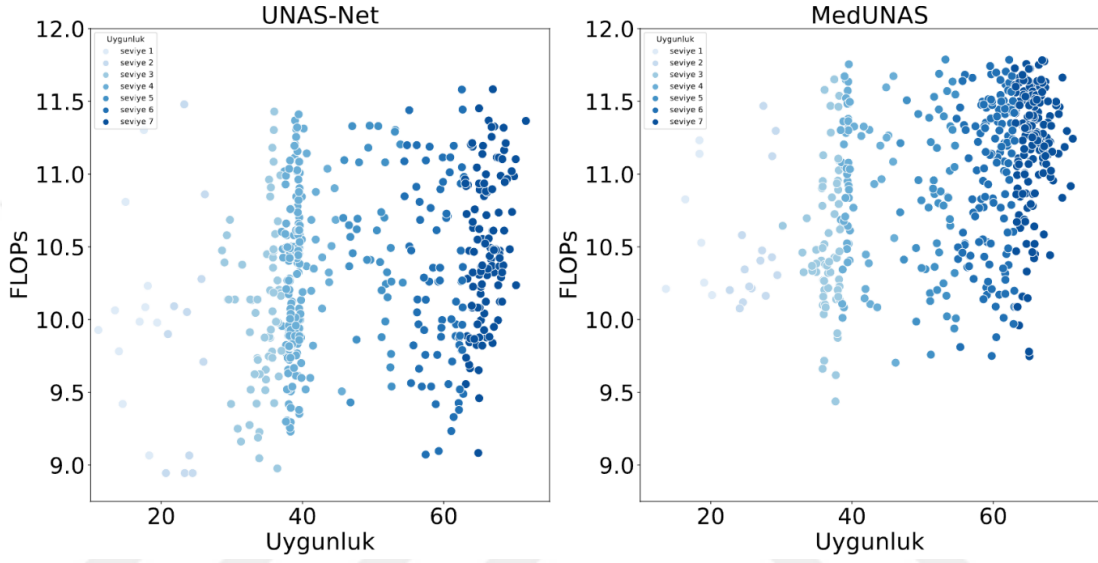
MedUNAS'ın daha fazla FN piksele sahip olduğu, FR-UNet'in ise daha fazla FP piksele sahip olduğu görülmüştür.



Şekil 3.6: Farklı segmentasyon modelleri tarafından elde edilen görüntülerin görsel olarak karşılaştırılması. Kırmızı renk yanlış segmente edilen negatif pikselleri (FN), mavi renk yanlış segmente edilen pozitif pikselleri (FP) ve beyaz renk ise doğru segmente edilen pikselleri (TP) temsil etmektedir.

Önerilen yöntemler, farklı kodlama stratejileri sunmakta ve aynı sinir mimarisi arama yöntemini (ODE) kullanmaktadır. Arama algoritması her iki yöntem için de ortak olduğundan dolayı farklı kodlama stratejileri adil bir şekilde karşılaştırılabilir. Bu çalışmada, UNAS-Net ve MedUNAS yöntemlerinde kullanılan farklı kodlama stratejileri ile arama uzayının ne kadar verimli dolaşıldığı karşılaştırılmıştır. Bu karşılaştırma, üretilen çözümlerin uygunluk değeri ve model karmaşıklığı üzerinden gerçekleştirilmiştir. Bu bölümde, UNAS-Net ve MedUNAS yöntemlerinin karmaşıklığının kıyaslanması için eğitilebilir parametre sayısı ölçütü (Param) kullanılmıştır. Bu ölçütün yanında, sinir mimarilerinin karmaşıklığı saniyede yapılan kayan noktalı hesaplama sayısını ölçen FLOPs ölçütü ile de kıyaslanabilmektedir. Bu nedenle, her iki yöntemin de karmaşıklığı diğer karşılaştırmalardan farklı olarak FLOPs değeri üzerinden gerçekleştirilmiştir. Şekil 3.7'de FLOPs değerleri okunabilirliği artırabilmek amacıyla log10 tabanında gösterilmiştir. Sonuçlar incelendiğinde, UNAS-Net ile üretilen ve yüksek uygunluk

değerine sahip çözümlerin (koyu mavi noktalar) geniş bir FLOPs aralığına yayıldığı görülmektedir. Buradan yola çıkarak, UNAS-Net kullanılarak daha az karmaşıklığa, düşük çıkarım sürelerine sahip olan iyi çözümlerin keşfedilebildiği görülmektedir. MedUNAS için ise benzer kaliteye sahip çözümlerin daha dar bir FLOPs aralığında keşfedilebildiği görülmektedir.



Şekil 3.7: Önerilen yöntemler için arama performansı kıyaslaması. Yöntemler model karmaşıklığı ve segmentasyon performansı açısından karşılaştırılmıştır. Görüntülerdeki koyu noktalar çözüm kalitesinin arttığını göstermektedir.

DÖRDÜNCÜ BÖLÜM

4. BaDENAS

4.1. Giriş

Tanımlanan arama uzayı ve çözüm kalitesinin değerlendirilmesi için seçilen ölçütün hesaplanma süresine bağlı olarak NAS çalışmalarının verimliliği değişkenlik gösterebilmektedir. Birçok derin öğrenme tabanlı NAS çalışmasında üretilen çözümlerin kalitesinin değerlendirilmesi uzun eğitim süreleri ile mümkün olabilmektedir. Bu durum, çalışma sürelerini uzatmakta ve makul sürelerde arama uzayında daha fazla çözümün ziyaret edilmesini zorlaştırmaktadır. Bu nedenle, tahmin ediciler ile NAS sürecinin birleştirilmesine duyulan ihtiyaç artmaktadır. Yapılan çalışmada retinal kan damar segmentasyonu için, BANANAS (White vd., 2021) ve Diferansiyel Gelişim (DE) yöntemlerinin avantajları birleştirilerek yeni bir NAS yöntemi önerilmiş ve önerilen yöntem BaDENAS olarak isimlendirilmiştir. Önerilen yöntem, BANANAS ve DE yöntemlerinden daha hızlı yakınsama sağlamış ve daha az karmaşık model ile yüksek segmentasyon başarısı göstermiştir.

4.2. Yöntem

4.2.1. Arama Uzayı

Yapılan çalışmada, Bölüm 2.2.2.1’de detaylı bir şekilde anlatılan ve U-Net’i optimize etmek için tasarlanmış olan UNAS-Net yapısı kullanılmış ve ilgili bölümde anlatılan arama uzayı aynı şekilde uygulanmıştır.

4.2.2. Kodlama Şeması

Bölüm 2.2.2.2’de detaylı bir şekilde anlatılan kodlama şeması BaDENAS yöntemi için de aynı şekilde uygulanmıştır. Üretilen çözümlerden ağ mimarilerinin oluşturulması için ilgili bölümde anlatılan kodlama şeması takip edilmiştir.

4.2.3. Arama Yöntemleri

4.2.3.1. Diferansiyel Gelişim

BaDENAS için kullanılan arama yöntemlerinden biri olan DE algoritması, Bölüm 1.5.1’de detaylı olarak anlatılmıştır.

4.2.3.2. Sinir Mimarileri için Bayes Optimizasyonu (BANANAS)

White vd., (2021) tarafından gerçekleştirilen çalışmada Bayes optimizasyonu ve NAS sürecinin avantajlarını birleştiren ve Sinir Mimarisi Araması için Sinir Mimarileri ile Bayes Optimizasyonu (Bayesian Optimization with Neural Architectures for Neural Architecture Search – BANANAS) olarak isimlendirilen yeni bir yöntem önerilmiştir.

BANANAS, olası sinir ağı mimarileri için arama uzayını verimli bir şekilde keşfetmek amacıyla Bayes optimizasyonunu kullanır ve çeşitli görevlerde diğer NAS yöntemlerinden daha iyi performans elde ettiği gösterilmiştir. BANANAS yönteminde, farklı sinir ağı mimarilerinin performansındaki belirsizliği temsil etmek için bir Bayes modeli kullanılır. Bu model, BANANAS'ın arama işlemi sırasında en umut verici mimarilere odaklanmasını sağlar ve algoritmanın yerel optimumlarda sıkışıp kalmasını önler. BANANAS ayrıca arama ve değerlendirme sürecini hızlandırmak için tahmin ediciler kullanır. Tahmin ediciler, BANANAS'ın bir sinir mimarisi ağını gerçekten eğitmeden performansını tahmin edebilmesine olanak tanır ve bu da önemli miktarda zaman tasarrufu sağlayabilir. BANANAS çalışmasında önerilen yöntem görüntü sınıflandırma, doğal dil işleme ve pekiştirmeli öğrenme gibi çeşitli görevler üzerinde değerlendirilmiştir. BANANAS tüm bu görevlerde daha iyi sonuçlar elde etmiş ve diğer NAS yöntemlerinden önemli ölçüde daha hızlıdır. BANANAS yöntemi genel olarak şu şekilde özetlenebilir: ilk olarak, yöntemin sinir ağı mimarileri için arama işlemini gerçekleştireceği arama uzayı tanımlanır. İkinci olarak, üretilen çözümlerin/sinir ağı mimarilerinin performansının tahmin edilmesi için bir tahmin edici (genellikle sinir ağı) eğitilir. Üçüncü olarak, arama uzayını verimli bir şekilde araştırmak ve üretilen mimariler arasından en umut verici mimarileri bulmak için Bayes optimizasyonu kullanılır. Dördüncü olarak, gradyan temelli optimizasyon kullanılarak en uygun mimari optimize edilir.

4.2.3.3. BaDENAS

Yapılan tez çalışmasında, BANANAS ve DE yaklaşımlarının sağladığı avantajlar kullanılarak, yenilikçi bir sinir mimarisi arama yöntemi önerilmiştir. Önerilen metod, BANANAS yöntemindeki komşu oluşturma adımlarını modifiye ederek daha üstün bir performans elde etmiştir.

Algoritma 4.1’de BaDENAS’ın sözde kodu verilmiştir. BaDENAS ilk adımda, rastgele bir şekilde üretilmiş NP adet çözüm içeren bir başlangıç popülasyonu (P_{G-1}) ile başlatılır. Popülasyon içerisindeki her bir çözüm $[0, 1]$ aralığında D boyutlu gerçek değerlerden oluşan vektörler ile temsil edilir. Başlatma aşamasından sonra P_{G-1} içerisindeki tüm çözümlerin gerçek uygunluk değerleri hesaplanır. Gerçek uygunluk değeri hesaplanan bu çözümler Tr kümesine dahil edilir ve bu küme tahmin edicinin eğitilmesi amacıyla kullanılır. Toplamda 10 katmandan oluşan yapay sinir ağı modeli tahmin edici olarak seçilmiştir. Seçilen tahmin edici yöntem üretilen çözüm vektörünü doğrudan girdi olarak kabul etmez. Eğitim veri kümesindeki her bir çözüm vektörü, yol tabanlı kodlama stratejisi (path-base encoding) ile kodlanır. Yol tabanlı kodlama yönteminde, her bir hücre yapısı (Şekil 2.1) birden fazla yoldan oluşan bir vektör olarak kodlanır. Her yol, ağdaki belirli bir katmandan diğerine olan bağlantıları temsil eder. Yolların kombinasyonu, nihai ağ yapısını oluşturur. Kodlama işlemi sırasında da, verilen çözüm vektöründen öncelikle hücre yapısı oluşturulur ve bu hücre yapısındaki yol bağlantıları oluşabilecek tüm yol bağlantıları ile kıyaslanır. Hücre yapısında var olan yol bağlantıları 1 olmayanlar ise 0 olarak işaretlenir. Bu işaretlemeler sonucunda verilen çözüm vektörünün yol tabanlı kodlaması elde edilir. Elde edilen kodlama vektörleri tahmin ediciye girdi olarak verilir ve her çözüm için tahmin edilen uygunluk değerleri çıktı olarak verilir. Eğitim kümesindeki çözümlerin gerçek uygunluk değerleri ile tahmini uygunluk değerleri arasındaki farka göre hata hesaplanır ve tahmin edici eğitilir. Tahmin edicinin eğitilmesinden sonra belirlenen sonlandırma kriteri gerçekleşene kadar Adım 7-19 arasındaki işlemler gerçekleştirilir. Adım 8-12 arasında başlangıç popülasyonundaki her bir çözüm için seçilen mutasyon ve çaprazlama operatörleri kullanılarak k adet aday çözüm üretilir ve üretilen aday çözümler C kümesine dahil edilir. Aday çözümlerden k adet üretilene kadar sırasıyla $rand1$, $rand2$, $currenttobest1$ ve $randtobest2$ (Gong vd., 2013) mutasyon operatörleri ile çözümler

mutasyona uğrattılır. Mutasyona uğrattılan çözümlerin her birine “ikili çaprazlama” işlemi uygulanır ve bu işlemler sonucunda C kümesindeki çözümlerin her birinin tahmini performansı tahmin edici kullanılarak hesaplanır (Adım 13). Bir sonraki adımda, C kümesinden NP adet umut vaadeden çözüm “Independent Thompson Sampling” yöntemi (White vd., 2021) ile seçilir ve seçilen çözümler için ekran kartı üzerinde gerçek uygunluk değeri hesaplaması gerçekleştirilir. Gerçek uygunluk değeri hesaplanan bu çözümler P_G popülasyonuna, bir sonraki popülasyon, aktarılır. P_G popülasyonundaki bu çözümler Tr kümesine eklenir ve bu kümedeki tüm çözümler ile tahmin edici tekrar eğitilir. Adım 7 ve 19 arasındaki tüm bu işlemler, gerçek uygunluk değeri hesaplanan çözümlerin sayısı 150 olduğunda sonlandırılır.

Algoritma 4.1: BaDENAS sözde kodu

Girdi: NP : Popülasyon boyutu; $f(\cdot)$: Amaç fonksiyonu; C : Aday çözümler kümesi; Tr : Eğitim kümesi

```

1  $G \leftarrow 1$ 
2  $Tr \leftarrow \emptyset$ 
3  $P_{G-1} \leftarrow$  Başlangıç popülasyonunu başlat
4  $Tr \leftarrow Tr \cup P_{G-1}$ 
5 Tahmin ediciyi  $Tr$  ile eğit
6 while Sonlandırma kriteri gerçekleşene kadar devam et do
7    $C \leftarrow \emptyset$ 
8   for  $i$  to  $NP$  do
9     for  $j$  to  $k$  do
10       $v \leftarrow \text{Mutasyon}(x_{i,G-1})$ 
11       $u \leftarrow \text{Çaprazlama}(x_{i,G-1}, v, CR)$ 
12       $C \leftarrow C \cup u$ 
13    $C$  kümesindeki çözümlerin tahmini performansını tahmin edici ile hesapla
14   Independent Thompson Sampling yöntemi ile  $C$  kümesinden  $NP$  adet çözüm seç
15   Bu çözümlerin gerçek uygunluk değerlerini hesapla
16    $P_G \leftarrow$  Seçilen çözümleri bir sonraki jenerasyona aktar
17    $Tr \leftarrow Tr \cup P_G$ 
18   Tahmin ediciyi  $Tr$  ile eğit
19    $G \leftarrow G + 1$ 

```

4.2.4. Deneyisel Çalışmalar

4.2.4.1. Veri Kümesi

Bu çalışmada, farklı yöntemler herkese açık olarak paylaşılan DRIVE retinal kan damar segmentasyonu veri kümesi üzerinde karşılaştırılmıştır. DRIVE veri kümesi, Bölüm 1.6.4’te detaylı olarak açıklanmıştır.

4.2.4.2. Geliştirme Detayları

BaDENAS, tahmin edici, edinim fonksiyonu (acquisition function) ve komşu sayısı gibi optimize edilmesi gereken hiper-parametrelere sahiptir. Bu hiper-parametrelerin doğru bir şekilde seçilmesi başarıyı doğrudan etkilemektedir ve bu nedenle ilgili hiper-parametrelerin doğru seçilmesi amacıyla bazı ön çalışmalar gerçekleştirilmiştir. Yapılan ön çalışmalarda tahmin edici yöntem için “XGBoost”, “Yapay Sinir Ağı” ve “Rastgele Orman” yöntemleri ve edinim fonksiyonu için de “Independent Thompson Sampling”, “Upper Confidence Bound” ve “Thompson Sampling” yöntemleri optimizasyona dahil edilmiştir. Gerçekleştirilen deneyler boyunca oluşabilecek tüm hiper-parametre çiftleri test edilmiş ve elde edilen sonuçlara bakıldığında tahmin edici olarak “Yapay Sinir Ağı” ve edinim fonksiyonu olarak da “Independent Thompson Sampling” yöntemlerinin beraber kullanılmasının en iyi sonuçları verdiği görülmüştür. Bu nedenle yapılan çalışmada, en iyi sonucu veren bu hiper-parametreler kullanılmıştır.

Derin öğrenme ağlarının gerçek performansının ölçülebilmesi için uzun eğitim dönemleri boyunca ağın tüm eğitim seti ile eğitilmesi ve öğrenme sürecinde yer almayan test seti üzerinde değerlendirilmesi gerekir. Ancak NAS çalışmalarında performansı GPU üzerinde değerlendirilmesi gereken birçok aday ağ bulunmaktadır. Tüm bu aday ağları, uzun eğitim periyotları boyunca tüm eğitim verileriyle eğitmek mümkün değildir. Bu nedenle, gerçekleştirilen değerlendirme adımları iki aşamaya ayrılmıştır: NAS dönemi ve NAS sonrası dönem.

NAS dönemi değerlendirme aşamasında, sezgisel yöntemin çalışması sırasında oluşturulan aday çözümler değerlendirilir. Bu adımda, aday çözümlerin eğitilmesi için eğitim veri kümesinin yarısından oluşan bir alt küme kullanılır ve bu alt kümenin %10'u kullanılarak oluşturulan doğrulama kümesi üzerinde ağın kalitesi değerlendirilir. Toplam eğitim süresini daha verimli kullanmak için Erken Durdurma

(Early Stopping) kullanılmış, aday çözümler maksimum 120 eğitim dönemi boyunca eğitilmiş ve 10 ardışık eğitim dönemi boyunca elde edilen en iyi doğrulama kaybı değeri iyileşmediğinde ilgili adımda eğitim sonlandırılmıştır. Eğitim sonunda elde edilen en iyi dice score değeri bu aday ağın kalitesini belirler ve çözümün uygunluk değeri olarak kullanılır. Seçilen sezgisel yöntemler, GPU üzerinde 150 sinir ağı mimarisi eğitildikten sonra sonlandırılır.

NAS sonrası değerlendirme aşamasında, sezgisel çalıştırma sonucunda (NAS dönemi sonu) elde edilen ilk beş ağın gerçek performansı değerlendirilir. Bu adımda, en iyi ağlar rastgele ağırlıklarla başlatılır ve 200 eğitim dönemi boyunca tüm eğitim veri seti ile eğitilir. Eğitilen bu ağlar, daha önce eğitime dahil edilmemiş (kısa süreli değerlendirme adımı dahil) test setinde değerlendirilir ve ağın gerçek performansı elde edilir.

Kullanılan veri kümesindeki eğitim örneklerinin sayısının artırılması ve aşırı öğrenmenin önüne geçebilmek amacıyla Bölüm 3.1.3.3'te anlatılan ön-işleme adımları bu çalışmada uygulanmıştır.

4.2.4.3. Performans Ölçütleri

BaDENAS'ın segmentasyon performansı açısından diğer yöntemler ile karşılaştırılması için F1 puanı (F1), doğruluk (ACC), duyarlılık (SEN), birleşimlerin kesişimi (IOU) ve özgüllük (SPE) ölçütleri kullanılmıştır. Bu ölçütler, Bölüm 3.1.3.2'de detaylı olarak açıklanmıştır. Aynı zamanda yöntemlerin model karmaşıklığı açısından karşılaştırılması için eğitilebilir parametre sayısı (Param) ölçütü kullanılmıştır.

4.2.5. Sonuçlar ve Tartışma

4.2.5.1. Literatürdeki NAS Çalışmaları ile Karşılaştırma

BaDENAS, temel alınan yöntemler (DE ve BANANAS) ve literatürde son zamanlarda gerçekleştirilen NAS çalışmaları ile karşılaştırılmıştır. Yöntemlerin hem segmentasyon performansı hem de model karmaşıklığı tartışılmış ve sonuçlar Tablo 4.1 verilmiştir.

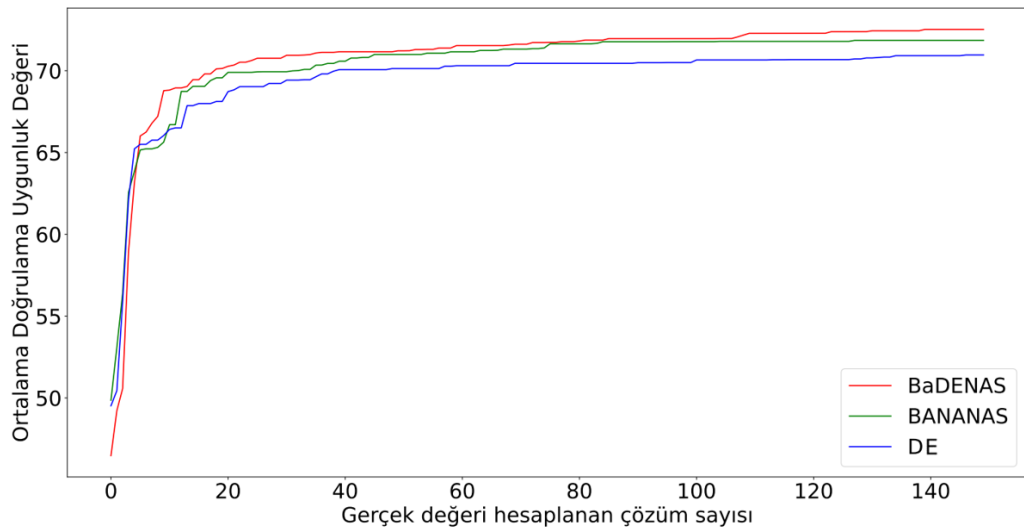
Yapılan karşılaştırmalarda rastsallığın etkisini en aza indirmek amacıyla farklı seed değerleri ile 5 bağımsız çalıştırma gerçekleştirilmiştir. Bu çalıştırmaların

ortalama sonuçları ve elde edilen standart sapma değerleri Tablo 4.1’de raporlanmıştır. DE, BANANAS ve BaDENAS dışındaki yöntemler için tekrar çalışma gerçekleştirilmemiş ve ilgili makalelerde sunulan sonuçlar alınmıştır. Tabloda gösterilen sonuçlar BaDENAS’ın karşılaştırılan yöntemlere kıyasla en az eğitilebilir parametre sayısına sahip modeli ürettiği ve bu modelin 152 kata kadar daha az eğitilebilir parametre sayısına sahip olduğunu göstermektedir. Ayrıca, BaDENAS model karmaşıklığı açısından en yakın rakibi olan Genetic U-Net yönteminden 2.45 kat daha az eğitilebilir parametre sayısı ile rekabetçi bir performans göstermiştir. BaDENAS, SPE hariç karşılaştırmaya dahil edilen diğer tüm segmentasyon performansı ölçütleri açısından en iyi ikinci sonuçları elde etmiştir. Aynı zamanda bu sonucun en az karmaşıklığa sahip olan model ile elde edildiğine de dikkat edilmelidir. BaDENAS ile BTU-Net yöntemi karşılaştırıldığında ise önerilen yöntemin SPE hariç diğer tüm ölçütlerde BTU-Net’ten daha iyi sonuçlar verdiği görülmektedir. Bu sonuçların BTU-Net yönteminden 152 kat daha az eğitilebilir parametreye sahip bir model ile elde edilebilmesi ise iyi segmentasyon sonuçlarının daha az karmaşık modeller ile de elde edilebileceğinin gösterilmesi açısından önemlidir. Ayrıca, BaDENAS temel alınan DE ve BANANAS yöntemleri ile de karşılaştırılmıştır. Elde edilen sonuçlar incelendiğinde, BaDENAS’ın temel alınan yöntemlerden daha az karmaşık bir model ürettiği ve SPE hariç tüm ölçütlerde daha iyi olduğu görülmektedir.

Tablo 4.1: Farklı yöntemlerin segmentasyon performansı, standart sapma ve karmaşıklık açısından karşılaştırılması. Kırmızı ve mavi renkler sırasıyla en iyi iki sonucu göstermektedir.

Yöntemler	Param (M)	DRIVE				
		ACC	SEN	SPE	F1	IOU
Popat vd., (2020)	8.10	95.34	75.01	98.31	-	-
HNAS	2.90	95.46	77.44	98.09	81.29	-
Genetic U-Net	0.27	97.07	83.00	98.43	83.14	-
BTU-Net	16.80	96.89	-	98.44	81.78	-
DE	0.19	96.98 ± 0.02	81.68 ± 1.02	98.38 ± 0.12	81.60 ± 0.10	68.95 ± 0.13
BANANAS	1.06	96.99 ± 0.06	82.30 ± 1.47	98.33 ± 0.18	81.75 ± 0.21	69.16 ± 0.30
BaDENAS	0.11	97.01 ± 0.04	82.30 ± 1.07	98.35 ± 0.14	81.86 ± 0.13	69.32 ± 0.19

Son olarak, BaDENAS yakınsama performansı açısından temel alınan yöntemler ile karşılaştırılmıştır. Karşılaştırma sonuçları, Şekil 4.1’de gösterilmiştir. Bu şekil incelendiğinde, BaDENAS’ın daha erken aşamalarda daha iyi ortalama uygunluk değerine sahip çözümleri yakalayabildiği görülmektedir. Bu sonuç önemlidir çünkü NAS yöntemlerinin daha az değerlendirme ile iyi çözümleri arama uzayında bulabilmesi önerilen yöntemin verimliliğini arttırmaktadır.



Şekil 4.1: Farklı yöntemlerin yakınsama performansı açısından karşılaştırılması

SONUÇ

Canlı hücrelerinde ve damarlarındaki yapısal değişimler, canlı vücudunda oluşabilecek hastalıkların en büyük sebeplerinden biridir (Todorov vd., 2020). Tüm beynin mikrovasküler yapısının izlenmesi, incelenmesi ve analiz edilmesi, beyin fonksiyonlarının ve oluşabilecek hastalıkların gözlenebilmesi anlamına gelmektedir. Bu sebeple damar ağı yapısının segmentasyonu, ilgili uzmanın bir hastalığın teşhis ve tedavisi hakkında görüş bildirmesi açısından büyük önem arz etmektedir. Deneyimli uzman ya da uzmanlar tarafından elle gerçekleştirilen hücre-damar ağı yapısı segmentasyonu, zaman alıcı bir süreçtir ve sürdürülebilirliği bulunmamaktadır. Hücre ve damar yapılarının makine öğrenmesi yaklaşımları ile otomatik segmentasyonu uzman gereksinimini ortadan kaldırarak, hızlı ve sürdürülebilir bir yapının oluşturulmasını sağlamaktadır. Ayrıca elle yapılan segmentasyonda, geleneksel görüntü işleme yaklaşımlarında uygulanan filtrelerin veya algoritmaların parametrelerinin incelenen örneğe özgü olarak yeniden kurgulanması gerekmektedir.

Son yıllarda derin öğrenme yöntemleri ile görüntü analizi yaklaşımlarının gelişmesiyle birlikte, otomatik hücre-damar segmentasyonu yüksek başarımla gerçekleştirilmektedir (Livne vd., 2019; Hilbert vd., 2020). Kullanılan bu derin sinir ağı modelleri kendi içerisinde optimize edilmesi gereken birçok hiper-parametre içermektedir. Bu hiper-parametrelerin ve derin sinir ağı mimarisinin seçilmesi yapılan segmentasyon çalışmasının başarısını doğrudan etkilemektedir (Wang vd., 2018; Baldominos vd., 2018; Dufourq vd., 2019). Bu sebeple, literatürde derin sinir ağı mimarilerinin otomatik bir şekilde oluşturulması için önerilen sinir mimarisi arama çalışmalarının sayısı giderek artmaktadır. Bu tez çalışmasında, medikal görüntülerin segmentasyonunu gerçekleştiren derin sinir ağı mimarilerinin otomatik belirlenmesi için yeni sinir mimarisi arama yaklaşımları önerilmiştir.

Yapılan tez çalışmasında, son yıllarda üzerinde yoğun olarak çalışılan “Sinir Mimarisi Arama” (Neural Architecture Search - NAS) problemi için üç farklı yöntem önerilmiştir. Önerilen yöntemler, hücre segmentasyonu, beyin damar segmentasyonu ve retinal kan damarı segmentasyonu gibi çeşitli medikal görüntüleme problemleri

üzerinde test edilmiş ve daha az karmaşık ağlar ile rekabetçi, hatta bazı problemler için daha da iyi sonuçlar elde edildiği gözlemlenmiştir.

Yapılan ilk çalışmada, hücre görüntülerinin otomatik segmentasyonunu gerçekleştiren sinir mimarilerinin aranması için yeni bir sinir mimarisi arama yaklaşımı önerilmiş ve UNAS-Net olarak isimlendirilmiştir. UNAS-Net, medikal görüntü segmentasyonu için iyi sonuçlar verdiği bilinen U-Net omurgasının optimizasyonu için kullanılmıştır. UNAS-Net, yapılan diğer çalışmalardan farklı olarak arama uzayında çeşitliliği korumak için literatürde önerilmiş olan farklı NAS tekniklerinin avantajlarını birleştirir (Liu vd., 2018; Ying vd., 2019). Oluşan her bir sinir ağının temsil edilmesi için NAS-Bench-101'de (Ying vd., 2019) sunulan hücre yapısı ve parametre ayarları kullanılmıştır. Ayrıca 'Diferansiyellenebilir Mimari Arama' (Differentiable Architecture Search - DARTS) algoritmasında sunulan ve başarılı sonuçlar verdiği kanıtlanan operasyonlar her bir hücre içerisinde kullanılır (Liu vd., 2018). Önerilen UNAS-Net yöntemi, DE, ODE ve LS olmak üzere üç farklı meta-sezgisel ile birlikte kullanılarak iki farklı hücre segmentasyonu veri kümesi üzerinde test edilmiştir: Optofil ve Hücre Çekirdeği veri seti. Elde edilen sonuçlara bakıldığında DE tabanlı yöntemlerin, hem segmentasyon performansı hem de hesaplama karmaşıklığı açısından temel alınan U-Net modelinden daha iyi performans gösterdiği görülmüştür. Aynı zamanda her iki veri kümesi için de DE yöntemi ile aranan modelin hem model karmaşıklığı hem de segmentasyon performansı açısından LS yönteminden de daha iyi olduğu görülmüştür.

İkinci çalışmada, U-Net ve onun bir başka varyantı olan Attention U-Net mimarilerini optimize etmek için iki DE tabanlı NAS yaklaşımı önerilmiştir. UNAS-Net ve Attention UNAS-Net mimarileri olarak adlandırılan bu yaklaşımlar, yaygın olarak kullanılan arama uzaylarının avantajlarını birleştiren hücre tabanlı mikro arama uzayı ve makro arama uzayından oluşmaktadır (Ying vd., 2019; Liu vd., 2018). Önerilen yaklaşımlar, 2B beyin damar segmentasyonu problemi için daha iyi segmentasyon performansı sağlayan ve mikro ve makro arama uzaylarını aynı anda arayan ağları bulur. Karşılaştırmalar, herkese açık bir veri kümesi (vesseINN) ve bu tez çalışması kapsamında önerilen kendi veri kümemiz (KUVESG) olmak üzere iki veri kümesi üzerinde gerçekleştirildi. Üretilen mimariler, segmentasyon performansı ve hesaplama karmaşıklığı açısından temel alınan U-Net ve Attention U-Net

mimarilerinden daha iyi performans göstermiştir. DE ve ODE sezgiselleri kullanılarak oluşturulan UNAS-Net ağı, 5.19 kata kadar daha az FLOPs değeri ile segmentasyon ölçütlerinin çoğunda U-Net'in segmentasyon performansından daha iyi performans göstermiştir. Benzer şekilde, Attention UNAS-Net, Attention U-net'ten daha iyi sonuçlar vermiş ve 9.15 kata kadar daha az karmaşık modeller üretmiştir. Segmentasyon performansı ve hesaplama karmaşıklığındaki bu önemli gelişmeler, önerilen yöntemlerin bu alanda uzman bilgisine sahip olmayan kullanıcılar tarafından da yaygın olarak kullanılması açısından önemlidir. Ayrıca, ablasyon çalışmalarıyla elde edilen sonuçların gelişmiş modellerin tasarımına ilişkin değerli bilgiler sağlayabileceği ve bu alanda gelecekte yapılacak araştırmalara rehberlik edebileceği bilinmektedir. Bu nedenle, kayıp fonksiyonu, model budama ve eşik değerlerinin segmentasyon performansı üzerindeki etkilerini araştıran ablasyon çalışmaları yaptık. Ablasyon sonuçları, kayıp fonksiyonlarının kombinasyonunun segmentasyon performansını artırdığını göstermektedir. Ayrıca, düşük eşik değerinin ince damar bilgisinin korunmasını iyileştirdiği ve segmentasyon performansını artırdığı gözlemlenmiştir. Ek olarak, hücre sayısının model karmaşıklığını belirlemede çok önemli bir rol oynadığı ve hücre yapısıyla eş zamanlı olarak optimize edilmesi gerektiği sonucuna varılmıştır. Ayrıca, genelleştirme deneylerinin sonuçları, önerilen yöntemlerin çeşitli veri kümelerinde iyi performans gösterdiğini ve bu da sonuçların genelleştirilebilir olduğunu göstermektedir.

Tezin üçüncü aşamasında, U-şekilli ağlar için yeni bir NAS yöntemi olan MedUNAS önerilmiştir. Bu yöntem, segmentasyon doğruluğu, hesaplama kaynakları ve eğitim süresi açısından retinal kan damar segmentasyonu problemi üzerinde son teknoloji yöntemler ve NAS çalışmaları ile oldukça rekabetçi sonuçlar vermiştir. Elde edilen sonuçlar, MedUNAS ODE ve GA ile karşılaştırılan veri kümelerinin çoğunda en iyi ve ikinci en iyi segmentasyon performansının elde edildiğini ve önerilen yöntemlerin U-şekilli son teknoloji yöntemlerin kullandığı parametre sayılarının %50'sinden daha azını kullandığını ortaya koymuştur. Ayrıca, önerilen yöntemler, değiştirilmiş kodlama şemaları ve esnek arama uzayı sebebiyle 15 kata kadar daha az parametreye sahip ağlar kullanıldığında bile herkese açık olarak sunulan dört adet retinal kan damar segmentasyonu veri kümesinde temel alınan U-Net'ten daha iyi performans göstermiştir. Bu durum, retinal kan damarı

segmentasyonu için daha az karmaşıklığa sahip yüksek kaliteli bir ağ yapısı oluşturmanın mümkün olduğunu göstermiştir. Yapılan çalışmada, ayrıca ODE ve GA'nın önerilen arama uzayı üzerindeki etkinliği vurgulanmış, ayrık ve sürekli kodlama stratejileri karşılaştırılmıştır. Ablasyon çalışmaları ile önerilen yöntemlerin hücre segmentasyonu gibi farklı tıbbi segmentasyon alanları için başarılı sonuçlar sağlayabildiği ve üretilen ağların ince ayar ile yeni medikal görevlere verimli bir şekilde uyarlanabileceği gösterilmiştir. Önerilen yaklaşım, klinik uygulamada retinal kan damar segmentasyonu sürecini otomatikleştirmek için değerli bir araç olarak hizmet edebilir ve klinisyenlerin çeşitli retina hastalıklarının ilerlemesini daha verimli ve doğru bir şekilde tanımlamasını ve izlemesini sağlayabilir.

Tezin dördüncü aşamasında, çözümleri oluşturan hücre yapılarının kodlanması için UNAS-Net ve MedUNAS yöntemlerinde kullanılan farklı kodlama şemalarının retinal damar segmentasyonu kümeleri üzerinde sayısal ve görsel olarak karşılaştırılması gerçekleştirilmiştir. Farklı kodlama şemaları DRIVE ve CHASE_DB1 veri kümeleri için literatürdeki NAS çalışmaları ile karşılaştırılmıştır. Elde edilen sonuçlara bakıldığında UNAS-Net çalışmasında önerilen kodlama şeması kullanılarak MedUNAS çalışmasında önerilen kodlama şemasına göre en az %50 daha az karmaşıklığa sahip modeller üretildiği görülmüştür. Bu, MedUNAS yöntemindeki iki iç-kenar kısıtı sebebiyle beklenen bir durumdur. Aynı zamanda önerilen yöntemler U-şekilli ağları kullanan son-teknoloji çalışmalar ile karşılaştırılmış ve bu yöntemlerin diğer yöntemlere kıyasla daha az karmaşıklıkta rekabetçi sonuçlar veren ağlar ürettiği görülmüştür. Son olarak, önerilen yöntemler, temel alınan U-Net modeli ile kıyaslanmış ve 25 kata kadar daha az karmaşık model ile tüm ölçütler açısından daha iyi sonuçlar elde edilmiştir.

Son olarak, tahmin ediciler ve NAS'ın beraber kullanıldığı yeni bir sinir mimarisi arama yöntemi önerilmiş ve BaDENAS olarak isimlendirilmiştir. Önerilen yöntemde, BANANAS ve DE yöntemlerinin avantajları birleştirilerek yeni bir sinir mimarisi arama yöntemi önerilmiştir. Önerilen bu yöntemde, tez çalışmasında önerilen diğer sinir mimarisi arama yöntemlerinden farklı olarak arama işlemi maliyetini azaltmak amacıyla tahmin ediciler kullanılmış ve sonuçlar karşılaştırılmıştır. Elde edilen sonuçlara bakıldığında BaDENAS'ın en az karmaşıklığa sahip modelleri ürettiği ve ACC, SEN, F1 ölçütleri için en iyi ikinci

sonuçları ürettiği görülmüştür. Aynı zamanda önerilen yöntem ile temel alınan DE ve BANANAS'ın performansının hem model karmaşıklığı hem de segmentasyon performansı açısından iyileştirildiği görülmüştür.

Sonuç olarak, gerçekleştirilen tez çalışmasında hücre, beyin damarı ve retinal kan damar segmentasyonu problemleri için otomatik segmentasyon gerçekleştirecek ağların araştırılması amacıyla üç yeni sinir mimarisi arama yöntemi önerilmiştir: UNAS-Net, MedUNAS ve BaDENAS. Önerilen yöntemlerin, birçok medikal segmentasyon problemi için daha az karmaşık ağlar ile rekabetçi hatta bazı problemler için daha iyi segmentasyon sonuçları verdiği gösterilmiştir. Aynı zamanda, önerilen yöntemler tarafından üretilen ağların kısa eğitim süreleri ile yeni medikal görevlere verimli bir şekilde uyarlanabileceği de gösterilmiştir. Önerilen yöntemler, klinik uygulamada hücre/damar segmentasyonu sürecini otomatikleştirmek için değerli bir araç olarak hizmet edebilir ve klinisyenlerin çeşitli hastalıklarının ilerlemesini daha verimli ve doğru bir şekilde tanımlamasını ve izlemesini sağlayabilir.

Yapılan tez çalışmasında, önerilen yöntemler medikal segmentasyon problemleri üzerinde test edilmiştir ancak bu yöntemler diğer segmentasyon, sınıflandırma vb. problemlere de kolaylıkla uygulanabilir. Gelecek çalışmalar arasında beyin-damar görüntülerinin 3-boyutlu segmentasyonu için önerilen yöntemlerin test edilmesi ve yeni sinir mimarisi arama yaklaşımlarının önerilmesi yer alabilir. Aynı zamanda, sinir mimarilerinin başarısını doğrudan etkileyen optimize edici yöntem, öğrenme oranı ve yığın boyutu gibi hiper-parametreler için de ayrı olarak bir optimizasyon çalışması gerçekleştirilebilir. Bu tez çalışmasında, tek amaç fonksiyonu ile çalışılmış ve gelecek çalışmalarda model karmaşıklığı ve segmentasyon performansını aynı anda iyileştirmeye çalışan iki veya çok amaçlı optimizasyon çalışması gerçekleştirilebilir. Tahmin ediciler ile meta-sezgisellerin birlikte kullanıldığı sinir mimarisi arama yaklaşımlarının daha az değerlendirme ile rekabetçi sonuçlar verebildiği görülmüş, bu alanda yapılacak çalışmaların öneminin giderek artacağı düşünülmektedir. Son olarak, önerilen yöntemlerin çok sınıflı segmentasyon problemleri üzerinde de test edilmesi gelecek çalışmalar arasında yer alabilir.

KAYNAKÇA

- Abràmoff, M. D., Garvin, M. K., & Sonka, M.** (2010). Retinal imaging and image analysis. *IEEE reviews in biomedical engineering*, 3, 169-208.
- Akbar, S., Sharif, M., Akram, M. U., Saba, T., Mahmood, T., & Kolivand, M.** (2019). Automated techniques for blood vessels segmentation through fundus retinal images: A review. *Microscopy research and technique*, 82(2), 153-170.
- Atli, I., & Gedik, O. S.** (2021). Sine-Net: A fully convolutional deep learning architecture for retinal blood vessel segmentation. *Engineering Science and Technology, an International Journal*, 24(2), 271-283.
- Awad, N., Mallik, N., & Hutter, F.** (2020). "Differential Evolution for Neural Architecture Search". *arXiv preprint arXiv:2012.06400*.
- Aydın, M., Kiraz, B., Eren, F., Uysal, Y., Morova, B., Özcan, S. C., ... & Kiraz, A.** (2022, February). A Deep Learning Model for Automated Segmentation of Fluorescence Cell images. In *Journal of Physics: Conference Series* (Vol. 2191, No. 1, p. 012003). IOP Publishing.
- Aydın, M.** (2021, June). Medical Image Segmentation Using Deep Neural Networks. In *2021 29th Signal Processing and Communications Applications Conference (SIU)* (pp. 1-4). IEEE.
- Aydın, O. U., Taha, A. A., Hilbert, A., Khalil, A. A., Galinovic, I., Fiebach, J. B., ... & Madai, V. I.** (2021). An evaluation of performance measures for arterial brain vessel segmentation. *BMC medical imaging*, 21(1), 1-12.
- Babu, A. A., Jegathesan, V., David, D. J., & Suriya, K. S.** (2022, April). Retinal Blood Vessels Segmentation Using Deep Learning Model-A Review. In *2022 6th International Conference on Devices, Circuits and Systems (ICDCS)* (pp. 375-379). IEEE.
- Back, T.** 1996. "Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms" (First Press). New York: Oxford University Press.
- Baker, B., Gupta, O., Naik, N., & Raskar, R.** (2016). Designing neural network architectures using reinforcement learning. *arXiv preprint arXiv:1611.02167*.

- Baldeon-Calisto, M., & Lai-Yuen, S. K.** (2020). AdaResU-Net: Multiobjective adaptive convolutional neural network for medical image segmentation. *Neurocomputing*, 392, 325-340.
- Baldominos, A., Saez, Y., Isasi, P.** (2018). "Evolutionary convolutional neural networks: An application to handwriting recognition", *Neurocomputing*, 283, 38-52.
- Bergstra, J., Bengio, Y.** 2012. "Random search for hyper-parameter optimization", *The Journal of Machine Learning Research*, 13(1), 281-305.
- Calisto, M. B., & Lai-Yuen, S. K.** (2021). EMONAS-Net: Efficient multiobjective neural architecture search using surrogate-assisted evolutionary algorithm for 3D medical image segmentation. *Artificial Intelligence in Medicine*, 119, 102154.
- Carballal, A., Novoa, F. J., Fernandez-Lozano, C., García-Guimaraes, M., Aldama-López, G., Calviño-Santos, R., ... & Pazos, A.** (2018). Automatic multiscale vascular image segmentation algorithm for coronary angiography. *Biomedical Signal Processing and Control*, 46, 1-9.
- Chaffer, C. L., & Weinberg, R. A.** (2011). A perspective on cancer cell metastasis. *science*, 331(6024), 1559-1564.
- Chen, C., Chuah, J. H., Ali, R., & Wang, Y.** (2021). Retinal vessel segmentation using deep learning: a review. *IEEE Access*, 9, 111985-112004.
- Cigla, C., & Alatan, A. A.** (2008, October). Region-based image segmentation via graph cuts. In 2008 15th IEEE International Conference on Image Processing (pp. 2272-2275). IEEE.
- de-la-Cruz-Martínez, S. J., & Mezura-Montes, E.** (2020, July). Boundary Constraint-Handling Methods in Differential Evolution for Mechanical Design Optimization. In 2020 IEEE Congress on Evolutionary Computation (CEC) (pp. 1-8). IEEE.
- Delgado-Ortet, M., Molina, A., Alférez, S., Rodellar, J., & Merino, A.** (2020). A deep learning approach for segmentation of red blood cell images and malaria detection. *Entropy*, 22(6), 657.
- Deshmukh, B. S., & Mankar, V. H.** (2014, January). Segmentation of microscopic images: A survey. In 2014 International Conference on Electronic Systems, Signal Processing and Computing Technologies (pp. 362-364). IEEE.
- Dice, L. R.** (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3), 297-302.

- Dong, F., Wu, D., Guo, C., Zhang, S., Yang, B., & Gong, X.** (2022). CRAUNet: A cascaded residual attention U-Net for retinal vessel segmentation. *Computers in Biology and Medicine*, 105651.
- Dong, J., Hou, B., Feng, L., Tang, H., Tan, K. C., & Ong, Y. S.** (2022). A cell-based fast memetic algorithm for automated convolutional neural architecture design. *IEEE Transactions on Neural Networks and Learning Systems*.
- Dufourq, E., Bassett, B. A.** (2017). “Eden: Evolutionary deep networks for efficient machine learning”. *Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)*, 110-115.
- Dzyubachyk, O., Niessen, W., Meijering, E.** (2008). Advanced level-set based multiple-cell segmentation and tracking in time-lapse fluorescence microscopy images, 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2008, (pp. 185-188), doi: 10.1109/ISBI.2008.4540963.
- Eladawi, N., Elmogy, M., Helmy, O., Aboelfetouh, A., Riad, A., Sandhu, H., ... & El-Baz, A.** (2017). Automatic blood vessels segmentation based on different retinal maps from OCTA scans. *Computers in biology and medicine*, 89, 150-161.
- Ess, A., Müller, T., Grabner, H., & Van Gool, L.** (2009, September). Segmentation-Based Urban Traffic Scene Understanding. In *BMVC* (Vol. 1, p. 2).
- Fan, Z., Wei, J., Zhu, G., Mo, J., & Li, W.** (2020). Evolutionary neural architecture search for retinal vessel segmentation. *arXiv preprint arXiv:2001.06678*.
- Feng, S., Zhuo, Z., Pan, D., & Tian, Q.** (2020). CcNet: A cross-connected convolutional network for segmenting retinal vessels using multi-scale features. *Neurocomputing*, 392, 268-276.
- Ferreira, A., Gentil, F., & Tavares, J. M. R.** (2014). Segmentation algorithms for ear image data towards biomechanical studies. *Computer methods in biomechanics and biomedical engineering*, 17(8), 888-904.
- Fraz, M. M., Remagnino, P., Hoppe, A., Uyyanonvara, B., Rudnicka, A. R., Owen, C. G., & Barman, S. A.** (2012). Blood vessel segmentation methodologies in retinal images—a survey. *Computer methods and programs in biomedicine*, 108(1), 407-433.

- Geiger, A., Lenz, P., & Urtasun, R.** (2012, June). Are we ready for autonomous driving? the kitti vision benchmark suite. In 2012 IEEE conference on computer vision and pattern recognition (pp. 3354-3361). IEEE.
- Gong, W., & Cai, Z.** (2013). Differential evolution with ranking-based mutation operators. *IEEE Transactions on Cybernetics*, 43(6), 2066-2081.
- Grathwohl, W., Creager, E., Ghasemipour, S. K. S., & Zemel, R.** (2018). Gradient-based optimization of neural network architecture.
- Guo, Y., Budak, Ü., & Şengür, A.** (2018). A novel retinal vessel detection approach based on multiple deep convolution neural networks. *Computer methods and programs in biomedicine*, 167, 43-48.
- Gülcü, A., & Kuş, Z.** (2020). Hyper-parameter selection in convolutional neural networks using microcanonical optimization algorithm. *IEEE Access*, 8, 52528-52540.
- Habibzadeh, M., Jannesari, M., Rezaei, Z., Baharvand, H., & Totonchi, M.** (2018, April). Automatic white blood cell classification using pre-trained deep learning models: ResNet and Inception. In Tenth international conference on machine vision (ICMV 2017) (Vol. 10696, p. 1069612). International Society for Optics and Photonics.
- Haft-Javaherian, M., Fang, L., Muse, V., Schaffer, C. B., Nishimura, N., & Sabuncu, M. R.** (2019). Deep convolutional neural networks for segmenting 3D in vivo multiphoton images of vasculature in Alzheimer disease mouse models. *PloS one*, 14(3), e0213539.
- He, K., Gkioxari, G., Dollár, P., & Girshick, R.** (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961-2969).
- Hilbert, A., Madai, V. I., Akay, E. M., Aydin, O. U., Behland, J., Sobesky, J., ... & Livne, M.** (2020). BRAVE-NET: fully automated arterial brain vessel segmentation in patients with cerebrovascular disease. *Frontiers in artificial intelligence*, 3, 78.
- Hollandi, R., Szkalitsy, A., Toth, T., Tasnadi, E., Molnar, C., Mathe, B., ... & Horvath, P.** (2020). nucleAIzer: a parameter-free deep learning framework for nucleus segmentation using image style transfer. *Cell Systems*, 10(5), 453-458.
- Houreh, Y., Mahdinejad, M., Naredo, E., Dias, D. M., & Ryan, C.** (2021). HNAS: Hyper Neural Architecture Search for Image Segmentation. In *ICAART* (2) (pp. 246-256).

- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q.** (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
- Hutter, F., Kotthoff, L., & Vanschoren, J.** (2019). Automated machine learning: methods, systems, challenges (p. 219). Springer Nature.
- I. Goodfellow, Y. Bengio, A. Courville and Y. Bengio** (2016), Deep learning (Vol. 1). Cambridge: MIT press.
- Jia, D., & Zhuang, X.** (2021). Learning-based algorithms for vessel tracking: A review. Computerized Medical Imaging and Graphics, 89, 101840.
- Kamran, S. A., Hossain, K. F., Tavakkoli, A., Zuckerbrod, S. L., Sanders, K. M., & Baker, S. A.** (2021). RV-GAN: Segmenting retinal vascular structure in fundus photographs using a novel multi-scale generative adversarial network. In Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part VIII 24 (pp. 34-44). Springer International Publishing.
- Khalaf, A. F., Yassine, I. A., & Fahmy, A. S.** (2016, September). Convolutional neural networks for deep feature learning in retinal vessel segmentation. In 2016 IEEE International Conference on Image Processing (ICIP) (pp. 385-388). IEEE.
- Kirbas, C., & Quek, F.** (2004). A review of vessel extraction techniques and algorithms. ACM Computing Surveys (CSUR), 36(2), 81-121.
- Koçak, T., Aydın, M., & Kiraz, B.** (2021, September). Empirical Comparison of Deep Neural Networks for Brain Vessel Segmentation. In 2021 6th International Conference on Computer Science and Engineering (UBMK) (pp. 751-756). IEEE.
- Kuş, Z.** (2019). Mikrokanonikal Optimizasyon Algoritması ile Konvolüsyonel Sinir Ağlarında Hiper Parametrelerin Optimize Edilmesi (Master's thesis, Fatih Sultan Mehmet Vakıf Üniversitesi, Lisansüstü Eğitim Enstitüsü).
- Kuş, Z., Aydın, M., Kiraz, B., & Can, B.** (2023a, February). Neural Architecture Search Using Metaheuristics for Automated Cell Segmentation. In Metaheuristics: 14th International Conference, MIC 2022, Syracuse, Italy, July 11–14, 2022, Proceedings (pp. 158-171). Cham: Springer International Publishing.

- Kuş, Z., Kiraz, B., Göksu, T. K., Aydın, M., Özkan, E., Vural, A., Kiraz, A., & Can, B.** (2023b). Differential Evolution-based Neural Architecture Search for Brain Vessel Segmentation. *Engineering Science and Technology, an International Journal*.
- LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., & Jackel, L.** (1989). Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P.** (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
- Lesage, D., Angelini, E. D., Bloch, I., & Funka-Lea, G.** (2009). A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical image analysis*, 13(6), 819-845.
- Li, Q., Feng, B., Xie, L., Liang, P., Zhang, H., & Wang, T.** (2015). A cross-modality learning approach for vessel segmentation in retinal images. *IEEE transactions on medical imaging*, 35(1), 109-118.
- Li, R. Q., Xie, X. L., Zhou, X. H., Liu, S. Q., Ni, Z. L., Zhou, Y. J., ... & Hou, Z. G.** (2021). Real-time multi-guidewire endpoint localization in fluoroscopy images. *IEEE Transactions on Medical Imaging*, 40(8), 2002-2014.
- Li, J., Gao, G., Liu, Y., & Yang, L.** (2023). MAGF-Net: A multiscale attention-guided fusion network for retinal vessel segmentation. *Measurement*, 206, 112316.
- Lin, G., Milan, A., Shen, C., & Reid, I.** (2017). Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1925-1934).
- Liskowski, P., & Krawiec, K.** (2016). Segmenting retinal blood vessels with deep neural networks. *IEEE transactions on medical imaging*, 35(11), 2369-2380.
- Liu, H., Simonyan, K., & Yang, Y.** (2018). Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*.
- Liu, W., Yang, H., Tian, T., Cao, Z., Pan, X., Xu, W., ... & Gao, F.** (2022). Full-resolution network and dual-threshold iteration for retinal vessel and coronary angiograph segmentation. *IEEE Journal of Biomedical and Health Informatics*, 26(9), 4623-4634.

- Liu, X., Song, L., Liu, S., & Zhang, Y.** (2021). A review of deep-learning-based medical image segmentation methods. *Sustainability*, 13(3), 1224.
- Liu, Y., Sun, Y., Xue, B., Zhang, M., Yen, G. G., & Tan, K. C.** (2021). A survey on evolutionary neural architecture search. *IEEE transactions on neural networks and learning systems*.
- Livne, M., Rieger, J., Aydin, O. U., Taha, A. A., Akay, E. M., Kossen, T., ... & Madai, V. I.** (2019). A U-Net deep learning framework for high performance vessel segmentation in patients with cerebrovascular disease. *Frontiers in neuroscience*, 13, 97.
- Long, J., Shelhamer, E., & Darrell, T.** (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431-3440).
- Ma, Z., Tavares, J. M. R., & Jorge, R. N.** (2009). A review on the current segmentation algorithms for medical images. In *Proceedings of the 1st International Conference on Imaging Theory and Applications (IMAGAPP)*.
- Mallipeddi, R., Suganthan, P. N., Pan, Q. K., & Tasgetiren, M. F.** (2011). Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied soft computing*, 11(2), 1679-1696.
- Meijering, E.** (2012). Cell segmentation: 50 years down the road [life sciences]. *IEEE signal processing magazine*, 29(5), 140-145.
- Mekheimer, K. S., & El Kot, M. A.** (2015). Suspension model for blood flow through catheterized curved artery with time-variant overlapping stenosis. *Engineering Science and Technology, an International Journal*, 18(3), 452-462.
- Moccia, S., De Momi, E., El Hadji, S., & Mattos, L. S.** (2018). Blood vessel segmentation algorithms—review of methods, datasets and evaluation metrics. *Computer methods and programs in biomedicine*, 158, 71-91.
- Mookiah, M. R. K., Hogg, S., MacGillivray, T. J., Prathiba, V., Pradeepa, R., Mohan, V., ... & Trucco, E.** (2021). A review of machine learning methods for retinal blood vessel segmentation and artery/vein classification. *Medical Image Analysis*, 68, 101905.
- Nath, S. K., Palaniappan, K., & Bunyak, F.** (2006, October). Cell segmentation using coupled level sets and graph-vertex coloring. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 101-108). Springer, Berlin, Heidelberg.

- Noh, H., Hong, S., & Han, B.** (2015). Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE international conference on computer vision (pp. 1520-1528).
- Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., ... & Rueckert, D.** (2018). Attention u-net: Learning where to look for the pancreas. arXiv preprint arXiv:1804.03999.
- Özkan, E., Vural, A., Kiraz, A., Kiraz, B., Aydın, M., Can, B., Kuş, Z., & Göksu, T.** (2022). KUVESG. Zenodo. <https://doi.org/10.5281/ZENODO.7383294>
- Popat, V., Mahdinejad, M., Cedeño, O. D., Naredo, E., & Ryan, C.** (2020). GA-based U-Net Architecture Optimization Applied to Retina Blood Vessel Segmentation. IJCCI, 192–199.
- Qin, A. K., Huang, V. L., & Suganthan, P. N.** (2008). Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE transactions on Evolutionary Computation, 13(2), 398-417.
- Qu, Z., Zhuo, L., Cao, J., Li, X., Yin, H., & Wang, Z.** (2023). TP-Net: Two-Path Network for Retinal Vessel Segmentation. IEEE Journal of Biomedical and Health Informatics.
- Rahnamayan, S., Tizhoosh, H. R., & Salama, M. M.** (2008). Opposition-based differential evolution. IEEE Transactions on Evolutionary computation, 12(1), 64-79.
- Rajesh, C., Sadam, R., & Kumar, S.** (2023). An evolutionary U-shaped network for Retinal Vessel Segmentation using Binary Teaching–Learning–Based Optimization. Biomedical Signal Processing and Control, 83, 104669.
- Real, E., Aggarwal, A., Huang, Y., & Le, Q. V.** (2019, July). Regularized evolution for image classifier architecture search. In Proceedings of the aaai conference on artificial intelligence (Vol. 33, No. 01, pp. 4780-4789).
- Rigamonti, R., & Lepetit, V.** (2012, October). Accurate and efficient linear structure segmentation by leveraging ad hoc features with learned filters. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 189-197). Springer, Berlin, Heidelberg.
- Ronneberger, O., Fischer, P., & Brox, T.** (2015). U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International

Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18 (pp. 234-241). Springer International Publishing.

- Samuel, P. M., & Veeramalai, T.** (2021). VSSC Net: vessel specific skip chain convolutional network for blood vessel segmentation. *Computer methods and programs in biomedicine*, 198, 105769.
- Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., ... & Cardona, A.** (2012). Fiji: an open-source platform for biological-image analysis. *Nature methods*, 9(7), 676-682.
- Sharif, J. M., Miswan, M. F., Ngadi, M. A., Salam, M. S. H., & bin Abdul Jamil, M. M.** (2012, February). Red blood cell segmentation using masking and watershed algorithm: A preliminary study. In 2012 International Conference on Biomedical Engineering (ICoBE) (pp. 258-262). IEEE.
- Shin, R., Packer, C., & Song, D.** (2018). Differentiable neural network architecture search.
- Siems, J., Zimmer, L., Zela, A., Lukasik, J., Keuper, M., & Hutter, F.** (2020). Nas-bench-301 and the case for surrogate benchmarks for neural architecture search. *arXiv preprint arXiv:2008.09777*, 11.
- Song, Y., Zhang, L., Chen, S., Ni, D., Li, B., Zhou, Y., ... & Wang, T.** (2014, August). A deep learning based framework for accurate segmentation of cervical cytoplasm and nuclei. In 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (pp. 2903-2906). IEEE.
- Soomro, T. A., Afifi, A. J., Zheng, L., Soomro, S., Gao, J., Hellwich, O., & Paul, M.** (2019). Deep learning models for retinal blood vessels segmentation: A review. *IEEE Access*, 7, 71696-71717.
- Sreejini, K. S., & Govindan, V. K.** (2019). Retrieval of pathological retina images using Bag of Visual Words and pLSA model. *Engineering Science and Technology, an International Journal*, 22(3), 777-785.
- Staal, J., Abràmoff, M. D., Niemeijer, M., Viergever, M. A., & Van Ginneken, B.** (2004). Ridge-based vessel segmentation in color images of the retina. *IEEE transactions on medical imaging*, 23(4), 501-509.
- Storn, R., & Price, K.** (1997). Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4), 341-359.
- Su, H., Yin, Z., Huh, S., & Kanade, T.** (2013). Cell segmentation in phase contrast microscopy images via semi-supervised classification over optics-related features. *Medical image analysis*, 17(7), 746-765.

- Sweeney, M. D., Kisler, K., Montagne, A., Toga, A. W., & Zlokovic, B. V.** (2018). The role of brain vasculature in neurodegenerative disorders. *Nature neuroscience*, 21(10), 1318-1331.
- Taha, A. A., & Hanbury, A.** (2015). Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15(1), 1-28.
- Teikari, P., Santos, M., Poon, C., & Hynynen, K.** (2016). Deep learning convolutional networks for multiphoton microscopy vasculature segmentation. *arXiv preprint arXiv:1606.02382*.
- Todorov, M. I., Paetzold, J. C., Schoppe, O., Tetteh, G., Shit, S., Efremov, V., ... & Ertürk, A.** (2020). Machine learning analysis of whole mouse brain vasculature. *Nature methods*, 17(4), 442-449.
- Van Grinsven, M. J., van Ginneken, B., Hoyng, C. B., Theelen, T., & Sánchez, C. I.** (2016). Fast convolutional neural network training using selective data sampling: Application to hemorrhage detection in color fundus images. *IEEE transactions on medical imaging*, 35(5), 1273-1284.
- Vonesch, C., Aguet, F., Vonesch, J. L., & Unser, M.** (2006). The colored revolution of bioimaging. *IEEE signal processing magazine*, 23(3), 20-31.
- Wang, B., Sun, Y., Xue, B., Zhang, M.** (2018). "Evolving deep convolutional neural networks by variable-length particle
- Wang, B., Qiu, S., & He, H.** (2019). Dual encoding u-net for retinal vessel segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part I 22* (pp. 84-92). Springer International Publishing.
- Wang, M., Zhou, X., Li, F., Huckins, J., King, R. W., & Wong, S. T.** (2008). Novel cell segmentation and online SVM for cell cycle phase identification in automated microscopy. *Bioinformatics*, 24(1), 94-101.
- Wang, X., He, W., Metaxas, D., Mathew, R., & White, E.** (2007, April). Cell segmentation and tracking using texture-adaptive snakes. In *2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro* (pp. 101-104). IEEE.
- Wang, X., Xiang, T., Zhang, C., Song, Y., Liu, D., Huang, H., & Cai, W.** (2021, September). Bix-nas: Searching efficient bi-directional architecture

- for medical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 229-238). Springer, Cham.
- Wei, C., Niu, C., Tang, Y., Wang, Y., Hu, H., & Liang, J.** (2022). Npenas: Neural predictor guided evolution for neural architecture search. *IEEE Transactions on Neural Networks and Learning Systems*.
- Wei, J., Zhu, G., Fan, Z., Liu, J., Rong, Y., Mo, J., ... & Chen, X.** (2022). Genetic U-Net: automatically designed deep networks for retinal vessel segmentation using a genetic algorithm. *IEEE Transactions on Medical Imaging*, 41(2), 292-307.
- White, C., Neiswanger, W., & Savani, Y.** (2021). BANANAS: Bayesian Optimization with Neural Architectures for Neural Architecture Search. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12), 10293-10301. <https://doi.org/10.1609/aaai.v35i12.17233>
- Wu, C., Zou, Y., & Zhan, J.** (2019, May). DA-U-Net: densely connected convolutional networks and decoder with attention gate for retinal vessel segmentation. In *IOP Conference Series: Materials Science and Engineering* (Vol. 533, No. 1, p. 012053). IOP Publishing.
- Wu, H., Wang, W., Zhong, J., Lei, B., Wen, Z., & Qin, J.** (2021). Scs-net: A scale and context sensitive network for retinal vessel segmentation. *Medical Image Analysis*, 70, 102025.
- Wu, Y., Xia, Y., Song, Y., Zhang, D., Liu, D., Zhang, C., & Cai, W.** (2019). Vessel-Net: retinal vessel segmentation under multi-path supervision. In *Medical Image Computing and Computer Assisted Intervention—MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part I 22* (pp. 264-272). Springer International Publishing.
- Xie, L., Yuille, A.** 2017. “Genetic cnn”, *Proceedings of the IEEE International Conference on Computer Vision* 1379-1388.
- Xu, A., Wang, L., Feng, S., & Qu, Y.** (2010, November). Threshold-based level set method of image segmentation. In *2010 Third International Conference on Intelligent Networks and Intelligent Systems* (pp. 703-706). IEEE.
- Yeghiazaryan, V., & Voiculescu, I. D.** (2018). Family of boundary overlap metrics for the evaluation of medical image segmentation. *Journal of Medical Imaging*, 5(1), 015006.

- Yin, Z., Bise, R., Chen, M., & Kanade, T.** (2010, April). Cell segmentation in microscopy imagery using a bag of local Bayesian classifiers. In 2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro (pp. 125-128). IEEE.
- Ying, C., Klein, A., Christiansen, E., Real, E., Murphy, K., & Hutter, F.** (2019, May). Nas-bench-101: Towards reproducible neural architecture search. In International Conference on Machine Learning (pp. 7105-7114). PMLR.
- Yu-Qian, Z., Wei-Hua, G., Zhen-Cheng, C., Jing-Tian, T., & Ling-Yun, L.** (2006, January). Medical images edge detection based on mathematical morphology. In 2005 IEEE engineering in medicine and biology 27th annual conference (pp. 6492-6495). IEEE.
- Zhang, Y., & Chung, A.** (2018, September). Deep supervision with additional labels for retinal vessel segmentation task. In International conference on medical image computing and computer-assisted intervention (pp. 83-91). Springer, Cham.
- Zhou, Y., Yu, H., & Shi, H.** (2021). Study group learning: Improving retinal vessel segmentation trained with noisy labels. In Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part I 24 (pp. 57-67). Springer International Publishing.
- Zhu, Z., Liu, C., Yang, D., Yuille, A., & Xu, D.** (2019, September). V-NAS: Neural architecture search for volumetric medical image segmentation. In 2019 International conference on 3d vision (3DV) (pp. 240-248). IEEE.
- Zoph, B., & Le, Q. V.** (2016). Neural architecture search with reinforcement learning. arXiv preprint arXiv:1611.01578.
- Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V.** (2018). Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 8697-8710).
- Zou, K. H., Warfield, S. K., Bharatha, A., Tempany, C. M., Kaus, M. R., Haker, S. J., ... & Kikinis, R.** (2004). Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports. Academic radiology, 11(2), 178-189.