



**BITEWING AĞIZ İÇİ RADYOGRAFİK
GÖRÜNTÜLERDE DERİN ÖĞRENME İLE DİŞ
SEGMENTASYONU**

Buse Yaren TEKİN

**2021
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ**

**Tez Danışmanı
Dr. Öğr. Üyesi Caner ÖZCAN**

**BITEWING AĞIZ İÇİ RADYOGRAFİK GÖRÜNTÜLERDE DERİN
ÖĞRENME İLE DİŞ SEGMENTASYONU**

Buse Yaren TEKİN

**T.C.
Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalında
Yüksek Lisans Tezi
Olarak Hazırlanmıştır**

**Tez Danışmanı
Dr. Öğr. Üyesi Caner ÖZCAN**

**KARABÜK
Eylül 2021**

Buse Yaren TEKİN tarafından hazırlanan “BITEWING AĞIZ İÇİ RADYOGRAFİK GÖRÜNTÜLERDE DERİN ÖĞRENME İLE DİŞ SEGMENTASYONU” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Dr. Öğr. Üyesi Caner ÖZCAN

.....

Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından Oy Birliği ile Bilgisayar Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 15/09/2021

Ünvanı, Adı SOYADI (Kurumu)

İmzası

Başkan : Dr. Öğr. Üyesi Emrullah SONUÇ (KBÜ)

.....

Üye : Dr. Öğr. Üyesi Caner ÖZCAN (KBÜ)

.....

Üye : Dr. Öğr. Üyesi Rafet DURGUT (BANÜ)

.....

KBÜ Lisansüstü Eğitim Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıştır.

Prof. Dr. Hasan SOLMAZ

.....

Lisansüstü Eğitim Enstitüsü Müdürü

“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Buse Yaren TEKİN

ÖZET

Yüksek Lisans Tezi

BITEWING AĞIZ İÇİ RADYOGRAFİK GÖRÜNTÜLERDE DERİN ÖĞRENME İLE DİŞ SEGMENTASYONU

Buse Yaren TEKİN

Karabük Üniversitesi

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:

Dr. Öğr. Üyesi Caner ÖZCAN

Eylül 2021, 61 sayfa

Bu çalışmada, diş sağlığı alanında detaylı incelenmesi gereken dişlerin tedavi ve tanı sürecinde kullanılan bitewing (ısıрма kanatlı) ağız içi radyografi görüntüleri üzerinde dişlerin segmentasyonu sağlanmıştır. Bitewing görüntüleme yöntemi, ağız içerisinde doğrudan görülemeyen, özellikle küçük ve büyük azı dişlerinin ara yüzlerindeki çürüklerin ve restorasyonların altında tekrarlayan çürüklerin tespiti için ideal bir tanı yöntemidir. Radyolojik görüntülerin uzmanlar tarafından raporlanması el ile yapılan, zaman alan ve dikkat gerektiren işlemlerdir. Bu işlemlerin başarılı ve hızlı bir şekilde gerçekleştirilebilmesi için derin öğrenme tabanlı bir yöntemle desteklenmesi amaçlanmıştır.

Çalışma kapsamında kullanılan bitewing diş görüntüleriyle ilgili yasal izinler alınmıştır ve çalışmada kullanılan veri kümesi, Ordu Diş Hekimliği Fakültesi Ağız, Diş ve Çene Radyolojisi Anabilim Dalı radyoloji görüntü arşivinden elde edilmiştir. Bu çalışmada, Evrişimli Sinir Ağları (CNN) türünden olan Bölgesel Tabanlı Evrişimli Sinir Ağı (R-CNN) kullanılmıştır. Derin öğrenme tabanlı yaklaşım ile dişlerin bölgelerinin tespiti ve otomatik segmentasyonu için ilgi alanında bulunan dişlere ait piksel bilgisi sağlayan R-CNN ağı türünden olan Mask R-CNN ağı kullanılmıştır. Bu tez çalışmasının amacı, bitewing radyografileri üzerindeki dişlerin otomatik olarak segmentasyonunun gerçekleştirilerek diş bölgelerinin tespit edilmesidir. Çalışmada sinir ağının eğitimi için i9 10980XE işlemcili ve NVIDIA Quadro RTX 5000 ekran kartına sahip bir bilgisayar kullanılmıştır. Ek olarak çalışmada, derin öğrenme kütüphanelerinden olan Keras ve TensorFlow başta olmak üzere Python programlama diline ait kütüphaneler kullanılmıştır. Mask R-CNN sinir ağı ile kullanılan omurga ağı ise Artık ağlardan (Residual networks) olan ResNet-101 ağıdır. Diğer ağ modellerine kıyasla artık değerlerin sonraki modele eklenmesiyle oluşan ResNet omurga ağı, bu sayede klasik model olmaktan çıkmıştır. Deneysel çalışmalar sonucunda ise gerekli şekil, grafik ve çizelgeler ile daha detaylı analiz yapılmış ve elde edilen sonuçlar tartışılmıştır.

Anahtar Sözcükler : Bitewing radyografisi, ağız içi görüntüleme, bilgisayar destekli teşhis, derin öğrenme, yapay sinir ağları, bölgesel tabanlı evrişimli sinir ağları, Mask R-CNN, örnek segmentasyon.

Bilim Kodu : 92414.

ABSTRACT

M. Sc. Thesis

DENTAL SEGMENTATION WITH DEEP LEARNING ON BITEWING INTRAORAL RADIOGRAPHIC IMAGES

Buse Yaren TEKİN

**Karabük University
Institute of Graduate Programs
Department of Computer Engineering**

Thesis Advisor:

Assist. Prof. Dr. Caner OZCAN

September 2021, 61 pages

In this study, segmentation of teeth was provided on bitewing intraoral radiography images used in the treatment and diagnosis process of teeth that need to be examined in detail in the field of dental health. Bitewing imaging method is an ideal diagnostic method for detecting caries that cannot be seen directly in the mouth, especially caries at the interfaces of premolars and molars, and recurrent caries under restorations. Reporting of radiological images by experts is a manual, time-consuming and careful process. It is aimed to support these processes with a deep learning-based method so that they can be performed successfully and quickly. Legal permissions were obtained for the bitewing tooth images used in the study, and the dataset used in the study was obtained from the radiology image archive of the Ordu Dentistry Faculty, Department of Oral, Dental and Maxillofacial Radiology. In this study, Regional Based Convolutional Neural Network (R-CNN), which is a type of Convolutional Neural

Networks (CNN), was used. The Mask R-CNN network, which is a type of R-CNN network that provides pixel information of the teeth in the region of interest, was used for the detection and automatic segmentation of the regions of the teeth with a deep learning-based approach. The aim of this thesis study is to determine the tooth regions by performing automatic segmentation of teeth on bitewing radiographs. In the study, a computer with i9 10980XE processor and NVIDIA Quadro RTX 5000 graphics card was used for the training of the neural network. In addition, libraries belonging to the Python programming language, especially Keras and TensorFlow, which are deep learning libraries, were used in the study. The backbone network used with Mask R-CNN neural network is ResNet-101 network, which is Residual networks. Compared to other network models, ResNet backbone network, which is formed by adding residual values to the next model, has thus ceased to be a classical model. As a result of the experimental studies, a more detailed analysis was made with the necessary figures, graphics and charts and the results were discussed.

Key Word : Bitewing radiography, intraoral imaging, computer aided diagnosis, deep learning, artificial neural networks, region based convolutional neural networks, Mask R-CNN, instance segmentation.

Science Code : 92414.

TEŞEKKÜR

Bu tez çalışmasının planlanmasında, araştırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteğini esirgemeyen, engin bilgi ve tecrübelerinden yararlandığım, yönlendirme ve bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren sayın hocam Dr. Öğr. Üyesi Caner ÖZCAN'a sonsuz teşekkürlerimi sunarım.

Çalışma süresince her türlü yol gösterici olan, olumlu tavırlarıyla beni cesaretlendiren, her zaman yanımda olan ve yoğun çalışmalarım sırasında bana sabır gösteren ve benden desteğini hiç esirgemeyen Cahit Berkay KAZANGİRLER'e teşekkür ederim.

Çalışmam boyunca bana destek olan ve yardımlarını hiç esirgemeyen Dilara ÖZDEMİR'e, tez çalışmam süresince bana ümit verdiği ve destek olduğu için Elif MEŞECİ'ye ve Süheda ÇİLEK'e teşekkür ederim. Çalışmanın geliştirilmesinde büyük rol oynayan ve çalışmada kullanılan veri etiketleme platformu olan DentiAssist'in geliştiricisi sayın hocam Öğr. Gör. Ahmet KARAOĞLU'na sonsuz teşekkürlerimi sunarım. Çalışma kapsamında kullanılan veri kümesinin elde edilmesinde büyük rol oynayan sayın Dr. Öğr. Üyesi Adem PEKİNCE'ye ve sayın Doç. Dr. Yasin YAŞA'ya sonsuz teşekkürlerimi sunarım.

Ek olarak bu tez çalışmamızı "2200272" proje numarası ile desteklemeye layık gören Türkiye Bilimsel ve Teknolojik Araştırma Kurumu TÜBİTAK'a teşekkürlerimi sunarım.

Sevgili aileme manevi hiçbir yardımı esirgemediği yanımda oldukları için tüm kalbimle teşekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
KABUL.....	ii
ÖZET	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ.....	xi
ÇİZELGELER DİZİNİ	xiii
SİMGELER VE KISALTMALAR DİZİNİ	xiv
BÖLÜM 1	1
GİRİŞ	1
BÖLÜM 2	3
AĞIZ İÇİ DENTAL RADYOGRAFİ.....	3
2.1. AĞIZ İÇİ RADYOGRAFİ TÜRLERİ	3
2.1.1. Periapikal Dental Radyografi	5
2.1.2. Okluzal Dental Radyografi	6
2.1.3. Bitewing Dental Radyografi	7
2.2. BITEWING DENTAL VERİ KÜMESİ VE VERİ ETİKETLEME	8
2.2.1. Bitewing Dental Veri Kümesi	9
2.2.2. Bitewing Veri Etiketleme	10
BÖLÜM 3	19
EVİRİŞİMLİ SİNİR AĞLARI VE MASK R-CNN.....	19
3.1. EVİRİŞİMLİ SİNİR AĞLARI (CNN).....	19
3.1.1. R-CNN.....	19
3.1.2. Faster R-CNN	21
3.1.3. Mask R-CNN.....	22

	<u>Sayfa</u>
3.2. DERİN ÖĞRENME KÜTÜPHANELERİ	24
3.2.1. Keras	25
3.2.2. TensorFlow	27
BÖLÜM 4	29
DİŞLERİN MASK R-CNN İLE SEGMENTASYONU	29
4.1. DERİN ÖĞRENME ORTAMININ KURULUMU	29
4.2. MASK R-CNN KONFIGÜRASYON SİSTEMİ	32
4.3. BÖLGE TEKLİF AĞI VE ÖZELLİK PİRAMİT AĞI SİSTEMİ	34
4.3.1. Bölge Teklif Ağı Sistemi	34
4.3.2. Özellik Piramit Ağı Sistemi	35
4.4. DİŞLERİN SEGMENTASYONU	40
BÖLÜM 5	45
DENEYSEL ÇALIŞMALAR	45
5.1. TENSORBOARD GÖRSELLEŞTİRME SONUÇLARI	47
BÖLÜM 6	53
SONUÇLAR VE ÖNERİLER	53
KAYNAKLAR	55
ÖZGEÇMİŞ	61

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1. Dental radyografi tipleri.....	4
Şekil 2.2. Örnek bir periapikal röntgen görüntüsü.....	6
Şekil 2.3. Okluzal filmin ağız içerisine yerleştirme adımları	6
Şekil 2.4. Örnek bir okluzal dental radyografi görüntüsü.....	7
Şekil 2.5. Örnek bir bitewing dental radyografi görüntüsü.....	8
Şekil 2.6. Veri kümesinden alınan örnek bitewing radyografi görüntüleri.....	10
Şekil 2.7. Veri etiketleme ara yüz ekranı	11
Şekil 2.8. Detaylandırılmış veri etiketleme paneli	11
Şekil 2.9. Görüntülerin ROI alanlarının belirlenmesi	12
Şekil 2.10. Klasik görüntü işleme yöntemleri ile segmentlere ayrılmış kategorilerden görüntü örnekleri	13
Şekil 2.11. Tespit edilen diş konumlarının merkezleri	14
Şekil 2.12. Faster R-CNN ile diş tespiti ve numaralandırma akış şeması	15
Şekil 2.13. Diş segmentasyonundan alınan test sonuçları	16
Şekil 2.14. Diş tespiti ve numaralandırma sistem mimarisi.....	16
Şekil 2.15. Derin evrişimli sinir ağı sistemi kullanarak bitewing radyografilerinde dişlerin numaralandırılması.....	17
Şekil 2.16. Bitewing görüntülerinde dişlerin düzenlenmesi	18
Şekil 3.1. Nesne algılama sistemine genel bakış.....	20
Şekil 3.2. R-CNN modeli örnek bir görüntü verisinde sonuçlar.....	20
Şekil 3.3. Faster R-CNN iş akış diyagramı	22
Şekil 3.4. Faster R-CNN örnek bir görsel üzerinde akış mimarisi	22
Şekil 3.5. Mask R-CNN iş akış mimarisi.....	23
Şekil 3.6. Mevcut verilerin geleneksel makine öğrenimi algoritmalarının performansı üzerindeki etkisini gösteren grafik	24
Şekil 3.7. Derin öğrenmede yazılım ve donanımın beraber çalışması.....	26
Şekil 3.8. Farklı derin öğrenme kütüphanelerinin arama motorunda aranması	26
Şekil 3.9. TensorFlow kullanım sebepleri ve avantajları.....	27
Şekil 3.10. Derin öğrenme projesi ilerleme döngüsü.....	28
Şekil 4.1. NVIDIA CUDA servisleri ve dağıtımları	30

Sayfa

Şekil 4.2. NVIDIA uyumlu ekran kartının hesaplama kapasitesi	30
Şekil 4.3. CUDA ve cuDNN araç kitlerinin kurulumu	31
Şekil 4.4. TensorFlow GPU ve CPU kontrolü	32
Şekil 4.5. Gerekli Python kütüphanelerinin içeriye aktarılması	32
Şekil 4.6. İlgili konfigürasyon parametrelerinin ve omurga ağının seçilmesi	33
Şekil 4.7. Seçilen konfigürasyon çıktıları	33
Şekil 4.8. Bölge teklif ağı.....	34
Şekil 4.9. Yapılan tahminlerle bir özellik piramidi olarak kullanılan model.....	35
Şekil 4.10. Örnek orijinal bitewing görüntüleri ve dış maskeleri.	36
Şekil 4.11. Dış nesnelere ait ayrıntılı maskelendirilmesi.	36
Şekil 4.12. Veri artırma ve yeniden boyutlandırmaya gidilerek dışların maske çıktıları	37
Şekil 4.13. Ankorların oluşturulmasına ait Python kodu	37
Şekil 4.14. Ankorların üretilmesi ve özetin yazdırılması.....	37
Şekil 4.15. Segmentasyon iş akış mimarisi	38
Şekil 4.16. 200 adet ROI'den gösterilen 10 adet örnek	39
Şekil 4.17. Seviyese olarak ankorların oluşumu ve özellik haritalarının indirgenmesi	39
Şekil 4.18. Örnek bitewing görüntüsünde sinir ağı eğitimi sonucu oluşan pozitif ankor kutuları	40
Şekil 4.19. Sinir ağının eğitimi için model oluşturma	40
Şekil 4.20. 100 iterasyon sayısı için sinir ağına var olan katmanların listelenmesi..	41
Şekil 4.21. Örnek bir bitewing görseline ait gerçek referans değerli maskeler	43
Şekil 4.22. Verilen görüntünün eğitim sonucu oluşturulan segmentasyon maskeleri	43
Şekil 5.1. TensorBoard histogram görselleştirme görseli	48
Şekil 5.2. TensorBoard uzantısının yüklenme kodu	49
Şekil 5.3. TensorBoard günlüklerinin görselleştirme kodu	49
Şekil 5.4. Eğitilen sinir ağı sonucu 50 iterasyon sayısına ait kayıp grafiği	50
Şekil 5.5. Eğitilen sinir ağı sonucu 100 iterasyon sayısına ait kayıp grafiği	50
Şekil 5.6. Eğitilen sinir ağı sonucu 400 iterasyona kadar olan kayıp grafikleri.....	51
Şekil 5.7. IOU değerinin gerçek referans ve tahmin sonucunda görselleştirilmesi ...	52
Şekil 5.8. Hassasiyet ve duyarlılık değerlerinin örnek görsel için grafikselendirilmesi	52

ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 3.1. VOC 2011'den alınan segmentasyon sonuçları.....	21
Çizelge 5.1. Önerilen sistemin test kümesinde ölçülen performans sonuçları.....	47

SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

- \mathcal{L} : kayıp fonksiyon değeri
 \mathcal{L}_{cls} : sınıf kayıp değeri
 \mathcal{L}_{box} : sınıflandırmadan alınan sınırlayıcı kutucuk değeri
 \mathcal{L}_{mask} : segmentasyondan alınan maske değeri
 k : toplam sınıf sayısı
 m : ilgi bölgesi boyutu
 y_{ij} : oluşan maskedeki bir koordinatın i ve j etiketi
 \hat{y}_{ij}^k : k sınıfı için tahmin edilen değer
 $\sum k$: k değeri için toplam sembolü

KISALTMALAR

- API : Application Programming Interface (Uygulama Programlama Arayüzü)
ASIC : Application Specific Integrated Circuit (Uygulamaya Özel Entegre Devre)
CNN : Convolutional Neural Network (Evrışimli Sinir Ağları)
CPU : Central Processing Unit (Merkezi İşlem Birimi)
CUDA : Compute Unified Device Architecture (Birleşik Aygıt Mimarisini Hesaplama)
CUDNN : CUDA Deep Neural Network Library (CUDA Derin Sinir Ağları Kütüphanesi)
DP : Doğru Pozitif
DN : Doğru Negatif
DVM : Destek Vektör Makineleri
FPN : Feature Pyramid Network (Özellik Piramit Ağı)

GPU	: Graphics Processing Unit (Grafik İşlem Birimi)
IOU	: Intersection Over Union (Birleşim Üzerinden Kesişim)
MAP	: Mean Average Precision (Ortalama Genel Hassasiyet)
MÖ	: Makine Öğrenimi
NMS	: Non-Maximum Suppression (Maksimum Olmayan Bastırma)
PA	: Pixel Accuracy (Piksel Doğruluğu)
R-CNN	: Region Based Convolutional Neural Networks (Bölgesel Tabanlı Evrişimli Sinir Ağları)
ResNet	: Residual Networks (Artık Ağlar)
RPN	: Region Proposal Network (Bölge Teklif Ağı)
ROI	: Region of Interest (İlgi Bölgesi)
TPU	: Tensor Processing Unit (Tensör İşleme Birimi)
YP	: Yanlış Pozitif
YN	: Yanlış Negatif

BÖLÜM 1

GİRİŞ

Diş hekimliğinde yaşanan en büyük zorluklardan biri, maksillofasiyal radyoloji uzmanının çok sayıda görüntüyü sınırlı bir sürede analiz etmesi ve hızlı bir tedavi planının sunulması zorunluluğudur. Ayrılan vaktin yeterli olmaması durumu, hekimden kaynaklı deneyim eksikliği ve konsantrasyon kaybı gibi insani koşullar, yapay zekâ destekli otomatik analiz sisteminin gerekliliğini tartışmasız bir şekilde ortaya koymaktadır. Yapay zekânın alt dalı olan derin öğrenme, genellikle sağlık görüntülerinin analizinde görüntü segmentasyonu ve sınıflandırma yapılarak kullanılır. Diş hekimliğinde X ışını kullanılarak vücudun incelenen bölgesinin kesitsel görüntüsünü (önden arkaya-coronal; yukarıdan aşağıya-axial; sağdan sola-sagittal) oluşturmaya yönelik 3 boyutlu bir radyolojik teşhis yöntemi olan Bilgisayarlı tomografi görüntüleri kullanıldığı gibi, rutin muayenede ise detaylı bakılması gereken dişlerin tedavi ve tanı sürecinde 2 boyutlu bitewing ağız içi radyografi görüntüleri tercih edilmektedir. Bitewing radyografiler, ağız içerisinde doğrudan görülemeyen, özellikle küçük ve büyük azı dişlerinin ara yüzlerindeki çürüklerin ve restorasyonların altında tekrarlayan çürüklerin tespiti için ideal bir tanı yöntemidir.

Bu tez çalışmasının amacı, bite wing radyografileri üzerindeki dişlerin otomatik olarak segmente edilip dişlerin tespit edilmesiyle diş hekimi ve radyologların iş yükünü hafifleterek kolaylık sağlamaktır. Bu çalışma boyunca ısırma kanatlı ifadesi bitewing olarak kullanılacaktır. Bitewing radyografisinde yer alan dişler öncelikle üst kategorisel açıdan segmentasyon maskelerine ayrıldı. Dişlerin analizi sırasında ilk ve en önemli işlem, diş nesnelere etiketlenmesi işlemidir. Bu işlemde diş nesnelere poligonal olarak etiketlenmektedir. Etiketleme işlemi için DentiAssist etiketleme modülü kullanılmıştır. Ardından ise derin öğrenme yöntemi olan evrişimli sinir ağı türüne ait Mask R-CNN kullanılmıştır. Ortaya konulan bu çalışma, genel kapsam gereği literatür taraması, deneysel çalışma, test ve analiz aşamalarından oluşmaktadır.

Çalışma kapsamında var olan bölümlerden ilki olan giriş bölümünde gerçekleştirilentez çalışmasının kısa özetine yer verilmiştir. İkinci bölümde, bu çalışmada kullanılan bitewing radyografi görüntülerini de kapsayan ağız içi radyografi görüntüleri ve türlerine yer verilmiştir. Diş hekimliği alanında geniş bir literatür taramasıyla çalışmaya aktarılmıştır. Üçüncü bölümde bu tez çalışmasında kullanılan bitewing ağız içi dental radyografik görüntülerinden örnekler verilerek etiketleme sisteminde diş etiketleri ve ek açıklamalarının oluşturulmasına yer verilmiştir. Bir sonraki bölüm olan dördüncü bölümde ise sinir ağı olarak kullanılan Mask R-CNN yöntemi detaylı bir biçimde tanıtılmıştır. Dişlerin segmentasyonuna yer verildiği bölüm olan beşinci bölümde ise bitewing görüntülerde bulunan her bir diş arka plandan soyutlanarak segmente edilip tespit edilmiştir.

Çalışmanın altıncı bölümünde, sinir ağının eğitilmesi sonucu gerçekleştirilen deneysel çalışmalara yer verilmiştir. Deneysel çalışmaların nihai sonuçlarının açıklandığı yedinci ve son bölümde ise deneysel çalışmalar sonucu elde edilen bulgular, deneysel çalışmanın amacına uygun bir biçimde yorumlanarak sonuçlandırılmıştır. Diş nesnelerinin analizi boyunca ortaya çıkarılan grafiklerle deneysel sonuçlar desteklenmiştir.

BÖLÜM 2

AĞIZ İÇİ DENTAL RADYOGRAFİ VE BITEWING VERİ ETİKETLEME

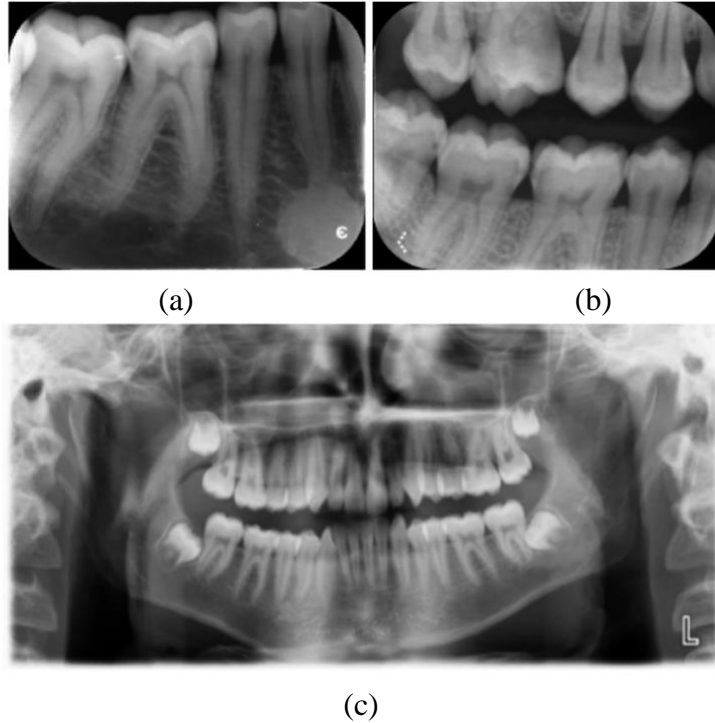
Derin öğrenme modelleri medikal görüntülerin analizinde, görüntü segmentasyonu ve sınıflandırmada kullanılmaktadır [1]. X ışınları, 1895 yılında Prof. Wilhelm Conrad Roentgen tarafından keşfedildikten kısa süre sonra ve ilk diş radyografisi Dr. Otto Walkhoff tarafından kullanılmıştır [2]. X ışını kullanılarak vücudun incelenen bölgesinin kesitsel görüntüsünü oluşturmaya yönelik radyolojik teşhis yöntemidir [3]. Ayrıca son yıllarda yaygınlaşan dijital radyografi tekniklerinin sağlıkta sıkça tercih edilmesi, hasta ve sağlık görevlisinin minimum radyasyon ile yüksek kalitede görüntü alınmasını sağlamıştır [4]. Diş hekimliği radyolojisinde, ağız içerisinde doğrudan görülemeyen küçük ve büyük azı dişlerinin ara yüzlerindeki çürüklerin ve restorasyonların altında tekrarlayan çürüklerin tespiti için bitewing radyografiler ideal bir tanı yöntemi olarak tercih edilmektedir. Bu tip dental radyografiler üzerinde yapılan analizler ile klinisyenler çürük tespitinde erken tanı koyabilme imkanı elde ederler [5].

Diş hekimleri tarafından yapılan güncel diş muayenelerinde çok sayıda radyografik teknikler kullanılmaktadır. Radyolojik görüntülerin maksillofasiyal radyologlar tarafından yorumlanması zaman alan ve dikkat gerektiren bir süreçtir. Özellikle yoğun çalışma temposu ve yorgunluk, hekimin radyografi incelemelerinde hatalı ya da eksik yorumlara neden olabilir. Bu koşullar göz önünde bulundurulduğunda çalışmada rutin diş hekimliği kliniğinde diş muayenesi ve çürük tespitinde en sık tercih edilen ve çok sayıda alınan bitewing radyografi görüntülerinin analizi yapılmasına karar verilmiştir.

2.1. AĞIZ İÇİ RADYOGRAFİ TÜRLERİ

Diş hekimliğinde kullanılan birden fazla radyografi görüntüleme türü mevcuttur. Bunlar hekimin ihtiyacına göre seçilip hastadan alınarak incelenmektedir. Örneğin,

bireyin kemik yapısı ve anatomik diş çerçevesi incelenmek istendiğinde panoramik radyografi röntgeni çekilmektedir. Bunun yanı sıra dişte bulunan çürük, implant, vb. gibi diş hastalıkları ve türleri incelenmek istendiğinde ağız içi radyografi röntgeni çekilmektedir. Radyolojik görüntüleme, diş görüntüleri için ağız içi radyografi ve ağız dışı radyografi olarak ikiye ayrılır. Ağız içine yerleştirilen cihaz ile ağız içi görüntüleme sağlanırken, tüm dişlerin görüntülenmesi için panoramik görüntüleme gibi ağız dışı görüntüleme sağlanır [6]. Ağız içi radyografik görüntüler, diş hekimleri tarafından yaygın olarak kullanılan bir görüntüleme türüdür. Ağız içi radyografi, film veya sensörün ağza yerleştirildiği radyolojik bir görüntü sağlar. Görüntülenecek dişlerin farklı yönlerini gösteren ısırma(bitewing), periapikal ve okluzal olmak üzere 3 çeşit ağız içi radyografi türü vardır. Diş sağlığı alanında ise detaylı bakılması gereken dişlerin tedavi ve tanı sürecinde bitewing ağız içi radyografi görüntülerinden yararlanılmaktadır. Bu radyografilerle daha fazla dişin kuru bölgesel olarak daha detaylı incelenmiş olur. Öte yandan ağız dışı radyografide ise röntgen cihazı ve görüntü sensörü hastayı dışarıdan göreceğ şekilde yerleştirilerek daha geniş bölgelerin görüntüleri ele alınır [7].



Şekil 2.1. Dental radyografi tipleri. a) Periapikal b) Bitewing c) Panoramik.

Ağız dışı radyografi türlerinin bazı örnekleri ise panoramik, sefalometrik ve bilgisayarlı tomografidir. Ekstraoral (ağız dışı) radyografilerde sadece dişler ön planda değil aynı zamanda çene ve kafatası da görüntülenmektedir. Bu nedenle radyografinin odak noktası sadece dişler değildir. Şekil 2.1’de sırasıyla periapikal, bitewing ve panoramik röntgen görüntüleri olmak üzere intraoral (ağız içi) ve ekstraoral (ağız dışı) röntgen örnekleri gösterilmektedir.

2.1.1. Periapikal Dental Radyografi

Ağız içi periapikal muayenenin amacı, tüm dişin ve çevresindeki yapıların bir görünümünü elde etmektir. Ağız içi periapikal radyografi, diş radyolojisinde yaygın olarak kullanılan bir ağız içi görüntüleme tekniğidir ve ağız içi periapikaller radyolojik muayenenin bir bileşenidir. Periapikal radyografiler (“peri”, “çevre” anlamına gelir ve “apikal, “diş kökünün ucu” anlamına gelir) dişlerin ve çevre dokuların ana hatlarını, konumunu ve meziodistal görünümünü yansıtır.

Periapikal radyografiler ile dişler ve dişleri çevreleyen kemik hakkında önemli bilgiler sağlanır. Bu filmler, yaygın diş hastalıklarının teşhisine yardımcı olmak için önemli bilgiler sağlar ve dişlerin çevresindeki alveolar kemiğiyle beraber tüm kuron ve kökün incelenmesini sağlar; özellikle diş çürüğü, diş apsesi ve periodontal kemik kaybı veya diş eti hastalığı bu tip radyografilerle tespit edilebilirler. Ayrıca restorasyonların durumu, gömülü dişler veya kırık diş parçaları, diş ve kemik anatomisindeki varyasyonlar gibi bulgular bu tip radyografilerle tespit edilebilir [8]. Şekil 2.2’de örnek bir periapikal röntgen görüntüsüne yer verilmektedir. Şekilde de görüldüğü üzere periapikal röntgen görüntüsünde belirli sayıda dişe yer verilmiştir. Böylelikle dişler ve dişi ilgilendiren hastalıklar kolaylıkla incelenmektedir.



Şekil 2.2. Örnek bir periapikal röntgen görüntüsü.

Ağız içi radyografiler, film veya dijital reseptörler kullanılarak elde edilirler. Periapikal ve bitewing görüntüleme türleri, tüm diş yapısının kapsamlı bir görünümü için çeşitli konfigürasyonlar oluşturmak üzere bir arada kullanılabilirler. Periapikal radyografiler tüm dişi ve destek dokuları görüntüleyerek çürükleri ve periodontal kemik kaybının boyutunu belirlemek ayrıca kök ve kemik patolojilerinin tanı ve tedavisine yardımcı olmak için kullanılır [2].

2.1.2. Okluzal Dental Radyografi

Okluzal röntgen filmi, birçok nedenden dolayı diş röntgen cihazı ile ağız bölgesinden ekstra görüntüler elde etmek için kullanılabilen çok yönlü bir filmidir. Şekil 2.3'te yer verildiği gibi okluzal incelemede filmin orta noktası ikinci premolar diş bölgesine denk gelecek şekilde ve filmin uzun eksenini ön-arka pozisyonda ve arka dişleri de içine alacak şekilde ağza yerleştirilir. Röntgen tüpünün konumu yatayda ve dikeyde belirli açılarla ayarlanarak x ışınları gönderilir ve istenen görüntüler elde edilir. [9].



Şekil 2.3. Okluzal filmin ağız içerisine yerleştirme adımları [9].

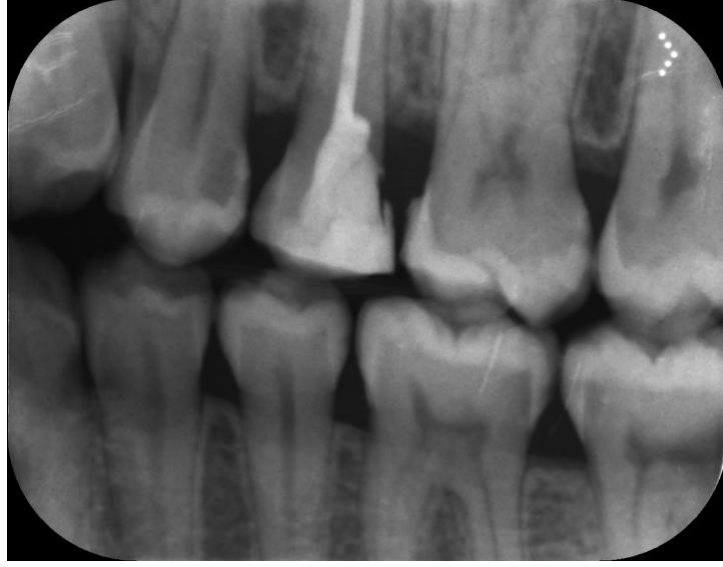


Şekil 2.4. Örnek bir okluzal dental radyografi görüntüsü [10].

Şekil 2.4'te görüldüğü üzere, okluzal bir radyografi görüntüsünde mandibular anterior bölgesinin incelenmesine olanak sağlanmaktadır. Bu radyografi türü, ortodonti ve çene fraktürlerinin tedavilerinde tercih edilirler. Bu filmlerde dişler ve çene kemikleri belirli açılardan incelenebilmektedir.

2.1.3. Bitewing Dental Radyografi

Bitewing radyografiler, ağız çene ve diş radyolojisinde yaygın olarak kullanılan bir ağız içi görüntüleme tekniğidir. İdeal tanısal görüntülerin oluşturulması her zaman kolay değildir hassas bir teknik ile elde edilmeleri gerekir. Bitewing radyografiler, maksiller ve mandibular dişlerin kuronlarını gösteren, dişlerin interproksimal yüzeylerinin net bir görüntüsünü sağlayan ve interproksimal çürüklerin tespitine izin veren bir görüntüdür. Eşzamanlı olarak, maksiller ve mandibular alveolar kretler görüntülenerek, seviyelerinin değerlendirilmesine izin verilir ve periodontal durumun değerlendirilmesine katkıda bulunur. Restorasyonların durumu ve diş taşının varlığı dahil olmak üzere bitewing radyografisinde ek önemli bulgular tespit edilebilir [11]. Şekil 2.5'te örnek bir bitewing dental radyografisine yer verilmektedir. Şekilde görüldüğü gibi sadece belirli sayıda dişler röntgen görüntüsünde yer almaktadır.



Şekil 2.5. Örnek bir bitewing dental radyografi görüntüsü.

Bitewing radyografiler, proksimal çürük lezyonlarının erken saptanmasına yardımcı olan bir görüntüleme yöntemidir. Ancak sadece mine tabakasının etkilendiği erken proksimal lezyonların saptanması için bazı durumlarda tek başına yetersiz kalabilir. Bu, çok erken lezyonların (diş minesinde histolojik olarak ilerlemiş) genellikle fark edilmeden kalacağı varsayımına yol açar. Daha ileri dentin veya kavitasyonlu lezyonların tespitinde ise daha hassas bir görüntüleme sağlar. Bitewing radyografilerinin özgülüğü yüksektir (0,9 civarında veya üzerinde), bu durum da tespit edilen lezyonların sadece yaklaşık %10'unun aslında sağlam yüzeyler olabileceği söylenebilir. Bitewing radyografisi genellikle uzun süreli hasta takibi sebebiyle kullanıldığı için 12-24 ayda bir tekrarlı muayeneler gerektirir. Bu nedenle, takip eden her muayenede diş kaybı riski azalır [12].

2.2. BITEWING DENTAL VERİ KÜMESİ VE VERİ ETİKETLEME

Radyolojik inceleme diş hekimliği pratiğinde önemli bir yere sahiptir ve ağız içi görüntüleme sıklıkla kullanılmaktadır. Bu amaçla kliniklerde periapikal ve bitewing filmleri rutin olarak kullanılmaktadır [13,14]. Radyografilerde diş durumunun görüntülenmesi ve yorumlanması, hastalık teşhisinde en önemli adımlardır [14]. Neredeyse tüm kayıtların dijital olduğu hastane bilgi sistemlerinde, hekimlerin diş kayıtlarını sisteme düzenli geçirmesi gerekmektedir [15]. Hekimlerin tecrübesine ve

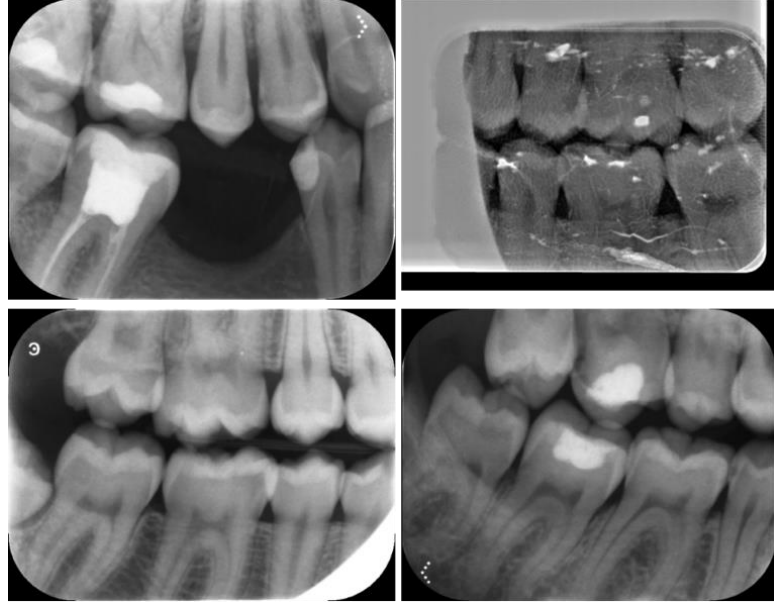
dikkatine bağı olarak yoğun kliniklerde yanlış veya yetersiz teşhis ortaya çıkabilmektedir. Bunu önlemek için dişlerin doğru bir şekilde tanımlanması ve numaralandırılması gerekir. Radyografilerde dişlerin tespiti ve numaralandırılması da adli incelemeler için önemlidir [16]. Doktorlara medikal ve dental görüntülemeye yardımcı olmak için bilgisayar destekli sistemler geliştirilmiştir [15,17]. Dişler, uzman bilgisine dayalı açık parametrelerle çeşitli mühendislik yöntemleri ile dental filmlerden tanımlanmıştır [18, 19].

Ağız içi radyografik görüntü türlerinden olan periapikal ve bitewing radyografisi tıbbi görüntülemeye önemli rol oynamaktadır. Okluzal radyografik görüntüleme, dişlerin incelenmesi için rutin tercih edilen bir görüntüleme türü değildir. Bu nedenle görüntü üzerinde sadece belirli dişlerin yer aldığı ve diş tiplerinin çiğneyici yüzeylerinden görüntülediği okluzal radyografiler bu tez çalışması için uygun bulunmamıştır. Bitewing radyografiler ise periapikal radyografiye göre daha fazla diş içermeleri, yüksek çözünürlüklü olmaları nedeniyle bu tez çalışmasında bu tip radyografilerden oluşan veri kümesi kullanılmıştır.

Bu tez çalışmasında kullanılan bitewing radyografileri, Kodak CS 2200 (Carestream Dental, Atlanta, GA) 220–240-V periapikal dental görüntüleme ünitesi kullanılarak 60 kVp, 7 mA ve 0.1 s aşağıdaki parametrelerle elde edilmiştir.

2.2.1. Bitewing Dental Veri Kümesi

Çalışma kapsamında kullanılan bitewing ağız içi röntgen görüntüleri ile ilgili Ordu Diş Hekimliği Fakültesi Ağız, Diş ve Çene Radyolojisi Anabilim Dalından gerekli yasal izinler alınmıştır. Çalışmada kullanılan veri kümesinde toplam 1200 adet bitewing ağız içi röntgen görüntüsü bulunmaktadır. Ağız içi bitewing görüntülerinin etiketlenme aşamasına gönderilmeden önce eğitim ve test kümesi olarak ayrılması gereklidir. Verilerin etiketlenmesi sonucu modele verilecek olan veriler makine öğrenimi modeli ile okunabilir hale gelmektedir. 1200 adet görüntünün rastgele seçilmiş 1000 tanesi eğitim kümesine ayrılırken rastgele seçilmiş 200 adet görüntü ise modelin doğruluğunun test edilmesi için gerekli olan test kümesine ayrılmıştır. Şekil 2.6'da çalışmada kullanılan veri kümesinden örnek görsellere yer verilmektedir.

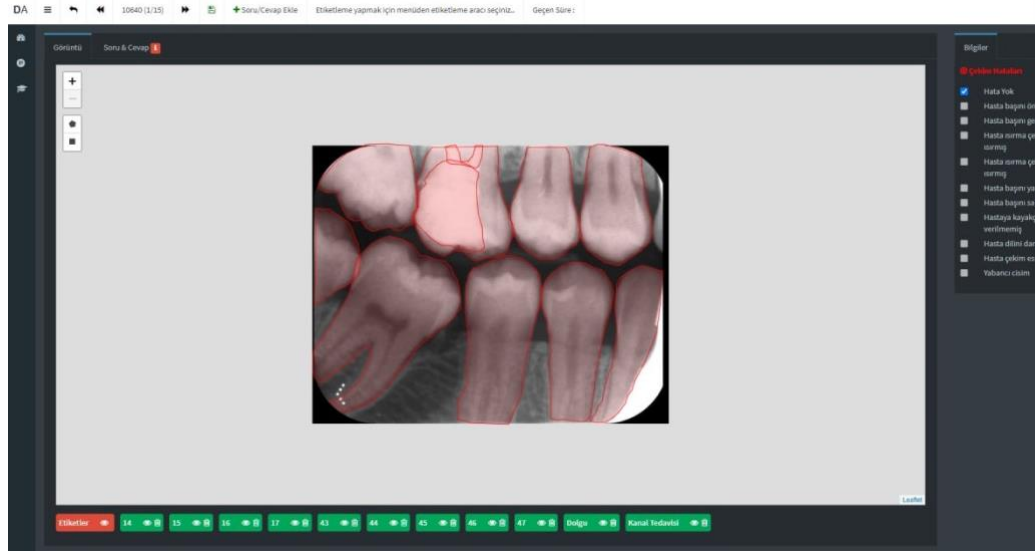


Şekil 2.6. Veri kümesinden alınan örnek bitewing radyografi görüntüleri.

Şekil 2.6’da yer alan veriler incelendiğinde, veri kümesinde dağılmış tüm verilerin rastgele seçildiği ve temiz veri olmadığı anlaşılmaktadır. Bunun sebebi, sinir ağı modelinin test amacıyla alışılmamış bir veri ile karşılaştığı durumda yüksek doğruluk ile dişleri tespit etme amacıdır.

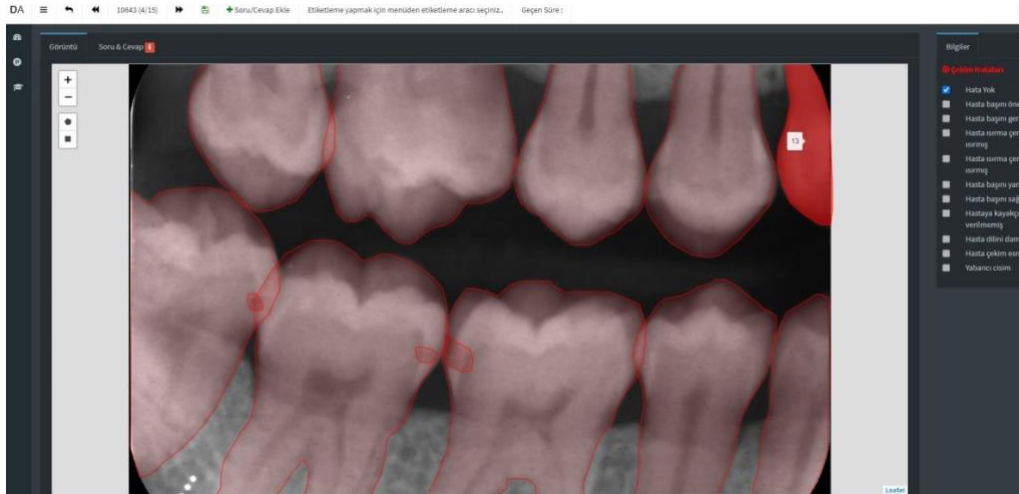
2.2.2. Bitewing Veri Etiketleme

Verilerin etiketlenmesi gözetimli herhangi bir makine öğrenimi ve derin öğrenme projelerinde çok önemli bir adımdır. Bir sinir ağı veya makine öğrenimi modelinin tahmin mekanizmasının yürütülmesi için verilerin gerçek referans adını verdikleri etiketlenme işleminin yürütülmesi gerekmektedir. Veri etiketleme, bir görüntü, metin veya ses kaynağında yer alan nesnelerin ve bilgilerin tanımlanması ve bu bölgeler için hangi nesneye ait ise o nesneye ait ek açıklamaların eklendiği işlemidir. Bu tez çalışmasında, DentiAssist [20] veri etiketleme platformu kullanılmıştır. Etiketleme ekranı Şekil 2.7’de gösterilmektedir.



Şekil 2.7. Veri etiketleme ara yüz ekranı [20].

Bitewing ağız içi görüntülerinde var olan dişler bir dikdörtgen olarak değil poligon nesnelere olarak etiketlenmiştir. İçerik bölümünde ilgili görselin gösterildiği, etiketlemenin yapıldığı ve görsel ile ilgili bilgilerin görüntülediği iki bölüm bulunmaktadır. Kullanıcı, etiketleme yaparken görüntünün sol tarafında bulunan işaretleme araçlarından birini seçmektedir. Bu araçlar, dikdörtgen ve poligon adı verilen çoklu çizgi seçimleri ile seçimlere izin verir.

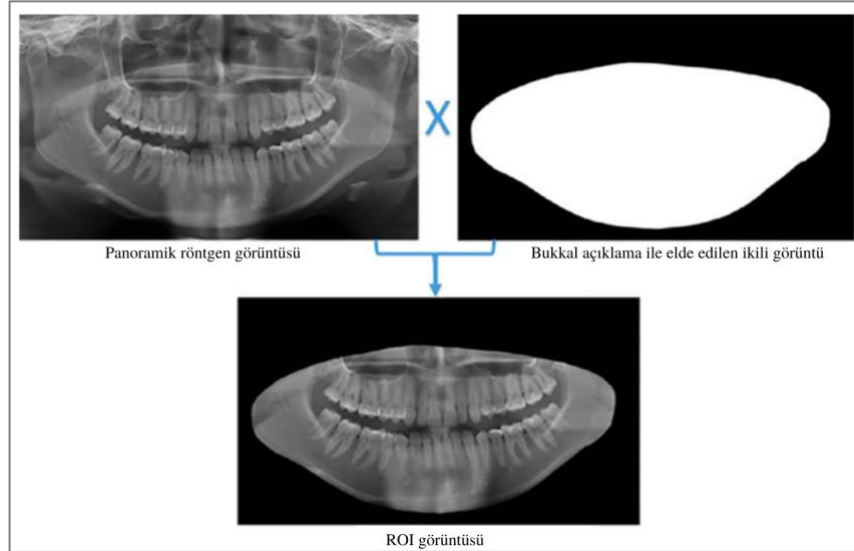


Şekil 2.8. Detaylandırılmış veri etiketleme paneli [20].

Poligon nesnelere olarak etiketlenmesinin en büyük sebebi, dişlerin anatomik olarak insan ağızında fazla girinti çıkıntıya sahip olmasından kaynaklanmaktadır. Şekil 2.8’de ise etiketlenen diş nesnelere daha detaylı görüntüsü yer almaktadır.

Tez çalışmasında kullanılacak olan R-CNN ağlar için radyografik görüntülerde gerçekleştirilen çalışmalar incelenirken bunun yanı sıra yalnızca sınıflandırma veya segmentasyon yapılan çalışmalar da incelenerek kapsamlı bir literatür taraması yapılmıştır. Nesne tespitinde bölge tabanlı evrişimli sinir ağı (R-CNN) yaklaşımı, nesne olma olasılığı yüksek olan bölümü değiştirmek [21, 22] ve evrişimli ağları değerlendirmek [23]. Bu algoritmalar, tıbbi görüntü sınıflandırması ve tespiti için yaygın olarak kullanılmaktadır.

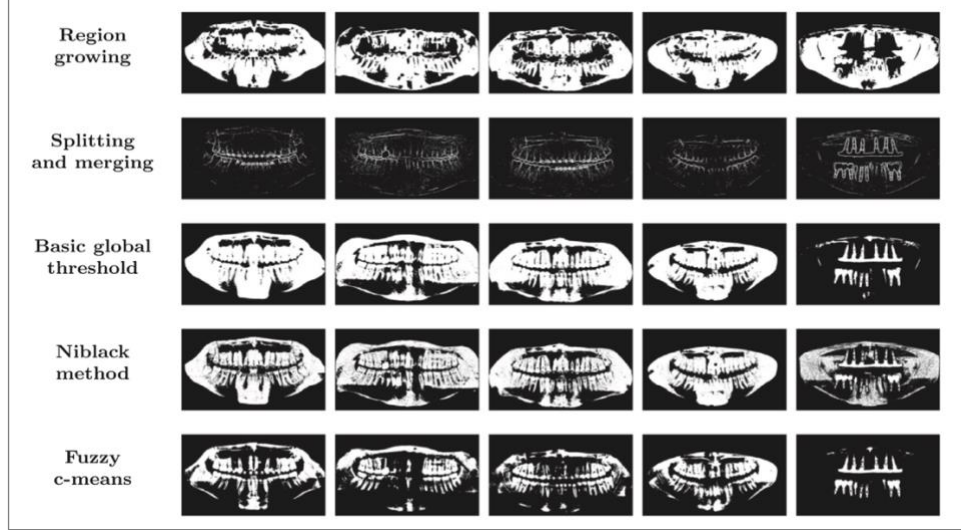
Silva vd. [24]’nin yaptıkları çalışmada, panoramik röntgen görüntülerinde her dişi segmentasyon maskelerine ayırmaya yönelik klasik yöntemlerin kapsamlı bir literatür incelemesini sunulmuştur. 1500 görüntü kullanılarak yapılan çalışma sonucunda Mask R-CNN [25] çözümünün dental görüntülerde kullanılan klasik yöntemlerden (bölge bazlı, eşik bazlı, küme bazlı, sınır bazlı) daha iyi olduğu kanıtlanmıştır.



Şekil 2.9. Görüntülerin ROI alanlarının belirlenmesi [24].

Bu çalışmada doğruluk metriklerinin hesaplanması, açıklamaları eklenmiş görüntü ile segmente edilmiş görüntü ilgi bölgesini (ROI) karşılaştırarak piksel piksel

doğrulayarak gerçekleştirilir. Şekil 2.9’da gösterilen aynı işlem izlenerek, analiz edilen her bir segmentasyon yöntemiyle segmente edilen tüm görüntüler üzerinde gerçekleştirilmiştir [24]. Panoramik görüntüde yer alan ağız ve diş bölümü ikili görüntüye alınarak ağızdaki bukkal boşluk başarıyla tespit edilmiştir.



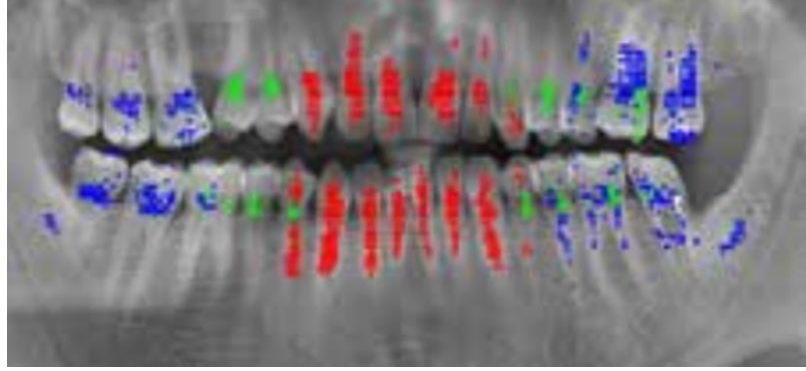
Şekil 2.10. Klasik görüntü işleme yöntemleri ile segmentlere ayrılmış kategorilerden görüntü örnekleri [24].

Şekil 2.10’da ise Silva vd. tarafından ortaya konulan çalışmada, klasik görüntü işleme yöntemlerinden bazılarında ait örnek görsellere yer verilmektedir. Böylelikle diş nesnelere anlamlandırılırken hangi işleme yönteminin daha performanslı çalıştığına karar verilebilmektedir. Bu çalışmada birbirinden farklı 10 adet kategori yer almaktadır. Bu kategorilerden bazılarında örnek olarak şunlar verilebilir:

- Restorasyonlu ve tıbbi cihazlı dişleri içeren tüm dişleri içeren görüntüler
- Diş implantı içeren görüntüler
- Eksik dişler, restorasyonsuz ve diş cihazı olmayan dişleri içeren görüntüler

Oktay [26] tarafından yapılan çalışmada, ağız boşluğu belirlendikten sonra diş tiplerinin klasik yapay sinir ağları ile tanımlanması ortaya çıkarılmıştır. Ayrıca sadece diş tipleri belirlenmiş olup, diş numaralandırma sistemi kullanılmamıştır. İlk olarak ağız boşluğunu bulduktan sonra, üç diş tipi için (kesici dişler, küçük azılar ve büyük azılar) uygun pozisyonlar belirlenmiştir. Dişler, çoklu sınıflandırmanın yapıldığı AlexNet

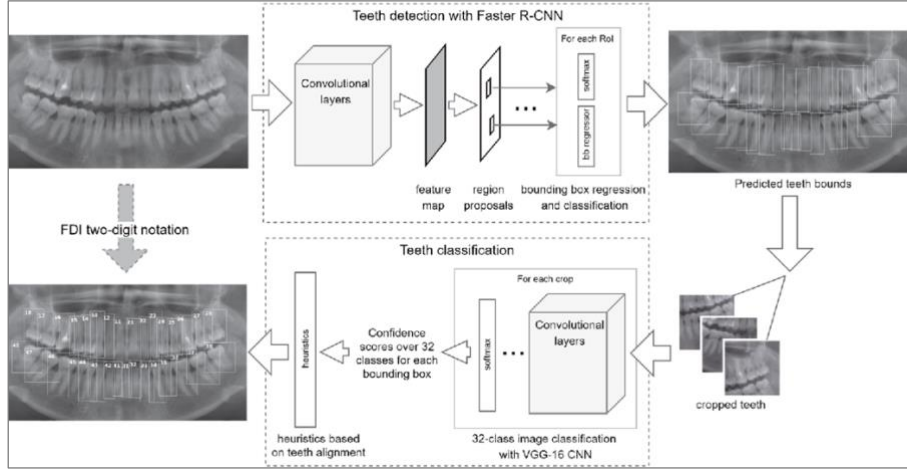
mimarisinin deęiştirilmiş bir versiyonu ile tespit edilir. Yapılan testler, yöntemin doğruluğunun ümit verici olduğunu ve sunulan yöntemin bilgisayar destekli uygulamaların ilk adımı olarak kullanılabileceğini göstermiştir [26].



Şekil 2.11. Tespit edilen diş konumlarının merkezleri [26].

Şekil 2.11’de tespit edilen pencerelerin merkezleri gösterilmiştir. Kesici dişler ve kaninler kırmızı, küçük azılar yeşil ve azı dişleri mavi ile gösterilmiştir. Tespit edilen diş merkezleri diş kökünde ortalanmış görünmektedir. Bazı yanlış algılanan diş konumları 3. molar diş için özellikle komşu dişlerde ve diş etlerinde bulunmaktadır. Komşu olan kanin ve premolarların birbirine çok benzemesi ve premolar olarak yanlış sınıflandırılmaları nedeniyle premolarların doğruluğu diğer diş tiplerine göre daha düşüktür [26].

Bir sonraki çalışmada ise Tuzoff vd. [27], Faster R-CNN [28] ile panoramik röntgen görüntülerinde diş tespiti ve numaralandırma gerçekleştirilmiştir. Yetişkinlerin panoramik görüntülerinden rastgele seçilen 1352 görüntüyle Faster R-CNN ağı eğitilmiştir. Dişlerin uzamsal düzenine ilişkin kurallara göre sonuçları iyileştirmek için sezgisel algoritma ile birlikte klasik VGG-16 [28] CNN’yi kullanılmıştır [27].



Şekil 2.12. Faster R-CNN ile diş tespiti ve numaralandırma akış şeması [27].

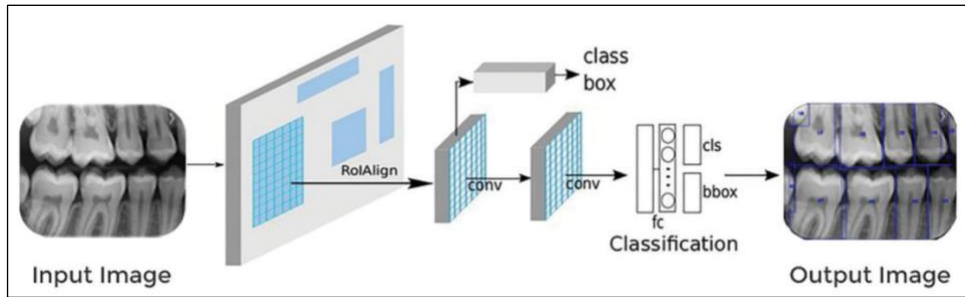
Şekil 2.12’de yer alan akış şemasında yer alan diş tespit modülü, her bir dişin sınırlarını tanımlamak için radyografiyi işlemektedir. Diş numaralandırma modülü ise kırılan her bölgeyi numaralandırma notasyonuna göre sınıflandırır, tüm dişleri birleştirir ve son diş numaralarını üreten sezgisel yöntemi uygulamaktadır.

Muresan vd. [29] panoramik görüntülerde diş problemlerini görüntü işleme teknikleri ve derin öğrenme ile sınıflandırmıştır. Daha sonra görüntü segmentasyonun gelişmesiyle birlikte Leite ve ark. [30], 153 adet radyografik görüntüyü bölerek diş nesnelerini sınıflandırmıştır. Poonsri vd. [31] ise diş röntgeni görüntülerinde şablon eşleştirme ile sezgisel bir yaklaşım sergileyerek diş segmentasyonunu gerçekleştirmiştir. Bu çalışmada görüntü işleme teknikleri ile diş bölgesi tanımlanmış ve ardından dişler arka plandan ayrı olarak segmentlere ayrılmıştır. Böylece diş hekimliğinde yapay zeka çalışmaları kapsamında diş ve diş olmayan alanların ve problemlili diş ve diş hastalıklarının segmentasyonu yapılmaktadır. Diş hekimliği alanında kullanılan yapay zeka uygulamaları sonucunda hasta kapasitesi ve muayene süresi optimize edilmiştir. Zhu vd. [32] röntgen olarak çekilmeyen ağız içi görüntülerinde Mask R-CNN ağı ile dişlerin tespiti ve segmentasyonu gerçekleştirilmiştir. Mask R-CNN’nin karmaşık ve kalabalık diş yapılarında da iyi bir segmentasyon etkisine sahip olduğunu göstermektedir [32].



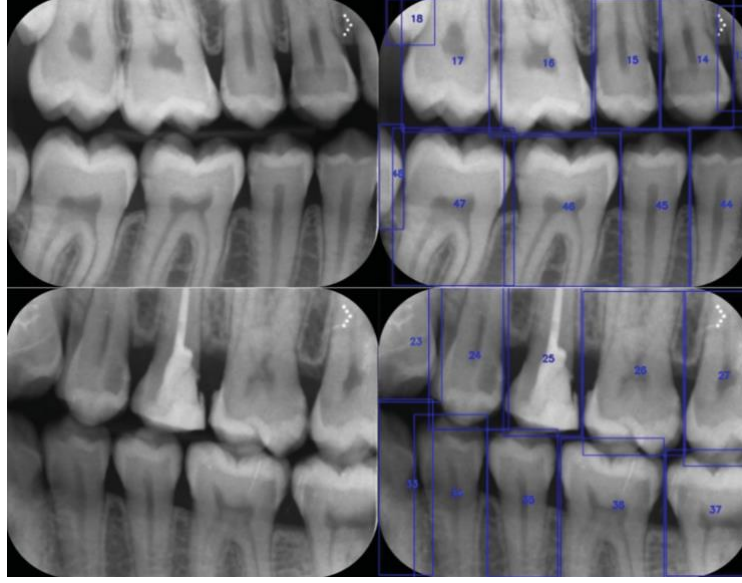
Şekil 2.13. Diş segmentasyonundan alınan test sonuçları. a) en iyi sonuç b) en kötü sonuç [32].

Şekil 2.13'te test sonuçlarından en iyi ve en kötü örnek görsellerinde segmentasyon maskelerine yer verilmiştir. Çalışmada kullanılan diş görüntülerinde az sayıda protez örneği olduğundan, tespit protezin doğruluğu düşüktür [32]. Bu çalışmalara ek olarak, literatürde yer alan bitewing radyografik röntgen görüntülerinde çalışmalar mevcuttur. Yasa vd. [33] tarafından ortaya konulan çalışmada bitewing radyografik görüntüler kullanılmıştır. Bu çalışmada Faster R-CNN kullanılarak bitewing görüntülerinde diş numaralandırması için otomatik bir algılama sistemi önerilmiştir.



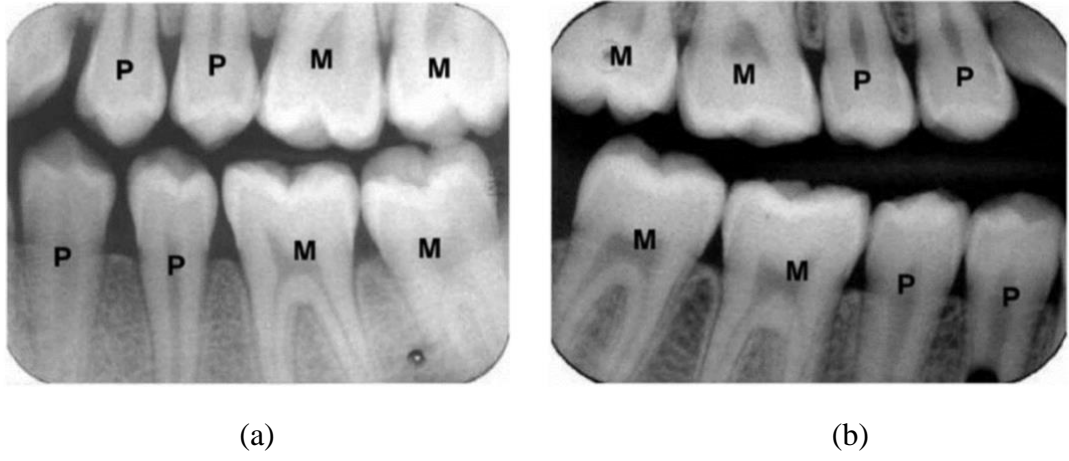
Şekil 2.14. Diş tespiti ve numaralandırma sistem mimarisi [33].

Şekil 2.14'te verilen diş tespiti ve numaralandırma mimarisinde adım adım Faster R-CNN ağı çalıştırılarak sınıflandıma gerçekleştirilmiştir. Çalışmada kullanılan 109 adet bitewing görüntüsündeki 715 diştten 697 tanesi doğru bir şekilde numaralandırılmıştır.



Şekil 2.15. Derin evrişimli sinir ağı sistemi kullanarak bitewing radyografilerinde dişlerin numaralandırılması [33].

Şekil 2.15'te çalışma sonucunda elde edilen bitewing radyografilerdeki diş sınıfları görülmektedir. Sol tarafta yer alan röntgen görüntüleri orijinal görüntüler ile temsil edilirken sağ tarafta mavi kutucuklar ise sınıflara ait sınırlayıcı kutular ve isimlerine aittir. Bitewing görüntülerinin analizine yönelik bir CNN yaklaşımı, dişlerin saptanması ve numaralandırılması için umut vaat etmektedir. Bu yöntem, diş çizelgelerini otomatik olarak hazırlayarak diş hekimlerine zaman kazandırılmaktadır. Bu çalışmada dişler, bitewing görüntülerinde sınıflandırmak ve sınıfların atanması için bir algoritma sunulmuştur. Çalışmada bitewing görüntüsünde azı dişleri ve diş hekimliğinde kullanılan ortak numaralandırma sistemine dayalı olarak her dişe bir mutlak sayı atamak için Bayes sınıflandırması kullanılmıştır. Çalışma sonucunda 400'den fazla diş içeren 50 bitewing görüntüsü ile yapılan deneyler yöntemin dişlere mutlak indeks sayısını yüksek doğrulukla sınıflandırma ve atama konusunda yetersiz olduğu gösterilmiştir [34].



Şekil 2.16. Bitewing görüntülerinde dişlerin düzenlenmesi. a) sol kadran ve (b) sağ kadran, M: Molar, P: Premolar [36].

Şekil 2.16’da verilen bitewing görüntülerinde var olan dişlerin sınıfları tespit edilmiştir. Şekilde de bahsedildiği gibi M molar ve P premolar sınıflarını temsil etmektedir. Molar dişler radyolojide azı diş anlamına gelirken premolar dişler ise küçük azıları ifade etmektedir. Çalışma sürecindeki deneyler, premolarların sınıflandırılmasının her iki çene için azı dişlerinden daha zor olduğunu göstermektedir [34].

BÖLÜM 3

EVRIŞİMLİ SİNİR AĞLARI VE MASK R-CNN

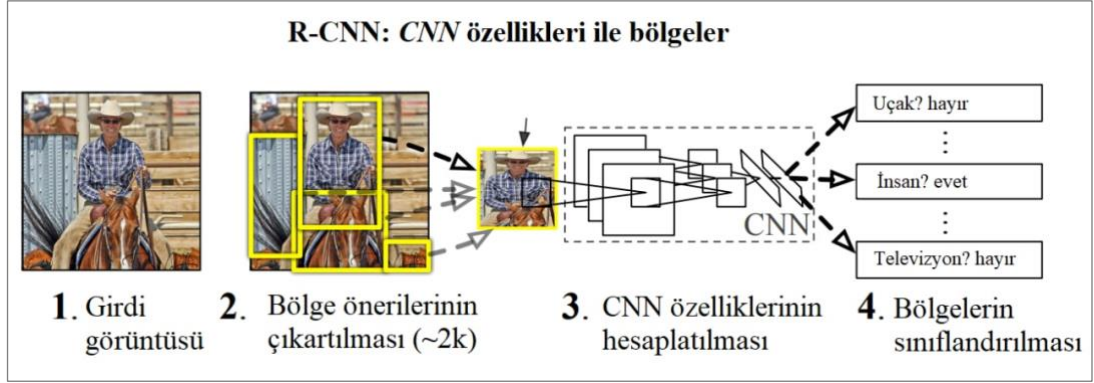
Derin öğrenmede, evrişimli bir sinir ağı (CNN), en yaygın olarak görsel görüntüleri analiz etmek için uygulanan bir yapay sinir ağı sınıfıdır [35]. Bu tez çalışmasında, etiketlenen gerçek referans değerli görüntülerde var olan dış nesnelere tespit edilmesi için yapay sinir ağları kullanılmıştır. Yapay sinir ağları, birden fazla katmanı içinde barındıran girdinin veri kümesi olarak modele verildiği ve çıktı olarak ise tahmin sonuçlarının elde edildiği ağ modelidir. Kullanılan yapay sinir ağı mimarisi olarak ise Mask R-CNN seçilmiştir.

3.1. EVRIŞİMLİ SİNİR AĞLARI (CNN)

CNN, özellikle görüntüler için derin öğrenme alanında sıklıkla kullanılan bir yapay sinir ağıdır. Bu sinir ağı, görüntüleri çeşitli belirlenmiş katmanlarda inceleyerek ayrıntılı olarak işler ve sınıflandırır. Bununla birlikte, genellikle yüksek başarı seviyelerine ulaşmakta zorluk çeken sinir ağıdır. Ek olarak, girdi özellikleri boyunca özellik haritaları olarak bilinen evrişim filtrelerinin paylaşılan ağırlık mimarisine dayanan yapay sinir ağları olarak da bilinirler [36,37].

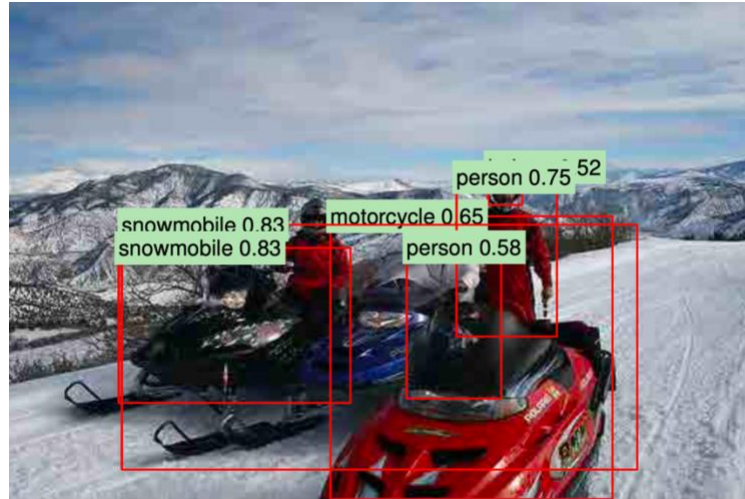
3.1.1. R-CNN

Bölge tabanlı Evrişimli Sinir Ağı (R-CNN) [38], nesne önerilerini sınıflandırmak için derin bir evrişimli sinir ağı kullanarak yüksek nesne algılama doğruluğu sağlamaktadır. R-CNN ağların amacı, nesne tespit problemini çözmektir. R-CNN ağı bir görüntüyü örnek olarak aldığı anda, tüm nesnelere üzerine sınırlayıcı kutular çizmektedir. Süreç boyunca, bölge önerisi adımı ve sınıflandırma adımı olmak üzere iki genel bileşene ayrılmaktadır.



Şekil 3.1. Nesne algılama sistemine genel bakış [38].

Şekil 3.1’de verilen akış şemasında R-CNN, seçici arama özelliğini kullanmaktadır. Şekildeki ikinci adımda bahsedilen bölümde R-CNN, nesne olması muhtemel bölgeleri en yüksek 2000 farklı bölge için seçici arama işlemini kullanmaktadır. Bir dizi bölge önerisi bulduktan sonra, bu öneriler daha sonra her bölge için bir özellik vektörü çıkaran eğitilmiş bir CNN’ye beslenebilecek bir görüntü boyutuna çarpılır ve Destek Vektör Makineleri (DVM)’ne girdi olarak verilir.



Şekil 3.2. R-CNN modeli örnek bir görüntü verisinde sonuçlar [38].

Şekil 3.2’de doğrulama klasörüne ait olan veriler üzerinde gerekli konfigürasyon değerleri seçilerek %31 mAP değerine ulaşılmıştır. Yukarıda bahsedilen yarışmalar ve veri setleri üzerinde her görüntü rastgele 0,5’ten büyük hassasiyetteki tüm algılamalar gösterilmiştir. Her algılama, tahmin edilen sınıfla ve dedektörün hassas geri çağırma eğrisinden bu algılamamanın kesinlik değeriyle etiketlenmektedir [38].

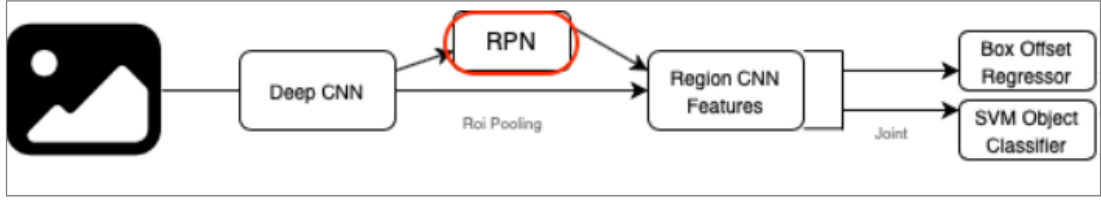
Çizelge 3.1. VOC 2011’den alınan segmentasyon sonuçları [39].

VOC 2011 val	arka plan	uçak	bisiklet	kuş	gemi	şişe	otobüs	araba	kedi	sandalye	inek
O ₂ P	84,0	69,0	21,7	47,7	42,2	42,4	64,7	65,8	57,4	12,9	37,4
full R-CNN f _{c6}	81,3	56,2	23,9	42,9	40,7	38,8	59,2	56,5	53,2	11,4	34,6
full R-CNN f _{c7}	81,0	52,8	25,1	43,8	40,5	42,7	55,4	57,7	51,3	8,7	32,5
fg R-CNN f _{c6}	81,4	54,1	21,1	40,6	38,7	53,6	59,9	57,2	52,5	9,1	36,5
fg R-CNN f _{c7}	80,9	50,1	20,0	40,2	34,1	40,9	59,7	59,8	52,7	7,3	32,1
full+fg R-CNN f _{c6}	83,1	60,4	23,2	48,4	47,3	52,6	61,6	60,6	59,1	10,8	45,8
full+fg R-CNN f _{c6}	82,3	56,7	20,6	49,9	44,2	43,6	59,3	61,3	57,8	7,7	38,4

R-CNN için nesne algılama sistemi üç modülden oluşmaktadır. Birincisi, kategoriden bağımsız bölge önerileri üretir. Bu öneriler, tespit edici için mevcut olan bir dizi aday tespitini tanımlar. İkinci modül, her bölgeden sabit uzunlukta bir öznitelik vektörü çıkaran büyük bir evrişimli sinir ağıdır. Üçüncü modül ise sınıfa özgü bir dizi doğrusal DVM’dir [38]. R-CNN ağı için başarımleri PASCAL VOC 2010-12 [39] ve ILSVRC2013 [40] veri setinde ve yarışmalarında tespit sonuçlarında gösterilmektedir. Çizelge 3.1’de görüldüğü üzere arka plan, bisiklet, kuş gibi görüntü verilerinde R-CNN ile tespit edilen nesnelere doğruluk puanları yer almaktadır. Böylelikle hangi sınıfa ait olan nesnenin daha iyi sonuçlar verdiği gözle görülmektedir. Evrişimli sinir ağları, zaman içerisinde hızlı bir şekilde Fast R-CNN [41] ve Faster R-CNN [30] olarak yeni bir dağıtım olarak ilerlemiştir. Bunun sebebi ise R-CNN ağlarının performanslı ancak oldukça yavaş çalışmasından kaynaklanmaktadır. Bu sebeple, R-CNN türlerinden olan Mask R-CNN [25] ise Faster R-CNN ağının türetilmiş farklı bir modelidir.

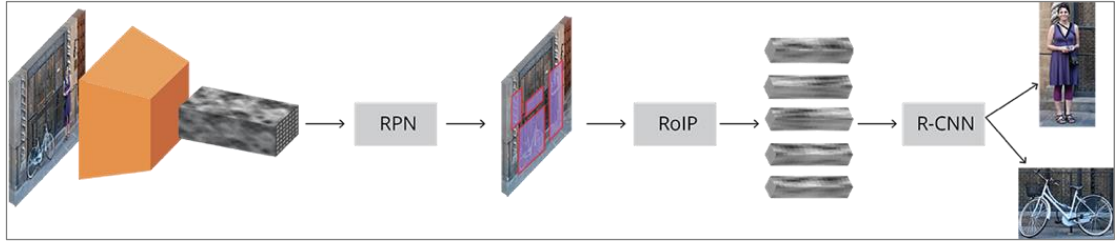
3.1.2. Faster R-CNN

Faster R-CNN, bir görüntüyü girdi olarak alır ve bu görüntü için özellik haritalarını döndüren evrişimli sinir ağına iletir. Bu özellik haritalarında Bölge Teklif Ağı (RPN) işlemi uygulanır [42]. Böylelikle objektiflik puanı ile nesne olma ihtimali teklifler sinir ağı modeline döndürülür.



Şekil 3.3. Faster R-CNN iş akış diyagramı.

Şekil 3.3’de verilen iş akış grafiğinde Bölge Teklif Ağı (RPN) uygulanırken tüm tekliflerin aynı boyuta getirilmesi için bu tekliflere İlgi Bölgesi Havuzlama (RoI Pooling) katmanı uygulanmaktadır.



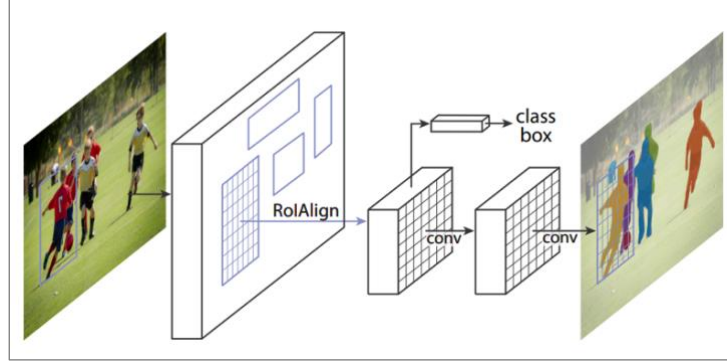
Şekil 3.4. Faster R-CNN örnek bir görsel üzerinde akış mimarisi [42].

Şekil 3.4’te verilen mimaride Faster R-CNN ağı için orijinal bir görüntünün sırasıyla RPN’den ve ROI Pooling yönteminden geçirilmesi gösterilmektedir. Bu mimarinin sonucunda bölge olması beklenen nesnelere ayrılmıştır. Faster R-CNN, içeriğinde bilgi olarak sınırlayıcı kutuların listesini, etiketlerini ve olasılık değerlerini tutar. Faster R-CNN, mimari içerisinde modele bölge teklif ağı kullanarak Fast R-CNN için hızlanma sağlamıştır ve modern nesne algılamanın temel taşı haline gelmiştir. Daha sonra, bazı çalışmalar ile bu algılayıcı ağı çeşitli detay problemlerini ele almak için genişletmiştir [43]. Genişletmenin yanı sıra yüksek kalitede segmentasyon maskeleri de elde edecek şekilde Mask R-CNN [25] sinir ağı elde edilmiştir.

3.1.3. Mask R-CNN

Mask R-CNN, sınıflandırma ve sınırlayıcı kutu regresyonu için mevcut dala paralel olarak, her bir ROI’de segmentasyon maskelerini tahmin etmek için Faster R-CNN’yi genişletir. Maske dalı, her bir ROI’ye uygulanan ve pikselden piksele bir şekilde bir segmentasyon maskesini öngören küçük bir tam bağı katmandır. Mask R-CNN, çok

çeşitli esnek mimari tasarımlarını kolaylaştıran Faster R-CNN çerçevesi göz önüne alındığında, uygulanması ve eğitilmesi kolaydır. Ek olarak, maske dalı yalnızca küçük bir hesaplama yükü ekleyerek hızlı bir sistem ve hızlı deney yapılmasını sağlar [25].



Şekil 3.5. Mask R-CNN iş akış mimarisi [25].

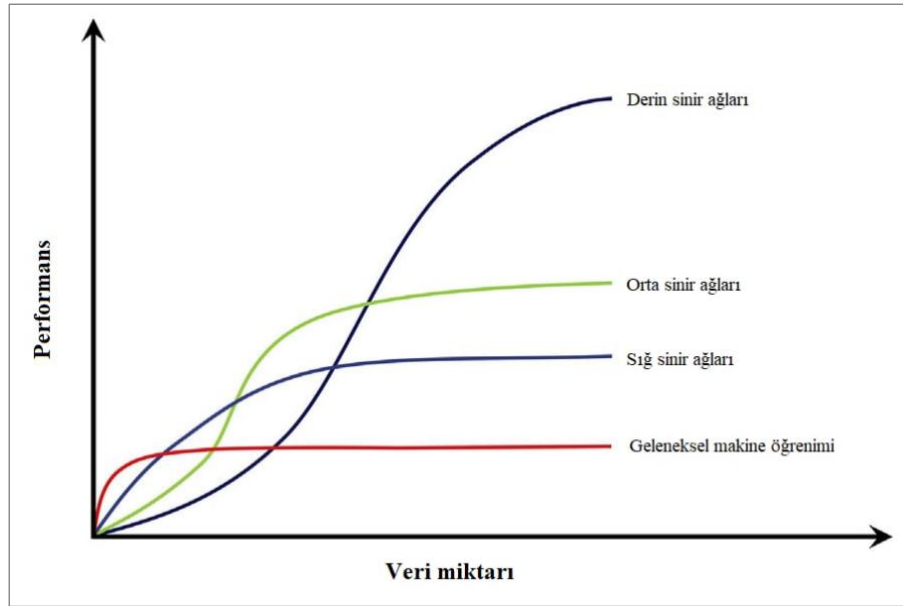
Şekil 3.5'te görüldüğü gibi örnek bir görüntü verisi sinir ağına verildiğinde İlgi Bölgesi Hizalama (ROI Align) yöntemi ile nesnelere hizalanması sağlanmaktadır. Şekildeki gibi evrişim filtrelerinden geçirilirken özellik haritaları çıkartılarak sınıf nesnelere belirlenmektedir. Sınıf nesne ismine ve puanına ek olarak çıktıda yüksek kalitede segmentasyon maskeleri oluşturulmaktadır. Prensipinde Mask R-CNN, Faster R-CNN'nin sezgisel bir uzantısıdır, ancak iyi sonuçlar için maske dalını doğru şekilde oluşturmak çok önemlidir. En önemlisi, Faster R-CNN, ağ girişleri ve çıkışları arasında pikselden piksele hizalama için tasarlanmamıştır. Bu en çok, örneklerle ilgilenmek için fiili çekirdek işlemi olan ROI Pooling yöntemi, özellik çıkarımı için kaba uzamsal nicelemeyi nasıl yaptığında belirgindir [43].

Faster R-CNN'nin bir uzantısı olan Mask R-CNN ağı, örnek segmentasyon görevini gerçekleştirmek için evrişimli ağların bir dalını içerir. Mask R-CNN ağı, omurga yapısında önceden eğitilmiş MS COCO [44] ağırlıklarını kullanmaktadır. ResNet-101'den özellikleri çıkardıktan sonra, bu özellikler bir Özellik Piramit Haritası (FPN) oluşturur; burada nihai olarak ankorlar tanımlanır ve ilgili bölgeler çıkarılır [45]. Mask R-CNN, örnek segmentasyon görevini gerçekleştirmek için evrişim ağlarının bir dalını içerir. Bu dal, bir özellik çıkarıcı olarak hizmet veren standart bir evrişimli sinir ağıdır. Ayrıca, çok ölçekli algılama gerçekleştirmek için, omurga ağında bir FPN kullanılır. FPN, birinci piramitten üst düzey özellikleri alan ve bunları alt katmanlara geçiren

ikinci bir piramit ekleyerek standart özellik çıkarma piramidini geliştirir [45]. Mask R-CNN, örnek segmentasyon görevini gerçekleştirmek için bir özellik çıkarıcı olarak işlev gören standart bir evrişimli sinir ağı olan evrişimli ağların bir dalını içerir. Bu tez çalışmasında, yapay sinir ağları yapısı oluşturulurken Mask R-CNN ağı kullanılmasına karar verilmiştir. Omurga ağı olarak ise Derin Artık Ağlar [46] gibi görüntü özelliklerini çıkaran bir ağ tercih edilmiştir. Prensip olarak, bu omurga ağı, ResNet-50 veya ResNet-101 [46] gibi görüntü özelliklerini ayıklayan herhangi bir ağ olabilir.

3.2. DERİN ÖĞRENME KÜTÜPHANELERİ

Makine öğrenmesi (MÖ), çalışılacak projeye ait ihtiyaç olan veri setini toplayarak temel istatistiksel bir modelin eğitilmesini tanımlama sürecidir. Derin öğrenme ise makine öğreniminin bir alt türüdür. Derin öğrenme, özellik çıkarma ve dönüştürme için birçok doğrusal olmayan işlem birimi katmanını kullanmaktadır. Her ardışık katman, önceki katmandaki çıktıyı girdi olarak alır [47].



Şekil 3.6. Mevcut verilerin geleneksel makine öğrenimi algoritmalarının performansı üzerindeki etkisini gösteren grafik [48].

Şekil 3.6’da geleneksel makine öğrenimi algoritmalarının ve yapay sinir ağlarının verilen veri miktarına karşılık performans etkileri gözlemlenmektedir. Bu durum göz

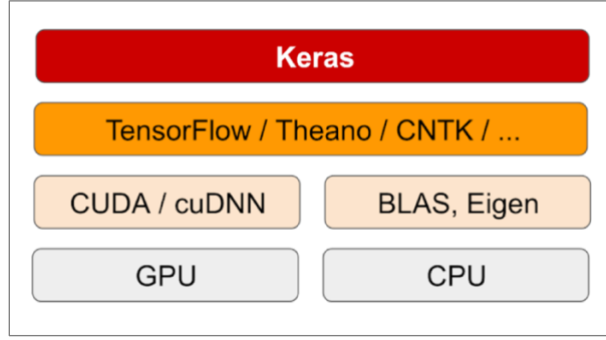
önünde bulundurulduğunda yeterli miktarda veri var olduğunda derin sinir ağlarının yüksek performansa sahip olduğu gözlemlenmiştir.

Bu tez çalışmasında seçilen sinir ağı gözetimli öğrenme türüne aittir. Yani derin öğrenme için yapay sinir ağlarının bir gözetimden geçirilmesi gerekmektedir. Sinir ağının eğitimi için ise i9 10980XE işlemcili ve NVIDIA Quadro RTX 5000 ekran kartına sahip bir bilgisayar kullanılmıştır. Sistem mimarisi oluşturulurken sinir ağının oluşturulması için Python programlama dilinde Keras [49], TensorFlow [50] gibi Python kütüphaneleri kullanılmıştır. Seçilen kütüphaneler derin öğrenme algoritmaları için son dönemde sıkça kullanılan kütüphanelerdir. Bu altyapının seçilmesinin nedeni ise yöntemlerin hızlı bir şekilde tasarlanması ve istenilen mimari değişikliklerin yapısal olarak hızlı bir şekilde sistem üzerinde değiştirilmesine olanak vermesidir.

3.2.1. Keras

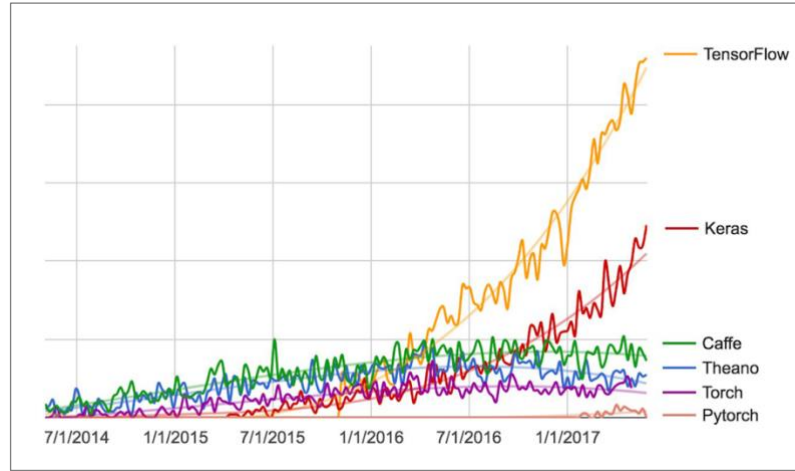
Keras, Python ile yazılmış, makine öğrenimi platformu TensorFlow üzerinde çalışan bir derin öğrenme API'sidir. Hızlı denemeyi mümkün kılmaya odaklanarak geliştirilmiştir. Bir fikirden sonuca olabildiğince hızlı gidebilmek, iyi araştırma yapmanın anahtarıdır. Keras iki yıl önce Mart 2015'te piyasaya sürüldü. Ardından bir kullanıcıdan yüz bine büyümeye devam etti. Keras, Python için bir derin öğrenme kütüphanesidir. Keras, TensorFlow kütüphanesi için bir arayüz görevi görür [49].

Keras 2.0 ile yapılan önemli bir güncelleme, API'nin TensorFlow [50] 1.2 versiyonundan başlayarak artık TensorFlow'un bir parçası olmuştur. Keras, giderek artan sayıda derin öğrenme bağlamında kullanılan bir özellik olan derin öğrenme için ortak bir dil haline gelmektedir [51]. Keras ile hem CPU hem de GPU kullanarak sinir ağlarının eğitimi ve öğrenmesi sağlanmaktadır. Şekil 3.7'de görüldüğü gibi, Keras 2.3 sürümüne kadar TensorFlow, Microsoft Cognitive Toolkit [52], Theano [53] ve PlaidML [54] dahil olmak üzere birden çok arka ucu desteklemiştir [55].



Şekil 3.7. Derin öğrenmede yazılım ve donanımın beraber çalışması [55].

Makine öğrenimi yarışmalarına imkan sağlayan Kaggle [56] platformunda Keras'ın çok yaygın kullanıldığı ve birçok yarışmayı kazandığı görülmektedir. Şekil 3.9'da görüldüğü üzere TensorFlow ve Keras derin öğrenme kütüphaneleri açık ara fark ile tercih edilmektedir. Şekil 3.8'de yer alan grafikte zaman içinde Keras kullanıcılarının sayısının büyümesi gösterilmektedir.



Şekil 3.8. Farklı derin öğrenme kütüphanelerinin arama motorunda aranması [55].

Özetlemek gerekirse, Keras kullanıcı seviyesine yakın olarak (yüksek seviyeli) derin öğrenme modelleri geliştirebilecek yapı taşlarını sunan bir kütüphanedir. Matris işlemleri, türev gibi daha düşük seviyeli işlemleri gerçekleştirmez. Bunun yerine, düşük seviyeli işlemleri yapan, iyi optimize edilmiş arka uç motoru kullanır [55]. Bu tez çalışmasında, sinir ağının oluşumu ve modelin eğitimi sürecinde Keras kütüphanesi kullanılmıştır.

3.2.2. TensorFlow

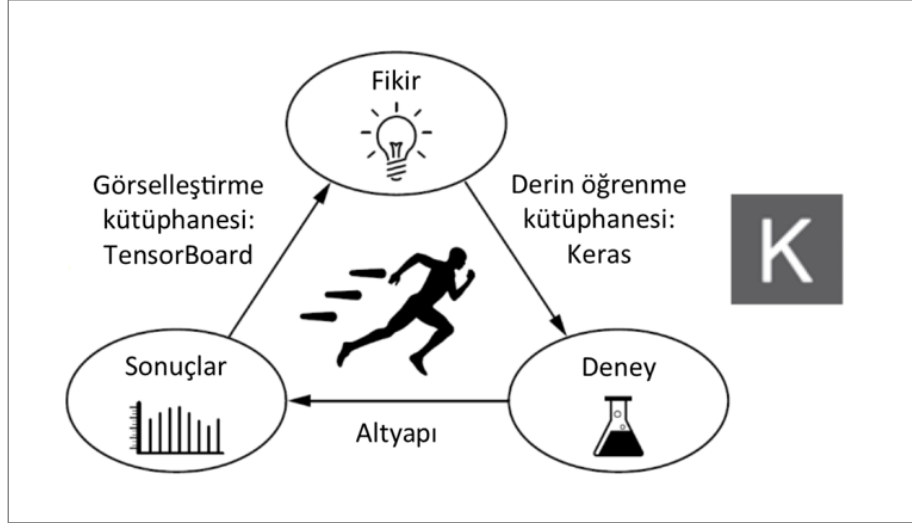
TensorFlow, makine öğrenimi için ücretsiz ve açık kaynaklı bir yazılım kitaplığıdır. Birçok görevde kullanılabilir ancak özellikle derin sinir ağlarının eğitimine odaklanmaktadır. Veri akışına ve türevlenebilir programlamaya dayalı sembolik bir matematik kitaplığıdır. Araştırmacıların MÖ'deki en son teknolojiyi kullanmasına ve geliştiricilerin MÖ destekli uygulamaları kolayca oluşturup dağıtmasına olanak tanıyan kapsamlı, esnek bir araç, kitaplık ve topluluk kaynakları ekosistemine sahiptir [50].

TensorFlow, büyük ölçekte ve heterojen ortamlarda çalışan bir makine öğrenimi sistemidir. Hesaplamayı, paylaşılan durumu ve bu durumu değiştiren işlemleri temsil etmek için veri akışı grafiklerini kullanır. Bir kümedeki birçok makinede ve bir makinede çok çekirdekli CPU'lar, genel amaçlı GPU'lar ve Tensör İşleme Birimleri (TPU) olarak bilinen özel tasarımı ASIC'ler dahil olmak üzere birden çok hesaplama aygıtında bir veri akışı grafiğinin düğümlerini eşler. Bu mimari, uygulama geliştiricisine esneklik sağlar: önceki "parametre sunucusu" tasarımlarında paylaşılan durum yönetimi sistemde yerleşik olarak bulunurken TensorFlow, geliştiricilerin yeni optimizasyonlar ve eğitim algoritmaları ile denemeler yapmasına olanak tanır. TensorFlow, derin sinir ağları üzerinde eğitim ve çıkarımlara odaklanarak çeşitli uygulamaları destekler [57].



Şekil 3.9. TensorFlow kullanım sebepleri ve avantajları [50].

Şekil 3.9’da TensorFlow’un avantajlarına yer verilmiştir. Bahsedilen bu avantajlara göre her ortamda ML projeleri geliştirilirken oluşan modellerin kolay elde edilmesinden bahsedilmiştir. Bu sebeplerin de içerisinde bulunmasından dolayı Keras’a ek olarak TensorFlow kütüphanesi kullanılmıştır.



Şekil 3.10. Derin öğrenme projesi ilerleme döngüsü [55].

Şekil 3.10’da yer alan proje ilerleme döngüsü bu tez projesi kapsamında geçerlidir. Derin öğrenme projeleri öncelikle fikrin ortaya atılması ile başlamaktadır. Ardından gerekli olan programlama kütüphaneleri tespit edilir. Örneğin bu tez çalışması için kullanılan derin öğrenme kütüphaneleri Keras ve TensorFlow olmuştur. Daha sonra modelin eğitimi tamamlandığında test sonuçları için TensorFlow görselleştirme kütüphanesi olan TensorBoard [58] kullanılır. Böylelikle bir derin öğrenme projesi daha sonlanır. Modelin test sonuçlarının tespit edilmesi için TensorBoard kullanım aşamasına bu tez çalışmasında Bölüm 4’te yer verilmektedir.

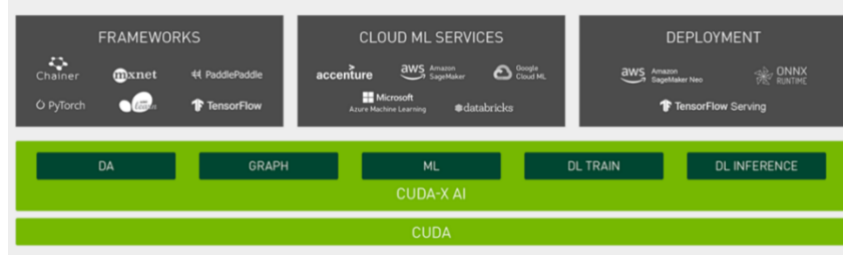
BÖLÜM 4

DİŞLERİN MASK R-CNN İLE SEGMENTASYONU

Çalışmanın derin öğrenme taraflı bölümünün gerçekleştirilmesi için bu bölümde sırasıyla bir derin öğrenme ortamı kurulumu ve Mask R-CNN ağına konfigürasyon parametrelerine yer verilmektedir. Bilgi işlemdeki son gelişmeler, üretilen veri miktarında bir büyümeye yol açmıştır. Sürekli büyüyen verilerin zamanında işlenmesi, çıktı hesaplamasını yeni ortaya çıkan uygulamalar için önemli bir unsur haline getirmiştir. Bir dizi önemli işlem hacmi hesaplama çekirdeğine ilişkin analiz bu çekirdeklerde onları günümüzün çok çekirdekli CPU'ları ve GPU'ları için uygun kılan bol miktarda paralellik olduğunu göstermektedir [59]. GPU kullanımını, CPU'ya karşı eğitim sürecini oldukça hızlı gerçekleştirdiği için bu çalışmada GPU kullanılmıştır.

4.1. DERİN ÖĞRENME ORTAMININ KURULUMU

Derin öğrenme ortamı içerisinde kullanılacak kütüphaneler Keras ve TensorFlow olarak belirlenmiştir. Öncelikli olarak GPU kullanılacağı için CUDA'nın kurulması gereklidir. Proje boyunca kullanılacak bilgisayar Windows 10 Pro işletim sistemine sahiptir. CUDA, paralel programların geliştirilmesini kolaylaştırmak için tasarlanmış bir bilgi işlem mimarisidir. Kapsamlı bir yazılım platformuyla birlikte, CUDA mimarisi programcıların grafik işlem birimlerinin muazzam gücünden yararlanmalarını sağlamaktadır [60]. CUDA programlama modeli, doğal olarak paralel hesaplamaları tanımlamak için basit bir araç sağlar ve NVIDIA'nın [61] Tesla GPU mimarisi [62], büyük ölçüde paralel problemlerde yüksek hesaplama verimi sağlar. Bu makalede, CUDA'nın çeşitli sorunlara uygulanmasıyla elde edilen deneyimler ve GPU'da anahtar hesaplamalar yapılarak elde edilen geleneksel CPU mimarileri üzerinde çalışan sıralı kodlar üzerindeki paralel hızlanmalar incelenmektedir [63].



Şekil 4.1. NVIDIA CUDA servisleri ve dağıtımları [61].

Şekil 4.1’de NVIDIA CUDA’ya ait makine öğrenimi kütüphaneleri, servisleri ve dağıtımları gösterilmektedir. Burada yer alan iş çerçevelerinden TensorFlow kütüphanesi kullanılacaktır. Bununla birlikte cuDNN kurulumu da yapılması gereklidir. CuDNN, CUDA için gerekli olan derin öğrenme kütüphanelerini barındıran bir araç kitidir. CuDNN’nin temel hedeflerinden biri sinir ağı çerçeveleri topluluğunun API’lerinden eşit olarak yararlanmasını sağlamaktır. Buna göre, cuDNN kullanıcılarının belirli bir yazılım çerçevesini veya hatta veri düzenini benimsemeleri gerekmez. CuDNN, ayrıca derin öğrenme için yaygın olarak kullanılan diğer işlevleri de sağlar. Örneğin, yaygın olarak kullanılan üç nöron aktivasyon fonksiyonu sağlar; Sigmoid, Rektifiye Lineer ve Hiperbolik Tanjant [64].

NVIDIA Quadro and NVIDIA RTX Mobile GPUs

GPU	Compute Capability
RTX A5000	8.6
RTX A4000	8.6
RTX A3000	8.6
RTX A2000	8.6
RTX 5000	7.5

Şekil 4.2. NVIDIA uyumlu ekran kartının hesaplama kapasitesi [63].

Şekil 4.2’de bilgisayarda bulunan ekran kartının hesaplama kapasitesi verilmiştir. Bu bilgiler doğrultusunda diğer ekran kartlarına göre daha hızlı hesaplama yeteneğine sahip olduğu anlaşılmıştır. Bu çalışmada gerekli CUDA araç kiti versiyonu 10 iken cuDNN versiyonu ise 7.4 olarak seçilmiştir. Şekil 4.3’te yer alan bu versiyonların seçilmesinin sebebi bilgisayarda bulunan ekran kartının NVIDIA Quadro RTX 5000 olmasından kaynaklanmaktadır. Sistem içerisinde bir hata ile karşılaşılmaması için sahip olunan ekran kartı için gerekli olan versiyonlar kurulmalıdır. Kurulum için ise

Anaconda [65] içerisinde terminal ekranı kullanılmıştır. Conda [65] kurulum paketi ile işlem yapılmıştır.

GPU

Sürüm	Python sürümü	Derleyici	Araçlar oluşturun	cuDNN	CUDA
tensorflow_gpu-2.4.0	3.6-3.8	MSVC 2019	Bazel 3.1.0	8.0	11.0
tensorflow_gpu-2.3.0	3.5-3.8	MSVC 2019	Bazel 3.1.0	7.6	10.1
tensorflow_gpu-2.2.0	3.5-3.8	MSVC 2019	Bazel 2.0.0	7.6	10.1
tensorflow_gpu-2.1.0	3.5-3.7	MSVC 2019	Bazel 0.27.1-0.29.1	7.6	10.1
tensorflow_gpu-2.0.0	3.5-3.7	MSVC 2017	Bazel 0.26.1	7.4	10
tensorflow_gpu-1.15.0	3.5-3.7	MSVC 2017	Bazel 0.26.1	7.4	10
tensorflow_gpu-1.14.0	3.5-3.7	MSVC 2017	Bazel 0.24.1-0.25.2	7.4	10
tensorflow_gpu-1.13.0	3.5-3.7	MSVC 2015 güncelleme 3	Bazel 0.19.0-0.21.0	7.4	10
tensorflow_gpu-1.12.0	3.5-3.6	MSVC 2015 güncelleme 3	Bazel 0.15.0	7.2	9.0
tensorflow_gpu-1.11.0	3.5-3.6	MSVC 2015 güncelleme 3	Bazel 0.15.0	7	9
tensorflow_gpu-1.10.0	3.5-3.6	MSVC 2015 güncelleme 3	Cmake v3.6.3	7	9

Şekil 4.3. CUDA ve cuDNN araç kitlerinin kurulumu [62].

CUDA ve cuDNN kiti için gerekli olan TensorFlow GPU kütüphanesinin versiyonu 1.14 olarak belirlenerek aşağıdaki kod satırında belirtilmiştir.

```
#conda paket yükleyici ile tensorflow gpu kütüphanesinin yüklenmesi  
conda install tensorflow-gpu==1.14.0
```

Bununla birlikte, sinir ağındaki katmanların oluşturulması ve modelin eğitilmesi için gerekli diğer bir kütüphane olan Keras kurulumu için versiyon 2.2.5 olarak seçilmiştir. Aşağıdaki kod satırında ise Conda ile Keras kütüphanesinin kurulumuna yer verilmiştir.

```
#conda paket yükleyici ile keras kütüphanesinin yüklenmesi  
conda install keras==2.2.5
```

Böylelikle bir derin öğrenme ortamı için gerekli olan tüm araçlar sisteme yüklenmiştir. Bir sonraki aşamada ise kullanılacak sinir ağı olan Mask R-CNN'nin konfigürasyonları ayarlanmaktadır.

4.2. MASK R-CNN KONFIGÜRASYON SİSTEMİ

Mask R-CNN için öncelikli olarak GPU aygıtlarının kontrolü yapılmalıdır. Şekil 4.4'te ise TensorFlow içerisinde python modülü olan *device_lib* fonksiyonu ile GPU ve CPU aygıtlarının görünürlüğü kontrol edilmiştir.

```
from tensorflow.python.client import device_lib
device_lib.list_local_devices()

[name: "/device:CPU:0"
 device_type: "CPU"
 memory_limit: 268435456
 locality {
 }
 incarnation: 5412419796901316595,
 name: "/device:GPU:0"
 device_type: "GPU"
 memory_limit: 15142682624
 locality {
   bus_id: 1
   links {
 }
 }
 incarnation: 15597130823223522319
 physical_device_desc: "device: 0, name: Quadro RTX 5000, pci bus id: 0000:17:00.0, compute capability: 7.5",
 name: "/device:GPU:1"
 device_type: "GPU"
 memory_limit: 15142682624
 locality {
   bus_id: 1
   links {
 }
 }
```

Şekil 4.4. TensorFlow GPU ve CPU kontrolü

Ardından kullanılacak Python kütüphaneleri Şekil 4.5'teki gibi içeriye aktarılmıştır. Bu kütüphaneler içerisinde görüntülerde yer alan pikseller üzerinde vektörel ve matematiksel işlemler yapmayı sağlayan NumPy [66] kütüphanesi de yer almaktadır.

```
import sys
import random
import math
import re
import time
import numpy as np
import cv2
import matplotlib
import matplotlib.pyplot as plt
import json
import datetime
import skimage.draw
```

Şekil 4.5. Gerekli Python kütüphanelerinin içeriye aktarılması.

Mask R-CNN için bitewing görüntülerde tespit edilecek nesne sınıfının adı *teeth* olarak belirlenmiştir. Kullanılacak GPU sayısı 1 olarak belirlenmiştir. Bunun yanı sıra artık ağlardan ise ResNet-101 kullanılmıştır. Çalışmada kullanılacak toplam sınıf sayısı ise 2 olarak belirlenmiştir. Bunun sebebi ise sadece dış ve arka plan nesnelerinin

var olmasından kaynaklanmaktadır. Gerekli tüm başlangıç konfigürasyonları Şekil 4.6'da gösterilmektedir.

```
# Yapılandırmaya tanınabilir bir ad verin
NAME = "teeth"

GPU_COUNT = 1

# İki görüntüyü sığdırabilen 12 GB belleğe sahip bir GPU kullanıyoruz.
# Daha küçük bir GPU kullanıyorsanız aşağı ayarlayın.
IMAGES_PER_GPU = 1
BACKBONE="resnet101"
# Sınıf sayısı (arka plan dahil)
NUM_CLASSES = 1 + 1 # Arka plan + dış

# Dönem başına eğitim adımı sayısı
STEPS_PER_EPOCH = 100

# < %90 güvenle algılamaları atla
DETECTION_MIN_CONFIDENCE = 0.9
```

Şekil 4.6. İlgili konfigürasyon parametrelerinin ve omurga ağının seçilmesi.

Ek olarak tespit edilecek minimum güven değeri ise 0.9 olarak belirlenmiştir. Bunun anlamı, bir görüntüde %90 ve üzeri dış olarak tespit edilen bölgeler dış etiketine sahip olacaktır. Şekil 4.7'de verilen değerler incelendiğinde ResNet-101 mimarisine uygun boyutların seçildiği görülmüştür. Bununla birlikte en son çıkarılacak olan maskelerin boyutu ise 28x28 olarak belirlenmiştir.

```
Configurations:
BACKBONE                resnet101
BACKBONE_STRIDES        [4, 8, 16, 32, 64]
BATCH_SIZE              1
BBOX_STD_DEV            [0.1 0.1 0.2 0.2]
COMPUTE_BACKBONE_SHAPE None
DETECTION_MAX_INSTANCES 100
DETECTION_MIN_CONFIDENCE 0.9
DETECTION_NMS_THRESHOLD 0.3
FPN_CLASSIF_FC_LAYERS_SIZE 1024
GPU_COUNT               1
GRADIENT_CLIP_NORM      5.0
IMAGES_PER_GPU          1
IMAGE_CHANNEL_COUNT      3
IMAGE_MAX_DIM           1024
IMAGE_META_SIZE         14
IMAGE_MIN_DIM           800
IMAGE_MIN_SCALE         0
IMAGE_RESIZE_MODE        square
IMAGE_SHAPE              [1024 1024 3]
LEARNING_MOMENTUM        0.9
LEARNING_RATE           0.001
LOSS_WEIGHTS             {'rpn_class_loss': 1.0, 'rpn_bbox_loss': 1.0, 'mrcnn_class_loss': 1.0, 'mrcnn_bbox_loss': 1.0,
'mrcnn_mask_loss': 1.0}
MASK_POOL_SIZE          14
MASK_SHAPE               [28, 28]
```

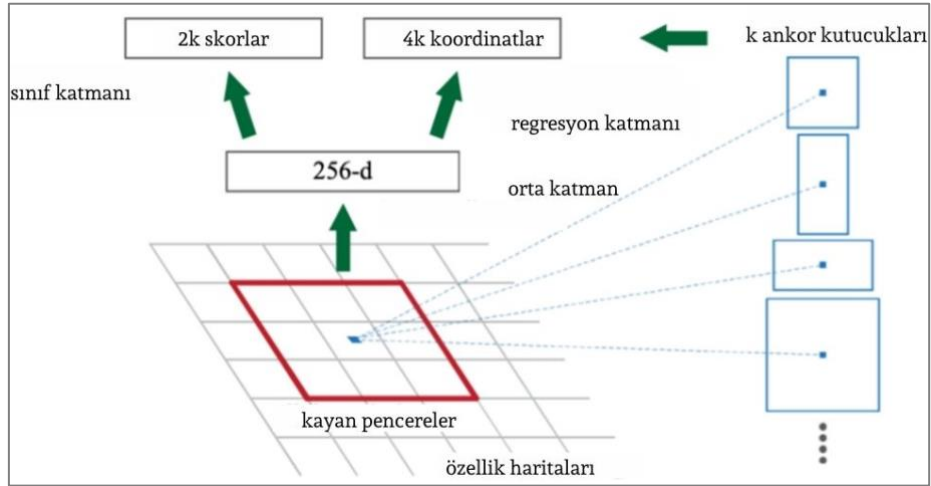
Şekil 4.7. Seçilen konfigürasyon çıktıları.

4.3. BÖLGE TEKLİF AĞI VE ÖZELLİK PİRAMİT AĞI SİSTEMİ

Bölge Teklif Ağı veya RPN, her konumdaki nesne sınırlarını ve nesnellik puanlarını eşzamanlı olarak tahmin eden tamamen evrişimli bir ağıdır. RPN, yüksek kaliteli bölge önerileri oluşturmak için uçtan uca eğitilmiştir [30]. Özellik piramitleri, farklı ölçeklerdeki nesnelere algılamak için tanıma sistemlerinde temel bir bileşendir. Ancak son zamanlardaki derin öğrenme nesnesi algılayıcıları, kısmen bilgi işlem ve bellek açısından yoğun oldukları için piramit temsillerinden kaçınmıştır [67].

4.3.1. Bölge Teklif Ağı Sistemi

RPN ve Fast R-CNN gibi algoritmalar, evrişimli özelliklerini paylaşarak tek bir ağda birleştirilebilir ve son zamanlarda popüler olan sinir ağları terminolojisini dikkat mekanizmalarıyla kullanan RPN bileşeni, birleşik ağa nereye bakacağını söylemektedir [28]. Şekil 4.8’de nesnelere kayan pencere mantığı ile k adet ankor kutucukları oluşur ve bölge önerileri ağa sunulur.



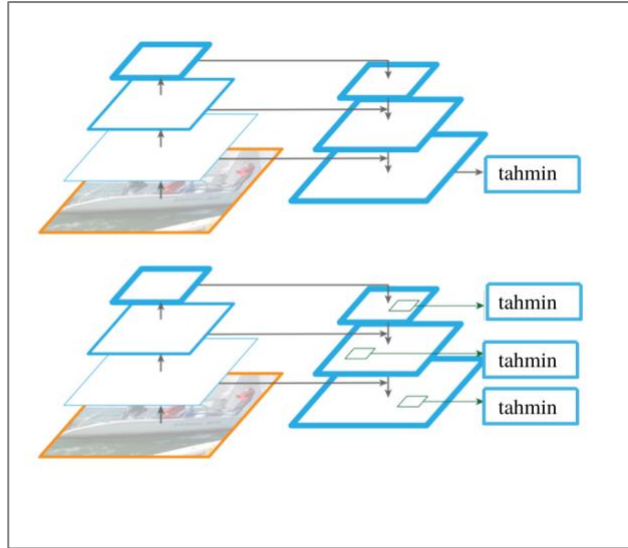
Şekil 4.8. Bölge teklif ağı [30]

Bölge önerileri oluşturmak için, son paylaşılan evrişim katmanı tarafından evrişimli özellik haritası çıktısı üzerinde küçük bir ağ kaydırılmaktadır. Bu küçük ağ, girdi evrişimli özellik haritasının bir $n \times n$ uzamsal penceresini girdi olarak alır. Her kayan pencere daha düşük boyutlu bir özelliğe eşlenir. Bu özellik ile iki kardeş tam bağlı katmana beslenir: bir kutu regresyon katmanı ve bir kutu sınıflandırma katmanı. Bu

mini ağ, Şekil 4.8’de soldaki tek bir konumda gösterilmektedir. Mini ağın kayar pencere şeklinde çalıştığı için, tam bağlantılı katmanların tüm uzamsal konumlar arasında paylaştığını unutmayın. Bu mimari doğal olarak bir $n \times n$ evrişim katmanı ve ardından iki kardeş 1×1 evrişim katmanı (sırasıyla reg ve cls için) ile uygulanır [28].

4.3.2. Özellik Piramit Ağı Sistemi

Tüm ölçeklerde üst düzey anlamsal özellik haritaları oluşturmak için yanal bağlantılara sahip yukarıdan aşağıya bir mimari geliştirilmiştir. Özellik Piramit Ağı (FPN) olarak adlandırılan bu mimari, çeşitli uygulamalarda genel bir özellik çıkarıcı olarak önemli bir gelişme gösterir [67].

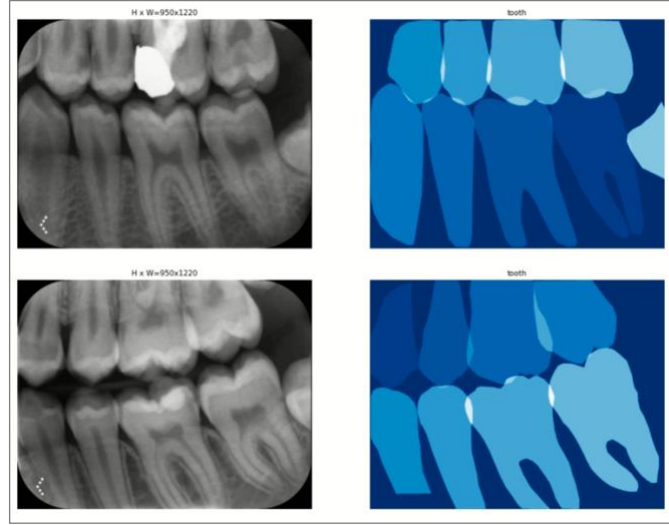


Şekil 4.9. Yapılan tahminlerle bir özellik piramidi olarak kullanılan model [67].

Şekil 4.9’da yer alan görseldeki üst bölümde tahminlerin en iyi seviyede yapıldığı atlama bağlantılı yukarıdan aşağıya bir mimari gösterilmektedir. Görselin alt bölümünde ise benzer bir yapıya sahip olan ancak bunu her düzeyde bağımsız olarak yapılan tahminlerle bir özellik piramidi olarak kullanılan model örneği verilmiştir. Orijinal RPN tasarımında, tek ölçekli bir evrişimli özellik haritasının üstünde, yoğun 3×3 kayan pencerelerde küçük bir alt ağ değerlendirilir, nesne/nesne olmayan ikili sınıflandırma ve sınırlayıcı kutu regresyonu yapılır. Bu, sınıflandırma ve regresyon için ağ başı olarak adlandırdığımız iki kardeş 1×1 evrişim tarafından takip edilen bir 3×3 evrişim katmanı ile gerçekleştirilir. Nesne/nesne olmayan kriteri ve sınırlayıcı

kutu regresyon hedefi, ankor olarak adlandırılan bir dizi referans kutusuna göre tanımlanır [66]. Ankorlar, farklı şekillerdeki nesnelere kapsamak için önceden tanımlanmış birden çok ölçek en boy oranına sahiptir [67].

Sistemde bulunan bitewing görüntüleri üzerinde önceden etiketlenmiş diş nesnelerinin görselleştirilmesi ise Şekil 4.10'da yer almaktadır. Bu adıma kadar olan bölümde Mask R-CNN konfigürasyonlarına ve detaylarına yer verilmiştir.



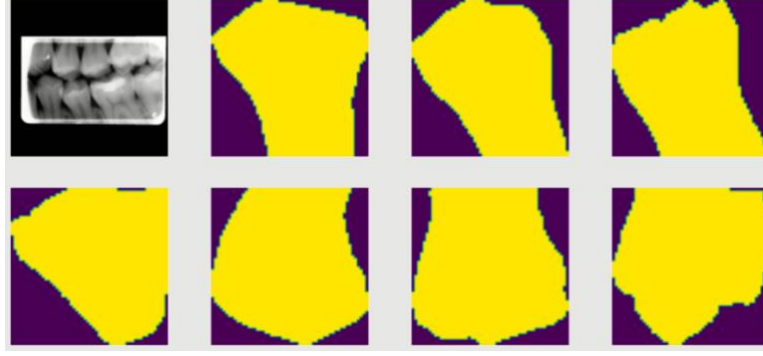
Şekil 4.10. Örnek orijinal bitewing görüntüleri ve diş maskeleri.

Şekil 4.10'da sol tarafta orijinal bitewing görüntüsü yer alırken sağ tarafta ise maskelenmiş nesnelere gösterilmektedir. Poligon olarak etiketlenmiş diş nesnelere koordinatsal olarak maviye boyanmıştır. Pksel yoğunluğuna göre ise koyudan açığa doğru renklendirilmesi yapılmaktadır. Şekil 4.11'de ise oluşturulan diş nesnelere daha ayrıntılı görüntülenmesi adına rastgele 3 adet maske yazdırılmıştır.



Şekil 4.11. Diş nesnelere ayrıntılı maskelendirilmesi.

Şekil 4.12’de ise 28x28 boyutuna sahip maskeler yeniden boyutlandırılarak 56x56 boyutuna çıkartılarak sınırlayıcı kutucuklar ile sinir ağı yapısında veri artırmaya gidilerek eğitim kümesinden seçilen örnek bir görüntüdeki dişlerin maskelenmesi görülmektedir.



Şekil 4.12. Veri artırma ve yeniden boyutlandırmaya gidilerek dişlerin maske çıktıları.

Şekil 4.13’te verilen görselde toplam oluşturulan ankorların kaç adet olduğu model dosyası üzerinden hesaplatılmıştır.

```
backbone_shapes = modellib.compute_backbone_shapes(config, config.IMAGE_SHAPE)
anchors = utils.generate_pyramid_anchors(config.RPN_ANCHOR_SCALES,
                                         config.RPN_ANCHOR_RATIOS,
                                         backbone_shapes,
                                         config.BACKBONE_STRIDES,
                                         config.RPN_ANCHOR_STRIDE)
```

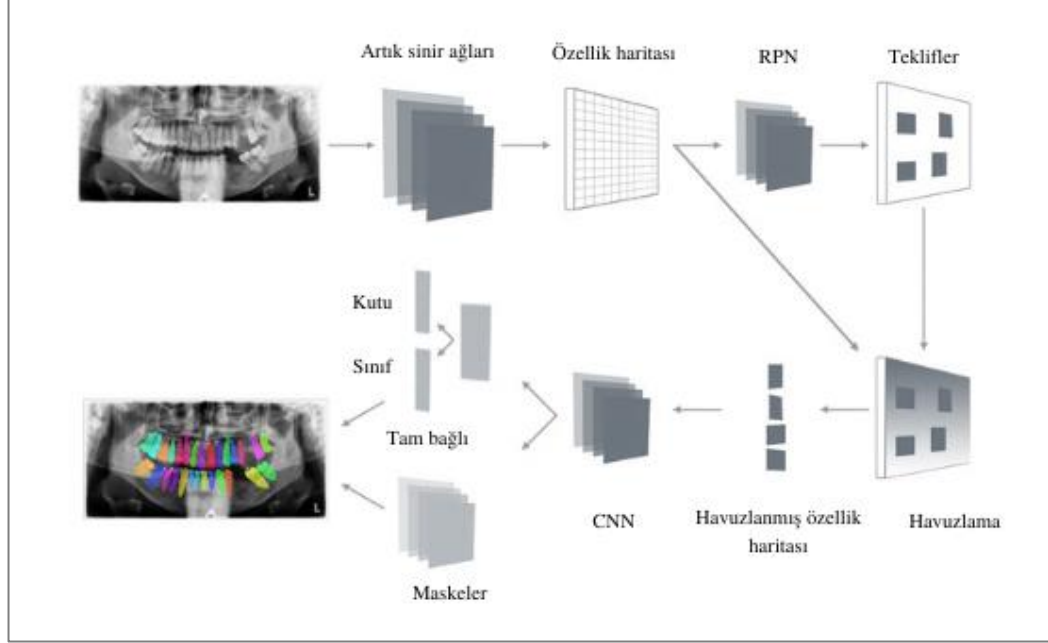
Şekil 4.13. Ankorların oluşturulmasına ait Python kodu.

Bununla birlikte, Şekil 4.14’te verilen görselde ise ankorların kaç seviyede olduğu ve değerleri verilmiştir.

```
Sayaç: 261888
Ölçekler: (32, 64,128, 256, 512)
Oranlar: [0.5, 1, 2]
Hücre başına ankor: 3
Seviye 0 ankorları: 196608
Seviye 1 ankorları: 49152
Seviye 2 ankorları: 12288
Seviye 3 ankorları: 3072
Seviye 4 ankorları: 768
```

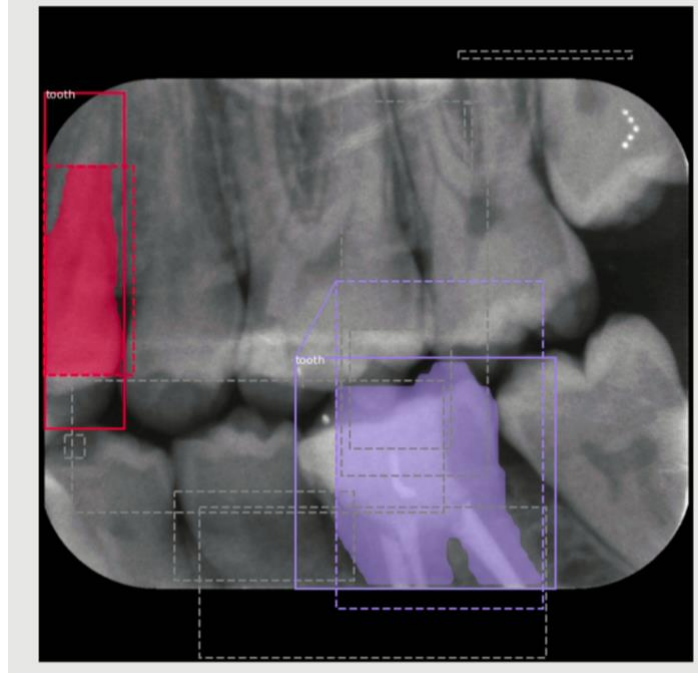
Şekil 4.14. Ankorların üretilmesi ve özetin yazdırılması.

Veri setindeki görüntüler Şekil 4.15'te gösterildiği gibi sırasıyla iki aşamalı bir RPN yapısından geçirildi. Bu adımda, bölge olması muhtemel alanlar RPN ağında tutulmuştur. Örtüşme kutuları, maksimum olmayan bastırma (NMS) algoritması için taranmıştır.



Şekil 4.15. Segmentasyon iş akış mimarisi.

Sonuç olarak Mask R-CNN ile sınırlayıcı kutular ve sınıf puanlarına ek olarak her bir ROI için ikili maske oluşturulmuştur. Şekil 4.16'da oluşturulan 200 adet ROI bölgesinden rastgele çekilmiş ankorlar ve ROI kutucukları görülmektedir. Bu görsele göre RPN'den geçirilmiş bölge teklif önerileri ile 200 adet toplam öneri arasından sadece 10 adet rastgele kutucuğun gösterilmesi sağlanmıştır. Böylelikle 2 adet dış olduğu varsayılan nesne bölgesi Mask R-CNN ile maskesel boyanmaya gönderilmiştir. Görselde gösterilen mor ve kırmızı boyama alanları dış nesnesi olduğunu ispatlamaktadır.



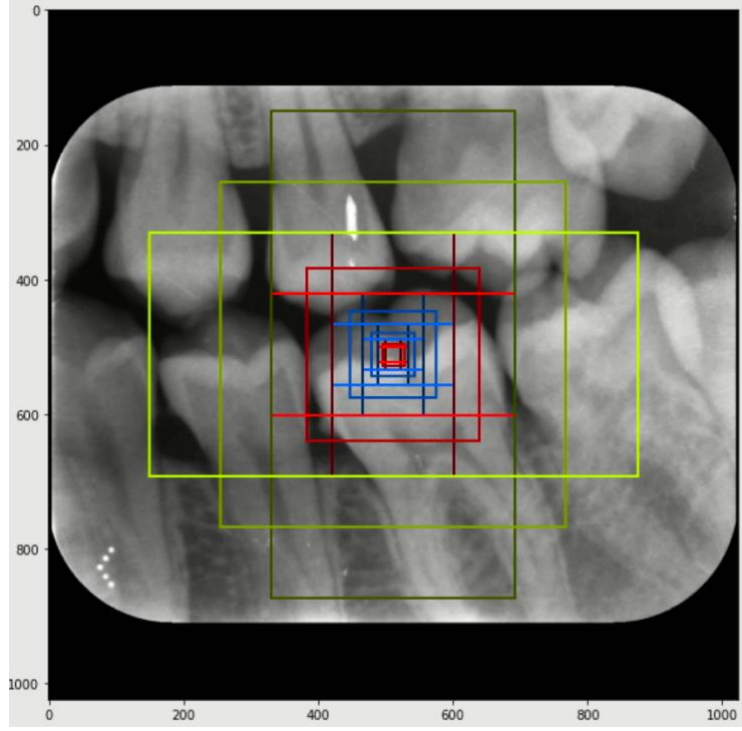
Şekil 4.16. 200 adet ROI'den gösterilen 10 adet örnek.

Şekil 4.17'de bir bitewing görüntüde yer alan farklı ankorlar gösterilmektedir. Mask R-CNN yapısı gereği örtüşen kutuların tespit edilmesi için sınırlayıcı ankor kutuları kullanılmıştır. Şekilde söz edilen ankorlar, tespit edilecek ankor nesnelерinin ölçeklerini yakalamak için oluşturulmuştur.

Seviye 0 ankorları: 196608 Özellik haritası boyutu: [256, 256]
Seviye 1 ankorları: 49152 Özellik haritası boyutu: [128, 128]
Seviye 2 ankorları: 12288 Özellik haritası boyutu: [64, 64]
Seviye 3 ankorları: 3072 Özellik haritası boyutu: [32, 32]
Seviye 4 ankorları: 768 Özellik haritası boyutu: [16, 16]

Şekil 4.17. Seviyesel olarak ankorların oluşumu ve özellik haritaların indirgenmesi.

Şekil 4.18'de ise oluşturulan ankorların 5 seviyeli olarak değişiminin görselleştirilmesine aittir. Şekil 4.17'de verilen görseldeki özellik haritalarının zamanla boyutsal olarak indirgenmesi görülmektedir. Şekil 4.18'de ise bu durumun ankor kutucuklarına yansımaları gösterilmektedir.



Şekil 4.18. Örnek bitewing görüntüsünde sinir ağı eğitimi sonucu oluşan pozitif ankor kutuları.

4.4. DİŞLERİN SEGMENTASYONU

Mask R-CNN ağının eğitimi için model dosyası içerisinde mod olarak eğitim modu açılarak daha önceden ayarlanmış konfigürasyonlar ile yükleme tamamlanmıştır. Şekil 4.19'da görüldüğü üzere, sinir ağındaki eğitim süreci için girdi değerlerinde kullanılacak ağırlıkların eğitiminin tekrarı yerine önceden eğitilmiş MS COCO [44] ağırlıkları kullanılmıştır.

```
# Eğitim modunda model oluşturma
model = modellib.MaskRCNN(mode="training", config=config,
                           model_dir=MODEL_DIR)
```

Şekil 4.19. Sinir ağının eğitimi için model oluşturma.

Model kodunun oluşum sürecinde ImageNet [40] içerisinde ağırlıklar çekildikten sonra mrcnn klasörü içerisinde model için gerekli yol ve maskeler ayarlanmıştır. Daha önceden çalıştırılmış bir eğitim dosyası mevcut ise son model dosyasını

kullanacak şekilde kod uyarlanmıştır. Eğitim modu için iki ayrı aşama mevcuttur. İlk aşama için sadece başlangıç katmanlardaki ağın eğitimi sağlanır. Bunun için ise gerekli işlemlerin sağlandığı program kodunun yapısı aşağıdaki gibidir.

```
# ilk dallar için eğitim yapma
```

```
# Katman= "heads" geçişi, head katmanları dışındaki tüm katmanları dondurur.  
Ayrıca, ad modeline göre hangi katmanların eğitileceğini seçmek için normal bir ifade  
iletebilirsiniz.
```

```
model.train(dataset_train, dataset_test,  
            learning_rate=config.LEARNING_RATE,  
            epochs=epoch_count,  
            layers='heads')
```

Ağın eğitimi için Şekil 4.20’de görüldüğü gibi 100 iterasyon sayısı için eğitim başlatıldıktan sonra sinir ağı katmanları sıralanmıştır. TensorFlow ve Keras kütüphanelerinin kullanıldığı yapı Sequential (Sıralı) oluşturulduğu için Şekil 4.20’de görüldüğü gibi katmanlar sırasıyla gelmiştir.

```
Starting at epoch 100. LR=0.0001  
  
Checkpoint Path: D:\bitewing\Mask_RCNN\samples\dentiAssist\newdatalogs\teeth20210324T1339\mask_rcnn_teeth_{epoch:04d}.h5  
Selecting layers to train  
conv1 (Conv2D)  
bn_conv1 (BatchNorm)  
res2a_branch2a (Conv2D)  
bn2a_branch2a (BatchNorm)  
res2a_branch2b (Conv2D)  
bn2a_branch2b (BatchNorm)  
res2a_branch2c (Conv2D)  
res2a_branch1 (Conv2D)  
bn2a_branch2c (BatchNorm)  
bn2a_branch1 (BatchNorm)  
res2b_branch2a (Conv2D)  
bn2b_branch2a (BatchNorm)  
res2b_branch2b (Conv2D)  
bn2b_branch2b (BatchNorm)  
res2b_branch2c (Conv2D)  
bn2b_branch2c (BatchNorm)  
res2c_branch2a (Conv2D)  
bn2c_branch2a (BatchNorm)  
res2c_branch2b (Conv2D)  
bn2c_branch2b (BatchNorm)  
res2c_branch2c (Conv2D)  
bn2c_branch2c (BatchNorm)  
res3a_branch2a (Conv2D)  
bn3a_branch2a (BatchNorm)  
res3a_branch2b (Conv2D)  
bn3a_branch2b (BatchNorm)  
res3a_branch2c (Conv2D)  
res3a_branch1 (Conv2D)
```

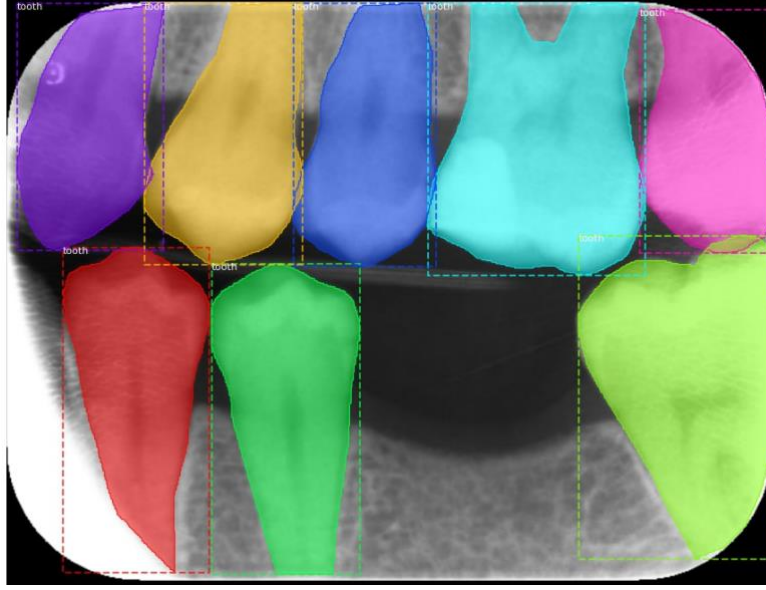
Şekil 4.20. 100 iterasyon sayısı için sinir ağına var olan katmanların listelenmesi.

Bunun haricinde eğitimin ikinci aşamasında ise sadece ilk katmanlar için değil tüm katmanlar kapsanacak şekilde ayarlar sağlanmıştır.

```
# Tüm katmanlara ince ayar yapın
# Katmanları geçmek = "all" tüm katmanları eğitir.
# Ayrıca, ad modeline göre hangi katmanların eğitileceğini seçmek için normal bir ifade iletebilirsiniz.
model.train(dataset_train, dataset_test,
            learning_rate=config.LEARNING_RATE,
            epochs=epoch_count,
            layers=all)
```

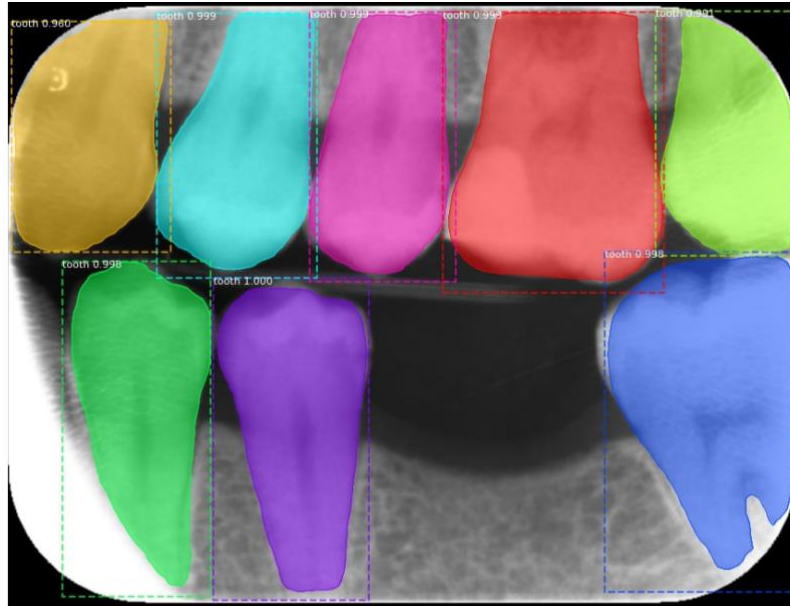
Yukarıdaki kod satırlarında görüldüğü gibi katmanlar tümünü kapsayacak şekilde ayarlanmıştır. Eğitim, 0,001 öğrenme oranı ve 400 epoch için Adam optimizasyon algoritması seçilerek tamamlanmıştır. Bu aşamada veri kümesinden rastgele seçilen 1000 adet eğitim görüntüsü sinir ağına verilmiştir. Daha sonra sinir ağına 200 adet test görüntüsü kullanılarak modelin doğruluğunun kontrolü sağlandı. Omurga ağı olarak artık ağlardan olan artık ağlar mimarisi için 101 katmanlı ağ tercih edilmiştir.

Model sırasıyla 1, 50, 100, 200 ve 400 iterasyon sayısı için eğitilerek en son 400 iterasyon sayısı için Şekil 4.21'de etiketlenmiş görüntülerin maskelenmesi gösterilmektedir. Şekil 4.21, uzman görüşü ile etiketlenmiş gerçek referans görseline aittir. Ağın bir sonucu olarak, görüntülerde ilgili bölgenin alan başına sınırlayıcı kutusu ve maskesi oluşturulmuştur. Bu nedenle, tanımlanacak belirli nesnelere, Şekil 4.21'de gösterildiği gibi kendi sınıflarının segmentasyon maskeleri ve sınırlayıcı kutulardadır. Model oluşturulduktan sonra gerçek referans değerli nesnelere görselleştirmesinin yapıldığı Şekil 4.21'de diş nesnelere **tooth** etiketi ile renklendirilmiştir. Diş olmayan bölgeler arka plan olarak nitelendirildiği için boş bırakılmıştır.



Şekil 4.21. Örnek bir bitewing görseline ait gerçek referans değerli maskeler.

400 iterasyon eğitimi sonucu bitewing görüntülerde diş nesnelerrinin maskelenmesi Şekil 4.22’de gösterildiđi gibidir. Diş segmentasyon maskelerine ek olarak diş olma olasılıkları da sınırlayıcı kutucukların yanında gösterilmektedir. Verilen görselde sınırlayıcı kutular ve eşleştirme puanlarına ek olarak diş maskeleri yer almaktadır. Bu nesnelerr, eğitim sonucunda örnek segmentasyon ile her bir ROI bölgesi için farklı bir maske rengine boyanmıştır.



Şekil 4.22. Verilen görüntünün eğitim sonucu oluşturulan segmentasyon maskeleri.

Şekil 4.22’de ise Mask R-CNN için 100 epoch eğitim sonucunda oluşturulan yüksek kaliteli segmentasyon maskelerine ve sınıf puanlarına yer verilmektedir. Örneğin sarı maskeli diş incelenecek olursa %96 oranında diş olduğu gösterilmektedir. Görsel incelendiğinde genel olarak başarılı bir eğitim sonunda test yapıldığı gösterilmektedir. Böylelikle dişlerin segmentasyonu arka plandan ayırtlandırılarak sağlanmıştır.

BÖLÜM 5

DENEYSEL ÇALIŞMALAR

Mask R-CNN ağının bitewing radyografik görüntülerde sağladığı başarımın test edilmesi için Jupyter Notebook ortamında TensorBoard [60] görselleştirme araç kiti kullanılmıştır. Sinir ağı modelinin eğitimi boyunca 1, 50, 100, 200 ve 400 iterasyonda modelin veriler üzerinde gezinmesi sağlanırken başarım metriklerinin hesaplanması sağlanmıştır.

Deneysel çalışmalara başlanırken, ilk olarak Mask R-CNN ağının alt yapısında gerçekleşen kayıp değerinin hesaplanması üzerinde incelemeler gerçekleştirilmiştir. Ardından gerçek referans görselleri ve tahmin sonuçları üzerinde karşılaştırmalar yapılmıştır, böylelikle karmaşıklık matrisi üzerinden başarı sonuçları incelenmiş oldu. Bununla birlikte, performans metriklerinden olan duyarlılık ve hassasiyet arasındaki ilişkiyi gösteren grafik yazdırılmıştır. 400 iterasyon için literatürde sıkça kullanılan hassasiyet, ortalama genel hassasiyet, duyarlılık ve f1-skoru için başarım sonuçları sırasıyla ölçülmüştür. Mask R-CNN için kullanılan kayıp fonksiyonu, sınıflandırma kaybını, gerçek referans kutusu koordinatlarını ve segmentasyondan alınan maskeleri birleştirmektedir. Eşitlik 5.1’de ise bu kayıp fonksiyonunun hesaplatılmış eşitliği bulunmaktadır.

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{box} + \mathcal{L}_{mask} \quad (5.1)$$

Eşitlik 5.1’de \mathcal{L}_{cls} ve \mathcal{L}_{box} Faster R-CNN [44] ağında kullanılan kayıp değerleri ile aynıdır ve \mathcal{L}_{mask} değeri Mask R-CNN ile segmentasyondan alınan maskeleri ifade etmektedir. \mathcal{L}_{mask} ise şu şekilde ifade edilir:

$$\mathcal{L}_{mask} = -\frac{1}{m^2} \sum_{1 \leq i, j \leq m} [y_{ij} \log \hat{y}_{ij}^k + (1 - y_{ij}) \log (\log \hat{y}_{ij}^k)] \quad (5.2)$$

Toplam sınıf sayısı k ile gösterilir. Her sınıf ve her bir ROI için $m \times m$ boyutunda bir maske oluşturulmuştur. Çıkış boyutu, $k \times m^2$ olarak alınmıştır. Eşitlik 5.2’de y_{ij} , oluşturulan maskedeki bir koordinatın (i, j) etiketidir ve \hat{y}_{ij}^k , Eşitlik 5.2’deki k sınıfı için tahmin edilen değerdir.

Eğitim sonucunda, Eşitlik 5.3, Eşitlik 5.4, Eşitlik 5.5 ve Eşitlik 5.6’da var olan başarımların metrikleri, mevcut test görüntülerinin doğruluğunu değerlendirmek için kullanılmıştır. Eşitlik 5.3’te gösterilen formülde var olan ortalama genel hassasiyet sonuçları Mask R-CNN eğitiminin doğrulanması için kullanılan başarımların metriğidir. Bununla birlikte Eşitlik 5.4’te formülü verilen ve hassasiyet olarak literatürde yer edinmiş değişken ise karmaşıklık matrisindeki tüm pozitif değerler ile hesaplanmıştır. Eşitlik 5.3’te bahsi geçen ortalama genel hassasiyet puanı, mevcut farklı tespit zorluklarına bağlı olarak tüm sınıflar ve/veya genel birleşim üzerinden kesişim (IOU) eşikleri üzerinden ortalama hassasiyet alınarak hesaplanmıştır.

$$\text{Ortalama Genel Hassasiyet} = \frac{DP + DN}{DP + DN + YP + YN} \quad (5.3)$$

$$\text{Hassasiyet} = \frac{DP}{DP + YP} \quad (5.4)$$

$$\text{Duyarlılık} = \frac{DP}{DP + YN} \quad (5.5)$$

$$F1 - \text{Skoru} = 2 \times \frac{\text{Hassasiyet} \times \text{Duyarlılık}}{\text{Hassasiyet} + \text{Duyarlılık}} \quad (5.6)$$

Eşitlik 5.5’te var olan duyarlılık metriği, oluşturulan modelin gerçek pozitiflerinin doğru bir şekilde tanımlanmasının ölçülmesiyle hesaplanmıştır. Eşitlik 5.3, Eşitlik 5.4, Eşitlik 5.5 ve Eşitlik 5.6’da var olan başarımların metriklerinde bahsi geçen DN değişkeni doğru ve negatifleri temsil ederken DP doğru pozitifleri temsil etmektedir. Bununla birlikte YP yanlış pozitifleri ifade ederken YN ise yanlış negatifleri ifade etmektedir. Bu değerler gerçek referans ve tahmin arasındaki farka bakılarak hesaplanmaktadır.

Çizelge 5.1’de ise Eşitlik 5.3, Eşitlik 5.4, Eşitlik 5.5 ve Eşitlik 5.6’da var olan başarımların hesaplanarak ölçüldüğü 400 iterasyona kadar eğitimin sağlandığı modelin sonuçları gösterilmiştir. Eşitlik 5.6’da yer alan F1-Skoru başarımların hesaplanmasında, çalışmada kullanılan ortalama genel hassasiyet ve ortalama genel duyarlılık metrikleri kullanılmıştır.

Çizelge 5.1. Önerilen sistemin test kümesinde ölçülen performans sonuçları.

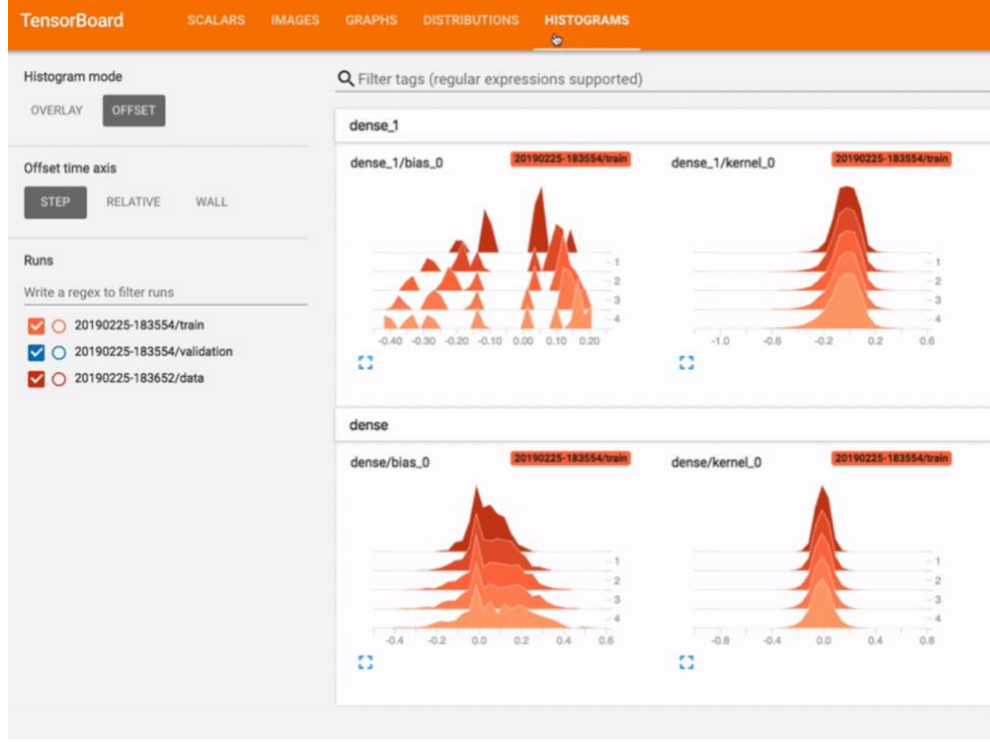
İterasyon Sayısı	Eğitim Süresi	Hassasiyet (%)	Ortalama Genel Hassasiyet (%)	Ortalama Genel Duyarlılık (%)	F1-Skoru (%)
1	5d	68,74	73,94	74,93	74,43
50	1s 47d 17s	89,99	92,54	83,45	87,76
100	3s 19d 40s	90,90	94,81	89,25	91,94
200	6s 48d 54s	92,70	95,51	93,57	94,53
400	13s 1d 43s	100	97,49	97,24	97,36

Çizelge 5.1’deki başarımların karşılaştırıldığında 400 iterasyon sonuçlarının diğer iterasyon sayılarına göre daha iyi çalıştığı gözlemlenmiştir. Başarımlardan ise ortalama genel hassasiyet değişkeni baz alınarak doğruluk %97,49 elde edilirken Hassasiyet değeri ise %100’e ulaşmıştır. Çizelge 5.1’de var olan süre sütunu incelendiğinde bu sütunda verilen süreler için s saati, d dakikayı, ve bir sonraki s harfi ise saniyeyi temsil etmektedir. Buna göre örneğin 100 iterasyon sayısına ait değerler incelenmek istenirse, eğitim süresi toplam 3 saat 19 dakika 40 saniye sürmüştür.

5.1. TENSORBOARD GÖRSELLEŞTİRME SONUÇLARI

Derin öğrenme projelerinde sıklıkla başarımların hakkında bilgi sahibi olarak modelin değerlendirilmesi sağlanır. TensorBoard ortaya çıkarılmadan önceki zamanlarda bu durum biraz daha insan gücüne dayalı iken şimdilerde ise TensorBoard sayesinde oldukça basit bir şekilde verilerin analizi yapılmaktadır. Kullanıcıların model yapılandırmalarını analiz etmelerini ve optimize etmelerini sağlayan TensorBoard görselleştirmesini entegre etmektedir [68]. TensorBoard’un orijinal web

sitesinde TensorBoard için makine öğrenimi deney sonuçlarını diğer kullanıcılar ile yüklemenizi ve paylaşmanızı sağlayan, yönetilen bir deneyim sunmaktadır [58].



Şekil 5.1. TensorBoard histogram görselleştirme görseli [60].

Şekil 5.1’de TensorBoard’un web sitesinden alınan örnek bir sinir ağı modelinin görselleştirilmesi sonucu alınan histogram örnekleri gösterilmektedir. TensorBoard, makine öğrenimi deneyi için gereken görselleştirmeyi ve araçları sağlar [58]:

1. Kayıp ve doğruluk gibi metrikleri izleme ve görselleştirme
2. Model grafiğini görselleştirme (işlemler ve katmanlar)
3. Ağırlıkların, yanlılıkların veya diğer tensörlerin histogramlarını zamanla değiştikçe görüntüleme
4. Görüntüleri, metinleri ve ses verilerini görüntüleme
5. TensorFlow programlarının profilini oluşturma

Tez çalışmasında TensorBoard’un çalıştırılabilir olması için öncelikle yüklü olması gerekmektedir. Aşağıdaki kod satırındaki gibi yükleme yapılması gerekmektedir.

pip install tensorboard

TensorBoard kütüphanesi yüklendikten sonra Jupyter Notebook platformunda Şekil 5.2’de yer alan eklentinin yüklenmesi sağlanmıştır.

```
#Tensorboard uzantısının yeniden yüklenmesi
%reload_ext tensorboard
```

Şekil 5.2. TensorBoard uzantısının yüklenme kodu.

Eğer TensorBoard’un yüklü olup olmadığı kontrol edilmek istenirse Şekil 5.3’teki gibi var olan kütüphanenin kontrolü yapılmaktadır. TensorBoard’un yüklenen sürüm ise Şekil 5.3’te görüldüğü üzere 2.4.1 versiyonuna aittir. Bununla birlikte, Şekil 5.3’te eğitim sonucu oluşan günlüklerin yüklenmesi için kod bulunmaktadır. Bu kod bloğu ile eğitim günlükleri görselleştirilmiştir.

pip show tensorboard

tensorboard günlüklerinin adresi

%tensorboard --logdir= "D:\bitewing\Mask_RCNN\samples\logs"

```
In [2]: pip show tensorboard

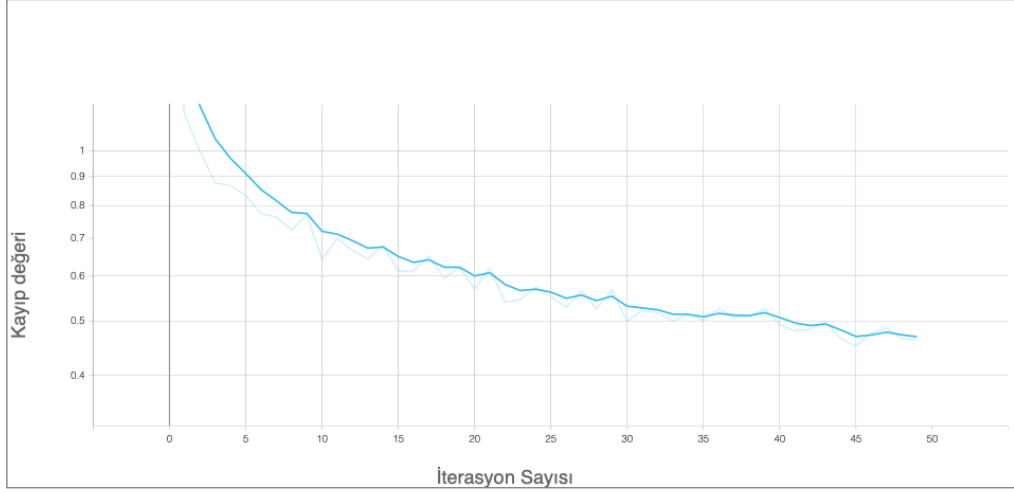
Name: tensorboard
Version: 2.4.1
Summary: TensorBoard lets you watch Tensors Flow
Home-page: https://github.com/tensorflow/tensorboard
Author: Google Inc.
Author-email: packages@tensorflow.org
License: Apache 2.0
Location: c:\users\windows10\anaconda3\envs\dentiass
Requires: requests, six, wheel, numpy, tensorboard-p
sl-py, google-auth, protobuf
Required-by: -ensorflow-gpu, tensorflow
Note: you may need to restart the kernel to use upda

In [3]: %tensorboard --logdir="D:\bitewing\Mask_RCNN\samples\
```

Şekil 5.3. TensorBoard günlüklerinin görselleştirme kodu.

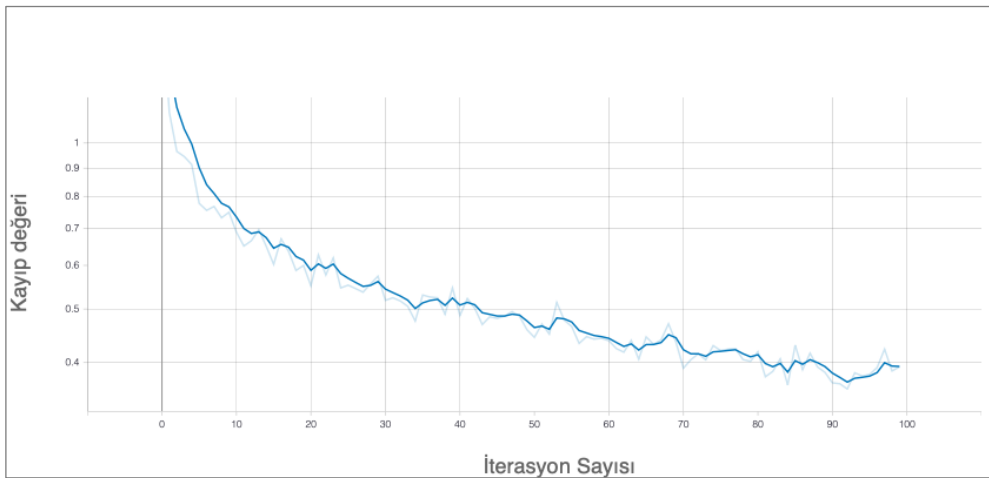
TensorBoard’un görselleştirme sayfası açıldığında içerisinde oluşan günlükler seçilerek 50, 100 ve 400 iterasyon sayısı için grafikler yazdırılmıştır. TensorBoard görselleştirmesi ile otomatik olarak bir sinir ağı modelinin eğitiminin ne kadar sürede

çalıştığı gözlemlenmektedir. Şekil 5.4'te 50 iterasyon sayısı için eğitilmiş günlüğe ait kayıp grafiği değerleri gösterilmektedir.



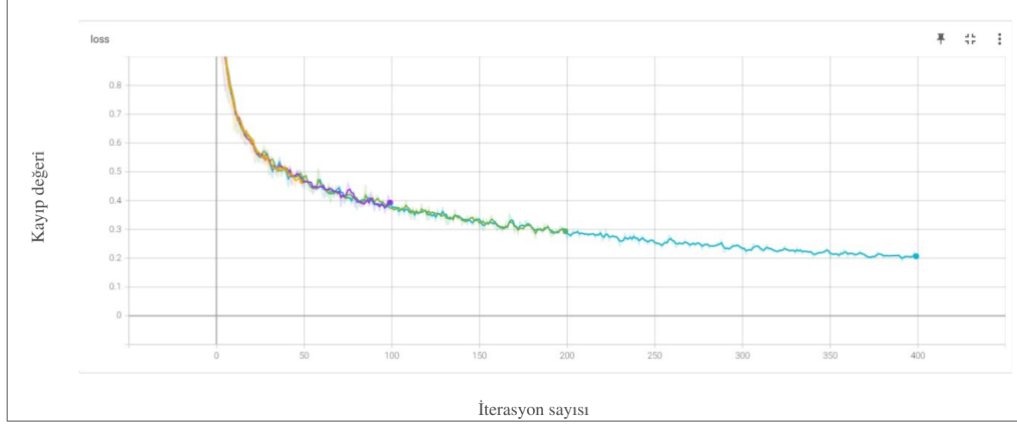
Şekil 5.4. Eğitilen sinir ağı sonucu 50 iterasyon sayısına ait kayıp grafiği.

Şekil 5.4'te oluşturulan grafikte x eksenini değerleri iterasyon sayılarına denk gelirken y eksenini doğrultusunda ise kayıp değerleri mevcuttur. Grafik detaylı olarak incelendiğinde sinir ağının doğruluğunun arttığını kayıp değerlerinin zamanla azalmasından anlaşılmaktadır. Şekil 5.5'te ise sinir ağının 100 iterasyon sayısına kadar eğitilmesiyle elde edilen kayıp grafiği gösterilmektedir. Şekil 5.5'te yer alan kayıp grafiği incelendiğinde 50 iterasyon sayısına ait grafiğe göre kayıp değerlerinin düşüşe geçtiği gözlemlenmiştir. Böylelikle doğruluğun arttığı sonucu çıkarılabilmektedir.



Şekil 5.5. Eğitilen sinir ağı sonucu 100 iterasyon sayısına ait kayıp grafiği.

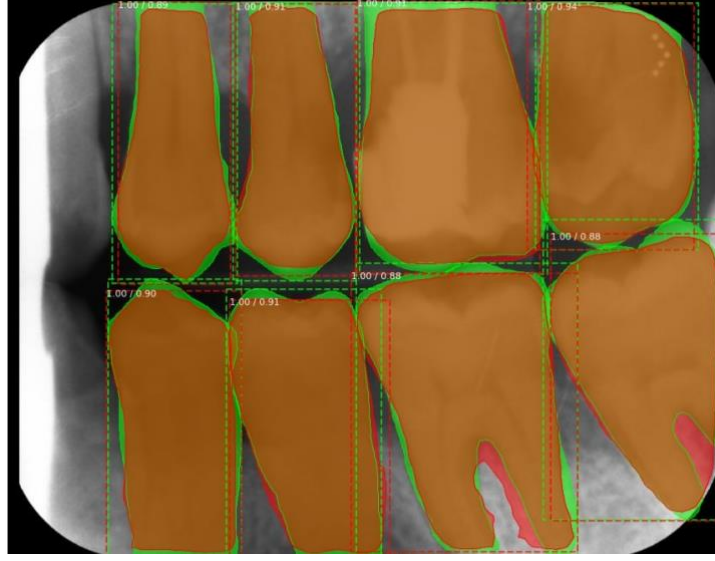
Çalışmada ek olarak, gerçek referans değerleri ve tahmin edilen sonuç maskeleri arasındaki kesişim ölçülerek ağın başarısı ölçülmüştür. Bunun için birleşim üzerinden kesişim (IOU) fonksiyonu kullanılarak örtüşme kutuları %50 eşik değeri ile hesaplanmıştır.



Şekil 5.6. Eğitilen sinir ağı sonucu 400 iterasyona kadar olan kayıp grafikleri.

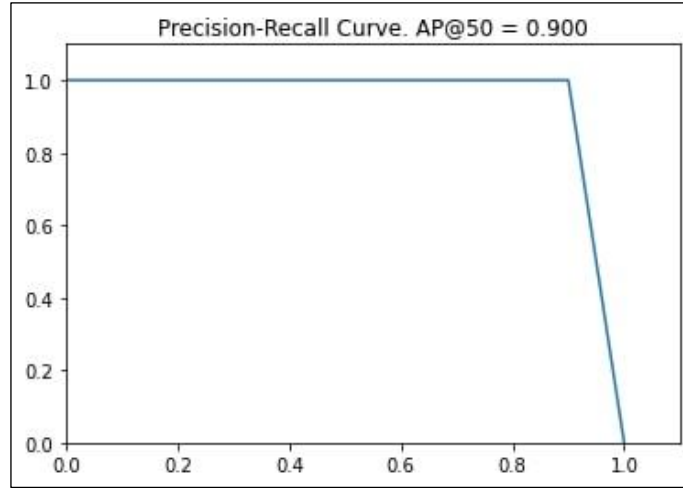
Şekil 5.6’da verilen görselde, 400 iterasyon sayısına kadar olan kayıp grafikleri yer almaktadır. Buna göre kayıp değerinin belirli bir iterasyondan sonra sabit bir değerde azaldığı çıkarımına varılmıştır. Bu sebeple, daha fazla iterasyona ihtiyaç duyulmamıştır. Şekil 5.7’de ise gerçek referans kutucukları ve maskeleri yeşil boyalı iken modelin tahmin ettiği sonuçlar kırmızı olarak boyanmıştır. Şekil 5.7’de verilen görselde toplam 8 adet dış maskesi verilmiştir. Görselde sol üstte verilen dış incelendiğinde iki adet puan değeri gösterilmektedir. Görseldeki dışın ilk puanı olan %100 gerçek referans değeri ile tahmin sonucunun kesişim puanını vermektedir. Bir sonraki puan ise bu iki maske için geçerli olan birleşim üzerinden kesişim değerini vermektedir. Elde edilen IOU değerinin hesaplanması Eşitlik 5.7’de gösterilmektedir.

$$\text{Birleşim üzerinden kesişim} = \frac{\text{Kesişim alanı}}{\text{Birleşim alanı}} \quad (5.7)$$



Şekil 5.7. IOU değerinin gerçek referans ve tahmin sonucunda görselleştirilmesi.

Eşitlik 5.4 ve Eşitlik 5.5'te hesaplanan hassasiyet ve duyarlılık değerlerinin örnek bir bitewing görüntüsü için hesaplatılmış grafiği ise Şekil 5.8'de gösterilmiştir.



Şekil 5.8. Hassasiyet ve duyarlılık değerlerinin örnek görsel için grafiksel gösterilmesi.

Şekil 5.8 incelendiğinde, örnek bir radyografinin %50 ortalama hassasiyet değeri üzerinde gerçek referans ve tahmin kutucukları karşılaştırılarak %90 eşleştiği sonucuna varılmaktadır. Böylelikle, modelin yüksek performansta çalıştığı gözlemlenmiştir.

BÖLÜM 6

SONUÇLAR VE ÖNERİLER

Bu çalışmada, bitewing radyografik görüntülerde evrişimli sinir ağı türünden olan Mask R-CNN kullanarak diş tespiti ve segmentasyonu gerçekleştirilmiştir. Mask R-CNN sonucunda diğer evrişimli sinir ağlarına göre sınırlayıcı kutu ve sınıf puanlarına ek olarak yüksek kaliteli segmentasyon maskeleri elde edilmiştir. Çalışmada nesne tespitinin yanı sıra segmentasyon yapılmasının en büyük sebebi, görsellerde birden fazla dişin anatomik olarak üst üste gelerek çakışma problemlerinde, dişlerin maskelenerek arka plandan ve diğer dişlerden ayırtlandırılmasıdır. Böylelikle diğer evrişimli sinir ağlarına göre daha yüksek performans göstermesi beklenmektedir. Dişlerin tespiti ve segmentasyonu süresince, Keras ve TensorFlow kütüphaneleri kullanılmıştır. Sonuçlar Windows 10 Pro işletim sisteminde Quadro RTX 5000 grafik kartı kullanılarak elde edilmiştir. Mask R-CNN ile kullanılan omurga ağı ise önceden eğitilmiş ResNet-101 olarak seçilmiştir.

Bu çalışmanın amacı, derin öğrenme teknikleri kullanılarak, bitewing radyografi görüntülerinde otomatik diş tespiti ve segmentasyonu yapmaktır. Çalışmada önerilen çözüm ile diş hekimlerinin radyografik analizler sırasında karar verme süreçlerinde destek sağlanması hedeflenmektedir. Tasarlanan sistem ile dişlerin analizi için yüksek hassasiyette performans sunulmuştur. Elde edilen sonuçlara dayanarak, sistemin derin öğrenme tabanlı otomatik diş tespiti ve segmentasyonu klinik uygulamalarda diş hekimleri tarafından doğrulanmıştır. Mask R-CNN için en yüksek başarı oranı, 400 iterasyon sayısı ile hassasiyet %100'e ulaşırken ortalama genel hassasiyet değeri %97,49 olarak elde edilmiştir.

Gelecek çalışmalar için daha karmaşık verilerde dişlerin tespiti ve segmentasyonu sağlanarak sinir ağı modelinin daha verimli sonuçlar elde etmesi sağlanabilir. Çalışmada eksik dişler, yanlış veri etiketlemesi, ağızda anatomik olarak asimetrisinin

var olması ve dişlerin çakışma oranlarının yüksek olduğu varsayılarak, gelecek çalışmalarda sezgisel bir yaklaşım ile bahsedilen ihtimallerin ortadan kaldırılması hedeflenmektedir. Bununla birlikte bitewing radyografik görüntülerde sadece dişlerin tespiti değil dişlerde çürük, implant ve dolgu tespiti gibi problemler tespit edilebilir. Çalışma sonucunda elde edilen diş sınıflarına ek olarak, evrensel bir bağlam etrafında toplanmak amacıyla ortak bir notasyon sistemi kullanılması gerektiğinden diş hekimlerinin sıklıkla kullandığı FDI numaralandırma sistemi ile dişlerin konumlarını belirlemek için sınıflandırma yapılabilir.

KAYNAKLAR

1. Kröger, E., Dekiff, M., and Dirksen, D., “3D printed simulation models based on real patient situations for hands-on practice”, *European Journal Of Dental Education*, Germany, Münster, 21 (4): e119–e125 (2017).
2. Williamson, G. F., “Intraoral radiography: Positioning and radiation protection”, *A Peer-Reviewed Publication, RDH* 26.12: 23 (2006).
3. İnternet: Medical Park, “X ışımı ile çalışan tıbbi cihazlar”, <https://www.medicalpark.com.tr/radyoloji/b-6725> (2021).
4. American Dental Association Council on Scientific Affairs, “The use of dental radiographs: update and recommendations”, *Journal of the American Dental Association*, vol. 137,9: 1304-12 (2006).
5. Berman, L., and Hargreaves, K., “Cohen's Pathways of the Pulp Expert Consult-E-Book”, *Elsevier Health Sciences* (2015).
6. Vandenberghe, B., Jacobs, R., and Bosmans, H., “Modern dental imaging: a review of the current technology and clinical applications in dental practice”, *European Radiology*, 20(11), 2637-2655 (2010).
7. Aps, J., “Extraoral Radiography in Pediatric Dental Practice”, *Imaging in Pediatric Dental Practice*, Springer, Cham, 31-49 (2019).
8. Gupta, A., Devi, P., Srivastava, R., and Jyoti, B., “Intra oral periapical radiography- basics yet intrigue: A review”, *Bangladesh Journal of Dental Research & Education*, 4.2, 83-87 (2014).
9. Serman, N., “Occlusal Radiology”, *Principles of Dental Imaging*, Langland and Langlais W&P, 158- 164 (2001).
10. Subramaniam, P., Kumar, K., Ramakrishna, T., and Bhadranna, A., “Bone regeneration with plasma-rich-protein following enucleation of traumatic bone cyst”, *European journal of dentistry*, 7(03), 377-381 (2013).
11. İnternet: Bitewing Radiographic Technique, “University of Toronto Faculty of Dentistry”, <https://iits.dentistry.utoronto.ca/bitewing-radiographic-technique> (2018).
12. Schwendicke, F. and Göstemeyer, G., “Conventional bitewing radiography”, *Clinical Dentistry Reviewed*, (2020).

13. Chan, M., Dadul, T., Langlais, R., Russell, D., and Ahmad, M., “Accuracy of extraoral bite-wing radiography in detecting proximal caries and crestal bone loss”, *The Journal of the American Dental Association*, 149(1), 51-58 (2018).
14. Vandenberghe, B., Jacobs, R., and Bosmans, H., “Modern dental imaging: a review of the current technology and clinical applications in dental practice”, *European Radiology*, 20(11), 2637-2655 (2010).
15. Suzuki, K., “Overview of deep learning in medical imaging”, *Radiological Physics and Technology*, 10(3), 257-273 (2017).
16. Miki, Y., Muramatsu, C., Hayashi, T., Zhou, X., Hara, T., Katsumata, A., and Fujita, H., “Classification of teeth in cone-beam CT using deep convolutional neural network”, *Computers in Biology and Medicine*, 80, 24-29 (2017).
17. Hung, K., Montalvao, C., Tanaka, R., Kawai, T., and Bornstein, M. M., “The use and performance of artificial intelligence applications in dental and maxillofacial radiology: A systematic review”, *Dentomaxillofacial Radiology*, 49(1), 20190107 (2020).
18. Mahoor, M. H., and Abdel-Mottaleb, M., “Classification and numbering of teeth in dental bitewing images”, *Pattern Recognition*, 38(4), 577-586 (2005).
19. Hosny, A., Parmar, C., Quackenbush, J., Schwartz, L. H., and Aerts, H. J., “Artificial intelligence in radiology”, *Nature Reviews Cancer*, 18(8), 500-510 (2018).
20. İnternet: DentiAssist Veri Etiketleme Platformu, “DentiAssist Uygulama Ekranı”, <http://app.dentiassist.com> (2021).
21. Cohen, S., and Hargreaves, K. M. (Eds.), “Cohen's pathways of the pulp”, *Mosby Elsevier* (2011).
22. Vandenberghe, B., Jacobs, R., and Bosmans, H., “Modern dental imaging: a review of the current technology and clinical applications in dental practice”, *European Radiology*, 20(11), 2637-2655 (2010).
23. Aps, J., “Extraoral Radiography in Pediatric Dental Practice”, *In Imaging in Pediatric Dental Practice*, pp. 31-49, Springer, Cham (2019).
24. Silva, G., Oliveira, L. and Pithon, M., “Automatic Segmenting Teeth in X-ray Images: Trends, a Novel Data Set, Benchmarking and Future Perspectives”, *Expert Systems with Applications*, vol. 107, pp. 15– 31 (2018).
25. He, K., Gkioxari, G., Dollár, P., and Girshick, R., “Mask r-cnn”, *In Proceedings of the IEEE international conference on computer vision*, pp. 2961-2969 (2017).
26. Oktay, A. B., “Tooth Detection with Convolutional Neural Networks”, *Biomedical Image Processing* (2017).

27. Tuzoff, D. V., Tuzova, L. N., Bornstein, M. M., Krasnov, A. S., Kharchenko, M. A., Nikolenko, S. I., and Bednenko, G. B., "Tooth detection and numbering in panoramic radiographs using convolutional neural networks", *Dentomaxillofacial Radiology*, 48(4), 20180051 (2019).
28. Ren, S., He, K., Girshick, R., and Sun, J., "Faster r-cnn: Towards real-time object detection with region proposal networks", *Advances in Neural Information Processing Systems*, 28, 91-99 (2015).
29. Muresan, M. P., Barbura, A. R., and Nedevschi, S., "Teeth Detection and Dental Problem Classification in Panoramic X-Ray Images using Deep Learning and Image Processing Techniques", *In 2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 457-463, IEEE (2020).
30. Leite, A. F., Van Gerven, A., Willems, H., Beznik, T., Lahoud, P., Gaêta-Araujo, H., and Jacobs, R., "Artificial intelligence-driven novel tool for tooth detection and segmentation on panoramic radiographs", *Clinical Oral Investigations*, 25(4), 2257-2267 (2021).
31. Poonsri, A., Aimjirakul, N., Charoenpong, T., and Sukjamsri, C., "Teeth segmentation from dental x-ray image by template matching", *In 2016 9th Biomedical Engineering International Conference (BMEiCON)*, pp. 1-4, IEEE (2016).
32. Zhu, G., Piao, Z., and Kim, S. C., "Tooth detection and segmentation with mask R-CNN", *In 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)* pp. 070-072 (2020).
33. Yasa, Y., Çelik, Ö., Bayrakdar, I. S., Pekince, A., Orhan, K., Akarsu, S., and Aslan, A. F., "An artificial intelligence proposal to automatic teeth detection and numbering in dental bite-wing radiographs", *Acta Odontologica Scandinavica*, 79(4), 275-281 (2021).
34. Mahoor, M. H., and Abdel-Mottaleb, M., "Classification and numbering of teeth in dental bitewing images", *Pattern Recognition*, 38(4), 577-586 (2005).
35. Valueva, M. V., Nagornov, N. N., Lyakhov, P. A., Valuev, G. V., and Chervyakov, N. I., "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation", *Mathematics and Computers in Simulation*, 177, 232-243 (2020).
36. Zhang, W., "Shift-invariant pattern recognition neural network and its optical architecture", *In Proceedings of Annual Conference of the Japan Society of Applied Physics* (1988).
37. Zhang, W., Itoh, K., Tanida, J., and Ichioka, Y., "Parallel distributed processing model with local space-invariant interconnections and its optical architecture", *Applied optics*, 29(32), 4790-4797 (1990).

38. Girshick, R., Donahue, J., Darrell, T., and Malik, J., “Rich feature hierarchies for accurate object detection and semantic segmentation”, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* pp. 580-587 (2014).
39. İnternet: The PASCAL Visual Object Classes Homepage, “Pascal VOC Challenges”, <http://host.robots.ox.ac.uk/pascal/VOC/> (2010).
40. İnternet: ILSVRC2013, “ImageNet Large Scale Visual Recognition Challenge 2013”, <https://image-net.org/challenges/LSVRC/2013/> (2013).
41. Girshick, R., “Fast r-cnn”, *In Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440-1448 (2015).
42. İnternet: Tryolabs Machine Learning, “Faster R-CNN: Down the rabbit hole of modern object detection”, <https://tryolabs.com/blog/2018/01/18/faster-r-cnn-down-the-rabbit-hole-of-modern-object-detection/> (2021).
43. Cai, Z., and Vasconcelos, N., “Cascade r-cnn: High quality object detection and instance segmentation”, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).
44. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., and Zitnick, C. L., “Microsoft coco: Common objects in context”, *In European Conference on Computer Vision*, pp. 740-755. Springer, Cham (2014).
45. Jader, G., Fontineli, J., Ruiz, M., Abdalla, K., Pithon, M., and Oliveira, L., “Deep instance segmentation of teeth in panoramic X-ray images”, *In 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 400-407 (2018).
46. He, K., Zhang, X., Ren, S., and Sun, J., “Deep residual learning for image recognition”, *In Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition* pp. 770-778 (2016).
47. L. Deng and D. Yu, “Deep Learning: Methods and Applications”, *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387 (2014).
48. Bahnsen, A. C., “Building ai applications using deep learning”, <https://blog.easysol.net/building-ai-applications> (2016).
49. İnternet: Keras Web Sitesi, “Keras”, <https://keras.io> (2021).
50. İnternet: TensorFlow Web Sitesi, “TensorFlow”, <https://www.tensorflow.org> (2021).
51. Gulli, A., and Pal, S., “Deep learning with Keras”, *Packt Publishing Ltd* (2017).
52. Etaati, L., “Deep Learning Tools with Cognitive Toolkit (CNTK)”, *In Machine Learning with Microsoft Technologies*, pp. 287-302 Apress, Berkeley, CA (2019).

53. Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., and Bengio, Y., Theano: a CPU and GPU math expression compiler. *In Proceedings of the Python For Scientific Computing Conference*, Vol. 4, No. 3, pp. 1-7 (2010).
54. Sivalingam, K., and Mujkanovic, N., “Graph compilers for AI training and inference” (2019).
55. Chollet, F., “*Python ile Derin Öğrenme*”, Birol Kuyumcu, *Buzdağı Yayınevi*, 978-605-69024-2-0 (2019)
56. İnternet: Kaggle Orijinal Web Sitesi, “Kaggle”, <https://www.kaggle.com> (2021).
57. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., and Zheng, X., “Tensorflow: A system for large-scale machine learning”, *In 12th {USENIX} symposium on operating systems design and implementation*, pp. 265-283 (2016).
58. İnternet: TensorBoard: TensorFlow'un görselleştirme araç takımı, “TensorBoard”, <https://www.tensorflow.org/tensorboard> (2021).
59. Lee, V. W., Kim, C., Chhugani, J., Deisher, M., Kim, D., Nguyen, A. D., and Dubey, P., “Debunking the 100X GPU vs. CPU myth: an evaluation of throughput computing on CPU and GPU”, *In Proceedings of the 37th annual international symposium on Computer architecture*, pp. 451-460 (2010).
60. Sanders, J., and Kandrot, E., “*CUDA by example: an introduction to general-purpose GPU programming*”, *Addison-Wesley Professional* (2010).
61. İnternet: NVIDIA, “Artificial Intelligence Computing Leadership from NVIDIA”, <https://www.nvidia.com/en-us/> (2021).
62. Lindholm, E., Nickolls, J., Oberman, S., and Montrym, J., “NVIDIA Tesla: A unified graphics and computing architecture”, *IEEE Micro*, 28(2), 39-55 (2008).
63. Garland, M., Le Grand, S., Nickolls, J., Anderson, J., Hardwick, J., Morton, S., and Volkov, V., “Parallel computing experiences with CUDA”, *IEEE micro*, 28(4), 13-27 (2008).
64. Chetlur, S., Woolley, C., Vandermersch, P., Cohen, J., Tran, J., Catanzaro, B., and Shelhamer, E., “cudnn: Efficient primitives for deep learning”, *arXiv preprint arXiv:1410.0759* (2014).
65. İnternet: Anaconda Orijinal Web Sitesi, “Anaconda Data Science Technology”, <https://www.anaconda.com> (2021).
66. İnternet: NumPy, “NumPy”, <http://numpy.org> (2021).
67. Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S., “Feature pyramid networks for object detection”, *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117-2125 (2017).

68. Nguyen, V., Dang, T., and Jin, F., “Predict saturated thickness using tensorboard visualization”, *Visualization in Environmental Sciences 2018* (2018).

ÖZGEÇMİŞ

Buse Yaren TEKİN ilk, orta ve lise öğrenimini aynı şehirde tamamladı. İzmir’de yer alan Alp Oğuz Anadolu Lisesi sayısal bölümünden mezun oldu. 2015 yılında Karabük Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü’nde öğrenime başlayıp 2019 yılında onur derecesi ile mezun oldu. 2017 yılında Anadolu Üniversitesi’nde Yönetim Bilişim Sistemleri Bölümü’nde öğrenim görmeye başladı. 2019 yılında Karabük Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda başlamış olduğu yüksek lisans programını, Karabük Üniversitesi Lisansüstü Eğitim Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı altında sürdürmektedir. 2019 yılında İsveç merkezli Global AI Hub şirketinde yapay zeka ve bilgisayarlı görü alanlarında içerikler üreterek 2020 yılına kadar blog yazarlığı yaptı. 2020 yılı itibariyle Karabük Üniversitesi Teknoloji Geliştirme Bölgesi’nde yer alan SimurgAI Yazılım Tasarım ve Danışmanlık Hizmetleri Limited Şirketi’nde Yapay Zeka Uzmanı olarak görev yapmaktadır. Aynı zamanda 2021 Mart ayı itibariyle Amerika’da yer alan Towards AI şirketinde stajyer pozisyonunda Makine Öğrenimi Mühendisi olarak görev yapmaktadır.