TEKNOFEST

HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

SAĞLIKTA YAPAY ZEKÂ YARIŞMALARI

(Bilgisayarlı Görüyle Abdomen (Karın) Bölgesi için Hastalık Tespiti Kategorisi)

PROJE DETAY RAPORU

TAKIM ADI: SEMRUK

BAŞVURU ID: 314834

İçindekiler

1. Proje Mevcut Durum Değerlendirmesi 1

1.1. Proje Sunuş Raporu Değerlendirmesi 1

1.2. Proje Sunumundan Sonra Yapılan Değişiklikler 2

1.3. Yapılan Çalışmalar 2

2. Özgünlük 5

3. Sonuçlar ve İnceleme 6

4. Deney ve Eğitim Aşamalarında Kullanılan Veri Setleri 8

5. Referanslar 9

1. Proje Mevcut Durum Değerlendirmesi

1.1. Proje Sunuş Raporu Değerlendirmesi

Yarışma teknik şartnamesinde belirtildiği üzere görüntüler DICOM formatında da paylaşılacaktır. Yarışma probleminin çözümü için bu format tercih edilecektir. Bu seçimin sebebi DICOM formatındaki görüntülerin 16 bit piksel değerlerinin [1] PNG formatındaki görüntülerin 8 bit piksel değerlerinden daha fazla detay içermesidir. Bu 16 bit piksel değerlerinin geliştirilen yapay zekâ modelinin ihtiyacı olan girdilere dönüştürülmesi için PyDicom kütüphanesi kullanılmıştır. Görüntülerden elde edilen PixelData [2] değerleri Rescale Slope [3], Rescale Intercept [4] ve Rescale Type [5] bilgileri ile Hounsfield Unit [6] değerlerine dönüştürülmüştür. Bu dönüşüm yapıldıktan sonra abdomen bölgesi için uygun olan pencerelemeler [7] yapılmıştır. Daha sonra elde edilen bu veriler 0, 255 arasına sıkıştırılarak görüntüler geliştirilen modelin ihtiyacı olan girdi şekillerine dönüştürülmüştür.

Derin öğrenme algoritmalarının test sırasında daha önce görmediği veri üzerindeki başarısı eğitim sırasında gördüğü veri miktarı ve çeşitliliği ile doğrudan ilişkilidir. Model eğitim setindeki görüntüleri ezberleyebilir, bu da test aşamasında farklı bir kaynaktan gelebilecek görüntülerin yanlış yorumlanması riskini taşımaktadır. Bu durumu engellemek için eldeki verileri çoğaltıp çeşitlendirecek augmentasyon metotları rastgele yapılacak yakınlaşma, uzaklaşma, döndürme, taşıma, çevirme, bulanıklaştırma, gürültü ekleme, parlaklık ve kontrast değiştirmedir. Bu işlemler için Albumentations [8] kütüphanesi kullanılmıştır.

Yapay zekâ modeli oluşturulurken Tensorflow [9] kütüphanesi kullanılmıştır.

Proje Sunuş Raporu hazırlanması sırasında taranan makalelerden elde edilen sonuçlara göre modeller genel olarak; bilgisayarlı görüde kendini ispat etmiş modellerin temel alınmasıyla oluşturulacak bir öznitelik çıkarma katmanı (feature extractor), tüm bu özelliklerin kombine edileceği bir öznitelik ağı (feature network) ve bu ağdan gelen bilgilerin yorumlanacağı tespit katmanından (detection head) oluşmaktadır. Bu tespit katmanının hem sınıflandırmayı hem de kutulama işlemini beraber yapması model hızı açısından önemlidir. Öznitelik çıkarma katmanında temel model olarak ImageNet veri seti üzerinde Top-1 doğruluk skoru en yüksek olan EfficientNetV2 [10] seçilmiştir. Öznitelik ağı ise makalelerde de önerildiği şekilde dikkat bloklarıyla takım tarafından oluşturulmuştur. Sonuç olarak tüm bu bilgileri esas alan bir model geliştirilmesine karar verilmiş ve özgün bir model geliştirmesi tamamlanmıştır.

Yarışma problemi esasında hem patolojilerin belirlenmesi için bir sınıflandırma hem de kutu lokasyonlarının belirlenmesi için regresyon içermektedir. Eğitim sırasında modelin belirtilen her iki problemde de dereceli alçalma (gradient descent) hesaplaması için daha önce obje tespiti için önerilen Focal Loss [11] ve Smooth L1 Loss [12] kombine şekilde kullanılmıştır. Optimizer olarak modelin genelleştirilmesi için adaptif bir metot olan AdamW [13] kullanılmıştır. Ölçüm metriği olarak ise yarışmada da metrik olarak kullanılan IoU kullanılmıştır.

Model eğitimi yapılırken takımlarla paylaşılan 1182 adet seri, eğitim ve test seti olarak ayrılmıştır. Eğitim için ayrılan verilerle eğitim yapılırken test için ayrılan verilerle doğrulama yapılmış ve bu sırada daha önce bahsedilen metrikte en yüksek değeri elde eden model kaydedilmiştir ve yarışma sırasında bu model kullanılacaktır.

1.2. Proje Sunumundan Sonra Yapılan Değişiklikler

Proje sunumundan sonra devam eden araştırmalar sonucu problem için üretilen çözümde bazı değişikliklere gidilmiştir.

Elimizdeki verileri çoğaltıp çeşitlendirecek augmentasyon metotları rastgele yapılacak yakınlaşma, uzaklaşma, döndürme, taşıma, çevirme, bulanıklaştırma, gürültü ekleme, parlaklık ve kontrast değiştirmedir. Bu işlemler yapabilmek için TensorFlow Keras kütüphanesinden Albumentations kütüphanesine geçilmiştir. Albumentations kütüphanesiyle yapılabilecek çoğaltma ve çeşitlendirme işlemlerinin daha fazla olması, daha önce benzer çözümlerde farklı firmalar tarafından kullanılmış olması, medikal resimler, uydu resimleri, endüstri uygulamaları gibi farklı alanlarda sınıflandırma, bölütleme ve nesne tespiti görevlerinde kullanabiliyor olması ve PyTorch ve Tensorflow ile entegre olabilmesi sebebiyle bu değişiklik yapılmıştır.

Öznitelik çıkarma katmanında EfficientNetV2 modelinin Large versiyonu seçilmiştir. Seçilme nedenine 1. bölümde değinilmiştir. EfficientNetV2L modeli 119 milyon parametreli bir evrişimli sinir ağı modeli olup, yapılan araştırmalarda bu ağların özellik çıkarıcı bloklar olarak kullanıldığı görülmüştür.

Öznitelik ağı kısmında en son katmandan sonra transformer [14] eklenerek bir self-attention katmanı oluşturuldu. Proje Sunuş Raporu hazırlanması sırasın da taranan makalelerde görüntüler için de transformerların kullanımının yaygınlaştığı ve başarılı sonuçlar elde edildiği görülmüştür. Bu sebeple transformerlar yarışma için geliştirilen çözüme eklenerek yapay zekâ modeli daha güçlü ve özgün hale getirilmiştir.

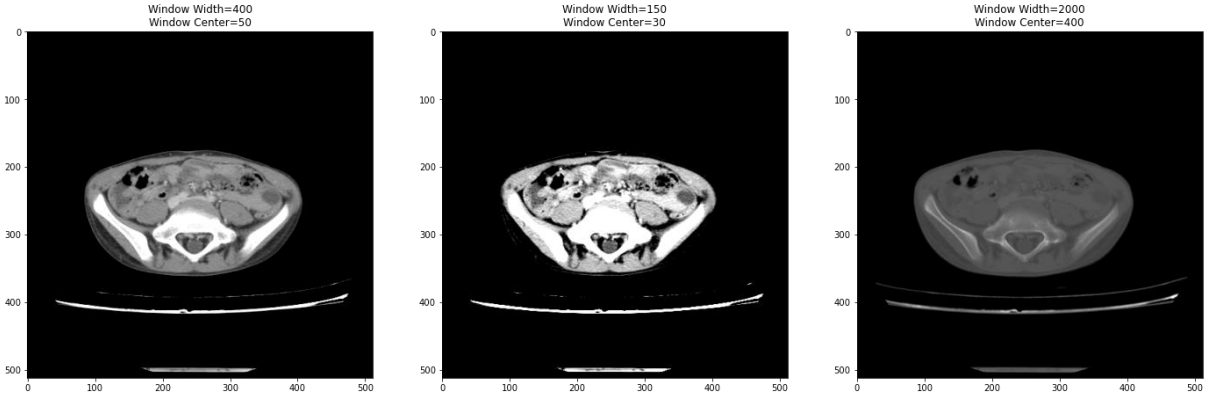
Loss fonksiyonu olarak bu tarz görevlerde çokça kullanıldığı görülen Focal Loss ve Smooth L1 Loss kombinasyonu kullanılmasına karar verilmiştir.

Son değişikliklerle birlikte elde edilen sonuçlar 3. bölümde sunulacaktır.

1.3. Yapılan Çalışmalar

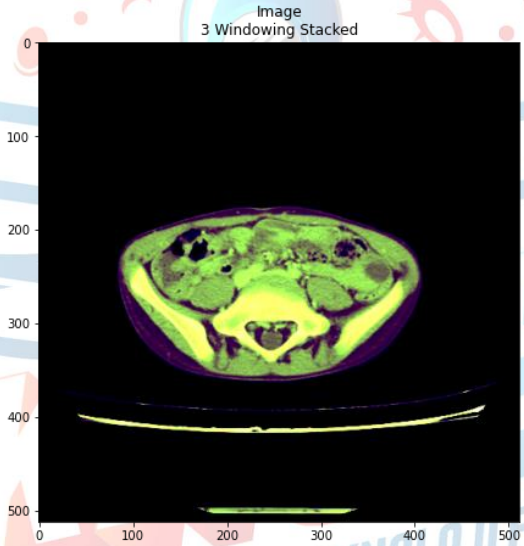
Sağlık Bakanlığı tarafından paylaşılan DICOM formatındaki görüntüler PyDicom kütüphanesi kullanılarak okunmuştur. Bu görüntülerden elde edilen piksel değerleri daha önce de belirtildiği gibi Rescale Slope, Rescale Intercept ve Rescale Type bilgileri ile Hounsfield Unit değerlerine dönüştürülmüştür. Elde edilen Hounsfield Unit değerlerine problemin çözümüne uygun pencerelemeler yapılmıştır. Yapılan pencelerelemelere ve elde edilen görüntülere örnek Şekil 1'de verilmiştir.

Yumuşak dokular için pencere genişliği 400, pencere merkezi 50, karaciğer için pencere genişliği 150, pencere merkezi 30 ve kalsifiye yapılar için pencere genişliği 2000, pencere merkezi 400 seçilerek görüntülere pencereleme işlemleri yapılmıştır [15].



Şekil 1. *Görüntülere uygulanan pencerelemeler ve pencereleme sonuçlarında elde edilen görüntüler*

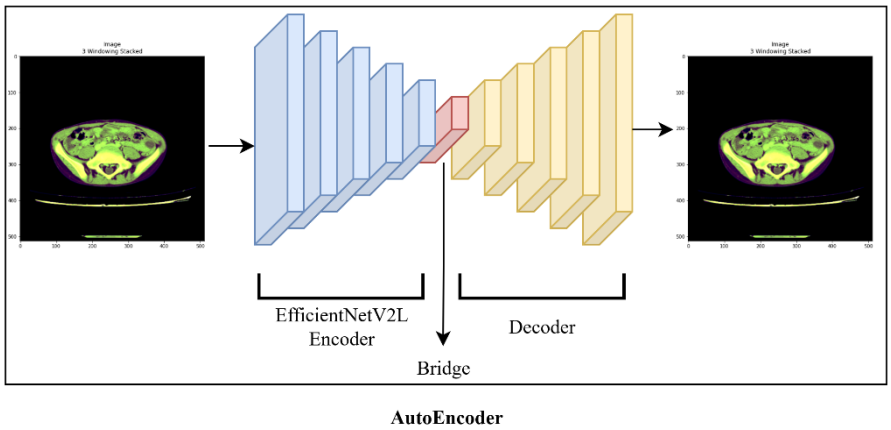
Bu işlemler sonucunda Şekil 1'de de görüldüğü gibi gri skalada 3 farklı görüntü elde edilmiş ve elde edilen bu görüntüler çakıştırılarak 3 katmanlı bir görüntü yapısı elde edilmiştir. Elde edilen bu 3 katmanlı görüntü Şekil 2'de verilmiştir.



Şekil 2. *Farklı pencereleme işlemleri sonucunda elde edilmiş 3 görüntünün çakıştırılması sonucu oluşturulan görüntü*

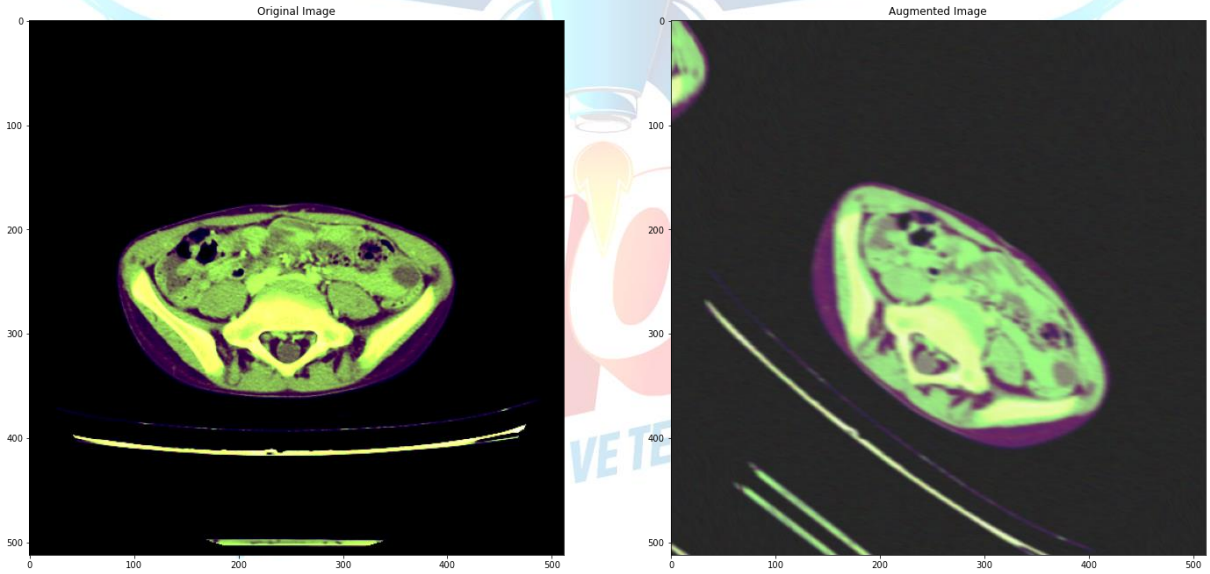
Görüntüler yukarıda bahsedilen ön işlemelerden sonra 512x512 piksel boyutuna yeniden boyutlandırılmıştır.

Modelin öznitelik çıkarma katmanında kullanılacağı belirtilen EfficientNetV2L modeline bir decoder bloğu eklenerek autoencoder hazırlanmış ve bu autoencoder 4. bölümde bahsedilecek DeepLesion veri seti ile eğitilmiştir. Bu eğitim sonucunda elde edilen ağırlıklar model içerisinde dondurularak model eğitimi hızlandırılmış ve ImageNet gibi hazır ağırlıklar kullanmak yerine probleme özgü ağırlıklar kullanılarak öznitelikler elde edilmiştir. Autoencoder yapısı Şekil 3'te gösterilmiştir.



Şekil 3. *Autoencoder mimarisi*

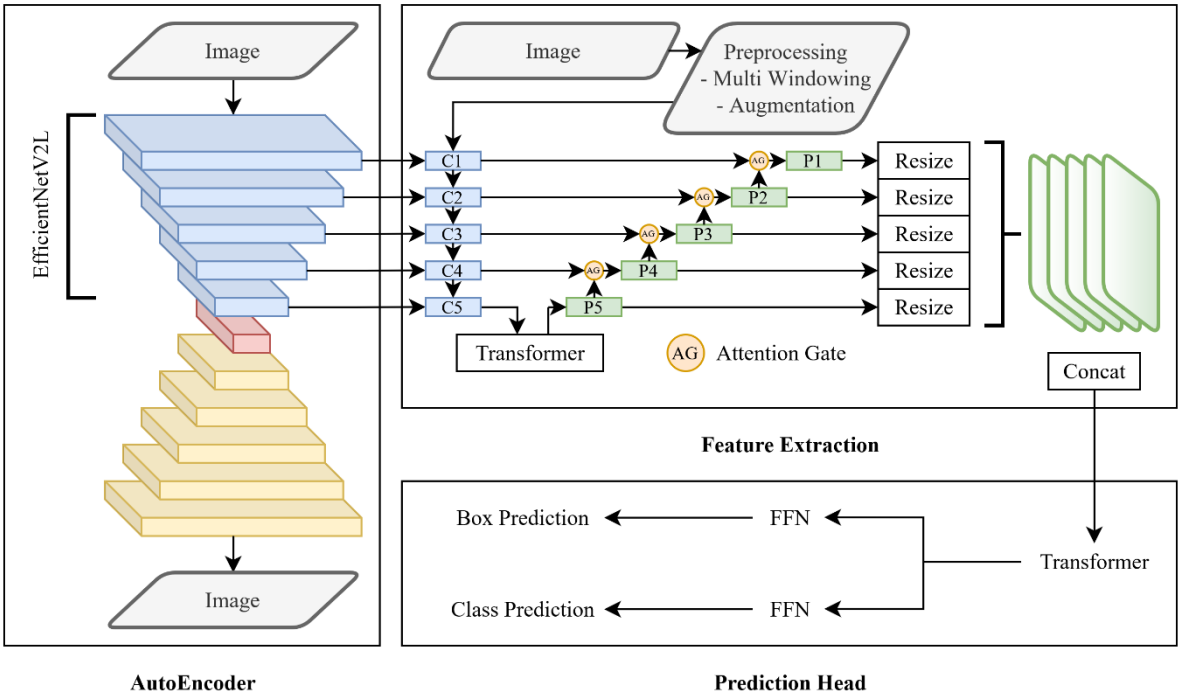
Daha önce de bahsedildiği gibi modelin eğitim setindeki görüntüleri ezberlemesini engellemek maksadıyla Albumentations kütüphanesi kullanılarak çeşitli augmentasyon metotları uygulanarak model eğitimi yapılmıştır. Belirtilen augmentasyon metotları yarışma problemi çözümüne de uygun olarak rastgele uygulanacak parlaklık, kontrast, renk doygunluğu, gama, bulanıklaştırma, gürültü ekleme, döndürme, çevirme, taşımadır. Bu yöntem uygulanarak elde edilmiş görüntü örneği Şekil 4'te gösterilmiştir.



Şekil 4. *Augmentasyon metotları sonucu elde edilen yeni görüntüler*

Proje Sunuş Raporu hazırlanması sırasında taranan makalelerden elde edilen sonuçlara göre modeller genel olarak; bilgisayarlı görüde kendini ispat etmiş modellerin temel alınmasıyla oluşturulacak bir öznitelik çıkarma katmanı (feature extractor), tüm bu özelliklerin kombine edileceği bir öznitelik ağı (feature network) ve bu ağdan gelen bilgilerin yorumlanacağı tespit katmanından (detection head) oluşmaktadır. Model öznitelik çıkarma katmanı temeli olarak kullanılan EfficientNetV2L modelinden 5 katman seçilmiştir. Böylece modelin görüntü hakkında daha fazla detay içeren sığ kısımları ve görüntü hakkında daha fazla öznitelik içeren derin kısımları da kullanılmıştır. Modelin en derin katmanına ise bir transformer bloğuyla self-attention [16] uygulanmış böylece modelin hastalık alanlarına odaklanması sağlanmıştır. Ayrıca takımın geçen sene yarışma modelinde kullandığı attention

katmanları [17] da öznitelik ağına eklenmiştir. Model derinleştikçe küçülen görüntü boyutları orijinal boyutlarına dönüştürülerek tüm bu görüntülere Concat işlemi uygulanmış böylece çıkarılan özniteliklerin çakışması sağlanarak modelin hastalık alanlarına odaklanması amaçlanmıştır. Tespit katmanında da transformer bloğunu takip eden ileri beslemeli bir sinir ağı kullanılarak hem patolojilerin belirlenmesi için sınıflandırma hem de kutu lokasyonlarının belirlenmesi için regresyon işlemi beraber yapılmıştır. Modelin basit hatlarıyla gösterimi Şekil 5'te yapılmıştır. Model daha önce bahsedilen kayıp fonksiyonu, optimizer ve metrikler kullanılarak eğitilmiştir. Eğitim sonuçları, test senaryoları ve sonuçları 3. Bölümde paylaşılacaktır.



Şekil 5. *Geliştirilen modelin basit hatlarıyla gösterimi*

TEKNO HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

2. Özgünlük

Takım, 2021 Sağlıkta Yapay Zekâ Yarışması'nda da kendine özgü bir model geliştirdiği gibi, bu yıl da hazır paketler ve modeller kullanmak yerine 1. bölümünde bahsedilen bilgiler ışığında kendi modelini geliştirmiştir. Ayrıca yine 1. bölümde belirtildiği gibi problem amacına uygun olarak iki farklı kayıp fonksiyonunun kombine edilerek yeni bir kayıp fonksiyonu oluşturulması özgünlüğe katkı sağlamaktadır.

Önceki Sağlıkta Yapay Zekâ Yarışması'nda, takım görüntü işlenmesi aşamasında farklı pencerelemelerle elde ettiği görüntüleri kombine ederek kullanmış ve bu yaklaşımın başarılı olduğunu görmüştür. Bu seneki yarışmada da yine aynı yöntemin kullanılmıştır.

Görüntülerden öznitelik çıkarmak maksadıyla 1. bölümde ayrıntılı olarak bahsedilen EfficientNetV2L modeline bir decoder bloğu eklenerek Autoencoder hazırlanmış ve bu autoencoder 4. bölümde bahsedilecek DeepLesion [18] veri seti ile eğitilmiştir. Daha sonra elde

edilen ağırlıklar dondurularak öznitelik çıkarma katmanında kullanılmış böylece modelin genel eğitim süresi kısaltılmış ve öznitelik çıkarma kapasitesi artırılmıştır.

Öznitelik ağı için de takımın geçen sene bölütleme işleminde de kullandığı attention katmanları kullanılmış ve sınıflandırma ile kutu sınırlarının tahmininin yapılacağı tahmin katmanı ise transformerlar ile geliştirilmiştir.

3. Sonuçlar ve İnceleme

Verilerin incelenmesi sırasında bazı görüntülerin görüntü boyutlarının 512x512 formatından farklı olduğu, bazı kutulamaların ise bu görüntü sınırlarının dışında yapıldığı ve tür etiketi sayılarında dengesizlik olduğu fark edildi. Böylece veri inceleme aşamasının önemi tekrar görülmüş oldu. Veri seti bu bilgiler ışığında temizlendi.

Eğitimden önce bounding boxlar ve tür etiketleri birleştirildi ve olgu numarası, kesit numarası ve birleştirme ile elde edilen listelerden yeni bir veri seti elde edildi. Bu veri setine ait örnek kesit Şekil 6'da verilmiştir.

	case_number	slice_number	bboxes
1440	11230/Seri4	25376445079	[[162, 184, 192, 205, 1]]
23188	25679/Seri3	37105096070	[[248, 218, 309, 284, 6]]
13636	22381/Seri4	37973428779	[[142, 212, 211, 290, 2]]
7551	16202/Seri4	32971753324	[[360, 275, 371, 290, 4]]
23499	25687/Seri4	38522861515	[[214, 240, 274, 299, 6]]

Şekil 6. *Eğitim için oluşturulan yeni veriseti*

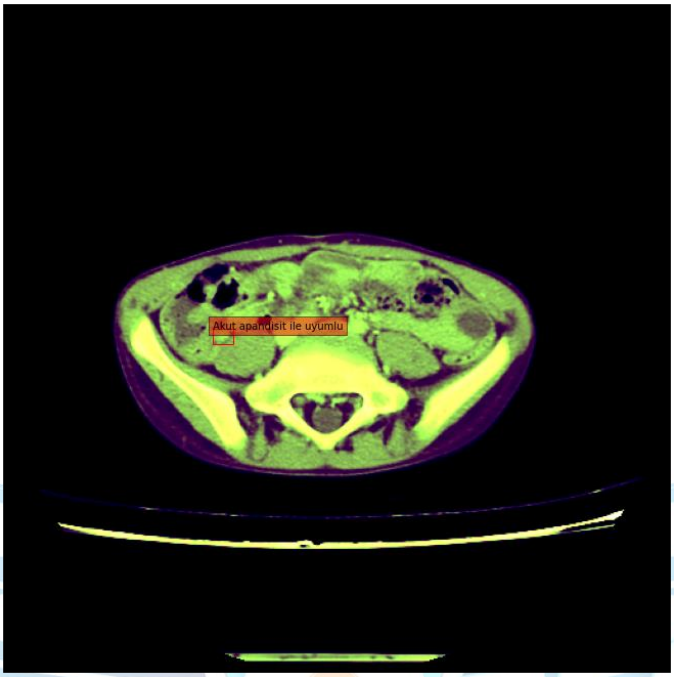
Eğitim sonucu yarışma sırasında da kullanılacak metriğe benzer şekilde IoU ile ölçülmüştür. IoU hesaplanırken tür etiketi için softmax aktivasyonu kullanıldığından çıkan sonuçlar argmax fonksiyonu ile 1 ve 6 arasındaki türlere dönüştürülmüştür. Bu metriğin hesaplanması için kullanılan kod bloğu Şekil 7'da gösterilmiştir.

```
def compute_score(box1, box2):

    box1_corners = box1[:4]
    box2_corners = box2[:4]

    lu = tf.maximum(box1_corners[:2], box2_corners[:2])
    rd = tf.minimum(box1_corners[2:], box2_corners[2:])

    intersection = tf.maximum(0.0, tf.math.subtract(rd, lu))

    intersection_area = tf.math.multiply_no_nan(intersection[0], intersection[1])

    box1_area = tf.math.multiply_no_nan(tf.math.subtract(box1_corners[0], box1_corners[2]),
                                        tf.math.subtract(box1_corners[1], box1_corners[3])
                                       )
    box2_area = tf.math.multiply_no_nan(tf.math.subtract(box2_corners[0], box2_corners[2]),
                                        tf.math.subtract(box2_corners[1], box2_corners[3])
                                       )
    union_area = tf.math.subtract(tf.math.add(box1_area, box2_area), intersection_area)

    iou = tf.clip_by_value(tf.math.divide_no_nan(intersection_area, union_area), 0.0, 1.0)

    box1_cls = box1[4]
    box2_cls = tf.math.argmax(box2[4:])

    cls_score = tf.cast(tf.math.equal(box1_cls, box2_cls), dtype=tf.float32)

    return tf.math.multiply_no_nan(iou, cls_score)
```

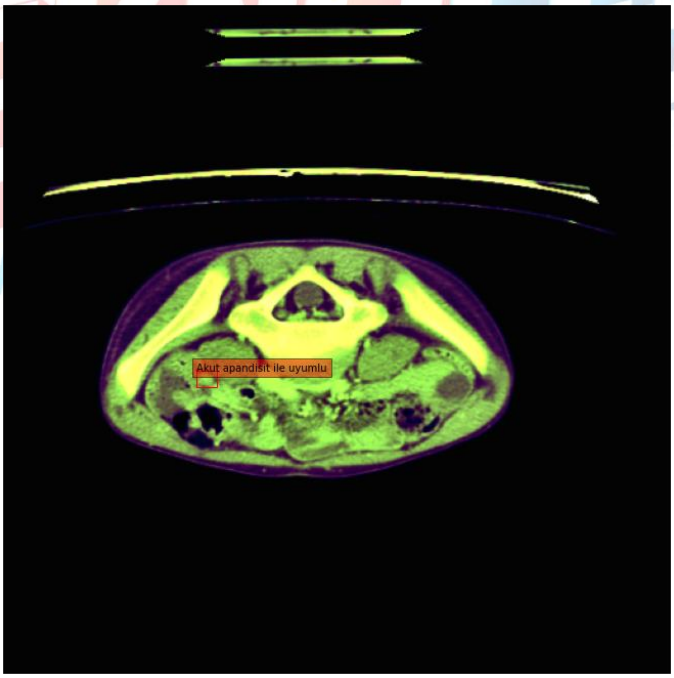
Şekil 7. *Yarışma metriği hesaplamak için kullanılan kod bloğu*

Model daha önceki bölümlerde de belirtildiği gibi takım tarafından geliştirilmiştir. Veri seti eğitim için %80 test için ise %20 olacak şekilde ayrıldı. Bu ayırma işlemi yapılırken tür

etiketleri oranının hem eğitim setinde hem de test setinde aynı olmasına dikkat edildi. Model eğitimi batch_size 1 seçilerek 100 epoch ile tamamlandı. Loss değeri test seti için 4.6302 elde edildi. Test setin üzerindeki ortalama yarışma metriği ise 0.7501 olarak elde edildi. Seçilen bir örneğin referans işaretlemesi Şekil 8'de, üzerinde augmentasyon işlemleri uygulandıktan sonra yapılan tahmin işleminin sonucu ise Şekil 9'da verilmiştir. Bu örneklerden yola çıkarak yapılan görsel doğrulamada da görüleceği gibi geliştirilen model farklı senaryolarda kabul edilebilir sonuçlar vermektedir. Ayrıca modelin bir kesit için tahmin süresi ortalama 75 ms kadardır.



Şekil 8. *Örnek resim referans işaretlemesi*



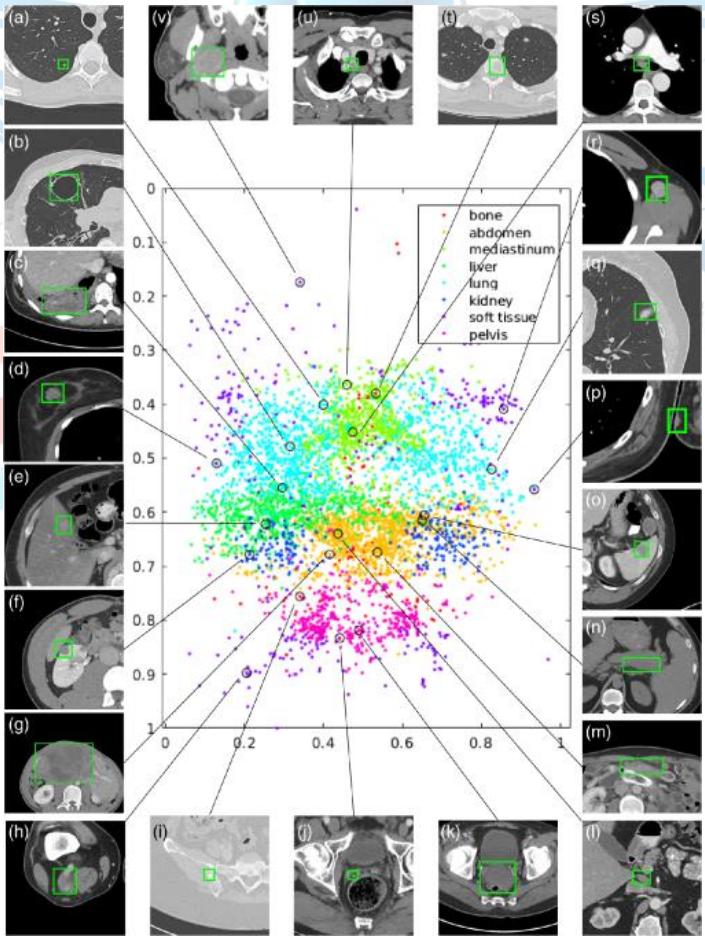
Şekil 9. *Örnek resimde yapılan augmentasyon işlemlerinden sonra model tahmini*

4. Deney ve Eğitim Aşamalarında Kullanılan Veri Setleri

Büyük ölçekli açıklamalı radyolojik görüntü veri kümelerinin çıkarılması, toplanması ve oluşturulması, oldukça önemli ancak zorlu bir sorundur. Aynı zamanda, tıbbi görüntü analizi için daha etkili veriye aç bilgi işlem paradigmaları (örneğin, derin öğrenme) tasarlamak için bir darboğazdır.

Ulusal Sağlık Enstitüleri (NIH) Klinik Merkezi, lezyonların tespit doğruluğunu iyileştirmesine yardımcı olmak için büyük ölçekli BT görüntülerini kamuya açık hale getirdi [19]. Kamuya açık tıbbi görüntü veri kümelerinin çoğunda binden az lezyon bulunurken, DeepLesion adlı bu veri kümesinde, BT görüntülerinde 32.000'den fazla etiketli lezyon (220 GB) tanımlanmıştır. DeepLesion, 4.427 benzersiz hastanın 10.594 çalışmasından 32.120 BT diliminde 32.735 lezyon içeren bir veri setidir. Bu veri setinde akciğer nodülleri, karaciğer tümörleri, genişlemiş lenf nodları vb. gibi çeşitli lezyon tipleri vardır. Bu veri seti yarışma probleminde tespit edilmesi beklenen lezyonları abdomen, karaciğer, yumuşak doku, böbrek lezyonları gibi genel başlıklar altında sınıflandırsa da çeşitli tıbbi görüntü uygulamalarında kullanılma potansiyeline sahiptir. Veri seti daha önce de belirtildiği gibi açık bilim girişimi kapsamında kamuya açık hale getirilmiştir.

Bu veri seti yarışma için geliştirilen problem çözümünde öznitelik çıkarmak için kullanılacak olan EfficientNetV2L mimarisi ile oluşturulmuş autoencoder eğitiminde kullanılmıştır. Böylece Sağlık Bakanlığı tarafından paylaşılan veri seti zenginleştirilerek modelin öznitelik çıkarma kapasitesi artırılmıştır.



Şekil 10. *DeepLesion veri seti*

5. Referanslar

1. pydicom.github.io/pydicom/stable/reference/generated/pydicom.dataset.Dataset.html

2. dicom.innolitics.com/ciods/ct-image/image-pixel/7fe00010

3. dicom.innolitics.com/ciods/ct-image/ct-image/00281053

4. dicom.innolitics.com/ciods/ct-image/ct-image/00281052

5. dicom.innolitics.com/ciods/ct-image/ct-image/00281054

6. Hounsfield, Godfrey N. "Computed medical imaging." Science 210.4465 (1980): 22-28.

7. Murphy, A., Baba, Y. Windowing (CT). Reference article, Radiopaedia.org. (accessed on 22 Mar 2022) https://doi.org/10.53347/rID-52108

8. albumentations.ai/docs/introduction/why_albumentations/

9. www.tensorflow.org/about?hl=en

10. Tan, Mingxing, and Quoc Le. "Efficientnetv2: Smaller models and faster training." *International Conference on Machine Learning*. PMLR, 2021.

11. Lin, Tsung-Yi, et al. "Focal loss for dense object detection." *Proceedings of the IEEE international conference on computer vision*. 2017.

12. Fu, Cheng-Yang, Mykhailo Shvets, and Alexander C. Berg. "RetinaMask: Learning to predict masks improves state-of-the-art single-shot detection for free." *arXiv preprint arXiv:1901.03353* (2019).

13. Loshchilov, Ilya, and Frank Hutter. "Decoupled weight decay regularization." *arXiv preprint arXiv:1711.05101* (2017).

14. Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).

15. Sheoran, Manu, et al. "DKMA-ULD: Domain Knowledge augmented Multi-head Attention based Robust Universal Lesion Detection." *arXiv preprint arXiv:2203.06886* (2022).

16. Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." *arXiv preprint arXiv:2010.11929* (2020).

17. Oktay, Ozan, et al. "Attention u-net: Learning where to look for the pancreas." *arXiv preprint arXiv:1804.03999* (2018).

18. Yan, Ke, et al. "Deeplesion: Automated deep mining, categorization and detection of significant radiology image findings using large-scale clinical lesion annotations." *arXiv preprint arXiv:1710.01766* (2017).

19. paperswithcode.com/dataset/deeplesion