TEKNOFEST

HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

SAĞLIKTA YAPAY ZEKÂ YARIŞMALARI

(Bilgisayarlı Görüyle Abdomen (Karın)

Bölgesi için Hastalık Tespiti Kategorisi)

PROJE DETAY RAPORU

GAiA-TECH

TAKIM ADI: GAiA-TECH

TAKIM ID: 398657

İÇİNDEKİLER

1. PROJE MEVCUT DURUM DEĞERLENDİRMESİ 3
2. ÖZGÜNLÜK 4
2.1. FİLTRELER 5
3. SONUÇLAR VE İNCELEME 6
3.1. ALGORİTMALARIN İNCELENMESİ 6
3.2. MODEL DOĞRULUĞU TEST AŞAMALARI 7
3.3. YAŞANILAN SORUNLAR VE ÇÖZÜMLERİ 9
3.4. SONUÇ 9
4. DENEY VE EĞİTİM AŞAMALARINDA KULLANILAN VERİ SETLERİ 10
5. REFERANSLAR 11

1. PROJE MEVCUT DURUM DEĞERLENDİRMESİ

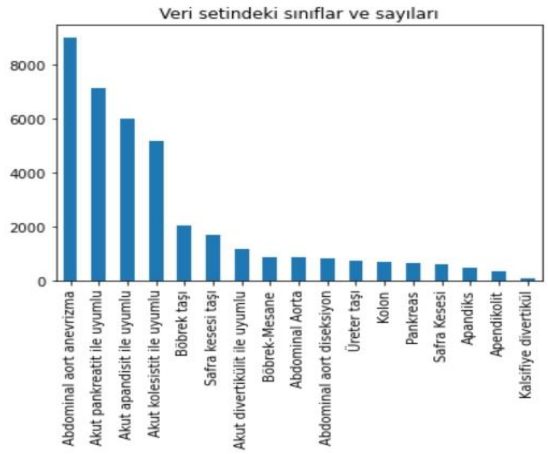
Yarışma amaçları doğrultusunda hastalıklar üzerinde çalışma yapılırken daha iyi bilgi sahibi olabilmek amacı ile literatür taramalarına devam edilmiştir. Bu bağlamda birçok makale incelenmiş ve hastalıklara özel özellikler üzerinde durulmuştur. Çalışmalarımızda oluşabilecek olası hatalar ve hastalıkları tanıma üzerine eksiklikler giderilmiştir.

Sağlık bakanlığı tarafından gönderilen veriler incelendiğinde veri seti içerisinde bulunan "Data.xlsx" dosyasında belirtilen etiketlerin dışında da mevcut görüntüler olduğu tespit edilmiştir. Şekil 1 Veri seti içerisinden örnekler alınarak sağlık alanında uzman görüşü ve desteği alınarak veriler kontrol edilmiştir. Tüm veri seti mevcut etiketlere göre düzenlenmiştir.

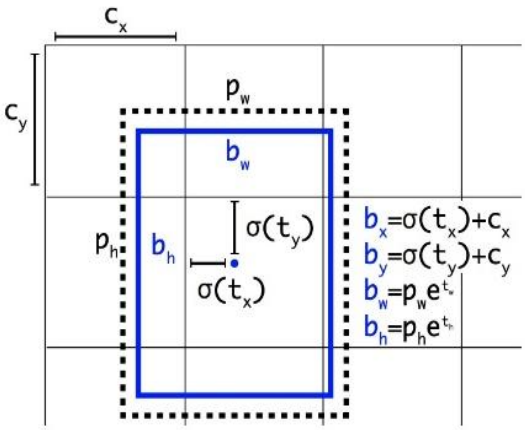
Sağlık Bakanlığı tarafından sağlanan veriler DCM formatında tarafımıza iletilmiştir. Bu formata uygun model eğitimi mümkün olmayacağından dolayı veriler üzerinde format dönüşümü uygulanmıştır. Veriler üzerinde DCM'den PNG'e dönüşümü yapılması amacıyla araştırmalar yapılmıştır. Birçok dönüşüm yöntemi denenmiştir ve görüntülerde bozulma olduğu tespit edilmiştir. Sorunun çözülmesi amacıyla eşik değerinde değişiklik yapılıp çalışmalara devam edilmiştir. Yapılan incelemelerde sonuçlar yetersiz görüldüğü için her fotoğrafa özgü eşik değeri hesaplaması yapılmış ve bu hesaplamalar sonucunda uygun eşik değerleri ile fotoğraflarda dönüşüm yapılmıştır.

Veri sitesi içerisinde bulunan "Data.xlsx" dosyasındaki etiketlerin dönüşümlerinin yapılması için algoritma tasarlanmıştır. Tasarlanan algoritmayı kullanarak teknofest tarafından takımımıza verilen ID'lere göre YOLO etiket formatında (Bkz. Resim 1) dönüşümü yapılmıştır. (Bkz. Tablo 1) Bu dönüşümün yapılması verilerin işlenmesini daha da kolaylaştırmaktadır. Etiketler ve veriler tek klasör içinde toplanıp daha kolay işlenebilir hale getirilmiştir.

Oluşturulmuş veri seti tek bir çatı altında toplanmıştır. Veri inceleme ve görselleştirme yöntemleri kullanılarak sınıf (class) sayısı ve etiket adedi tespit edilmiştir. Elde edilen sonuçlara Tablo 1'de yer verilmiştir.



Şekil 1: Veri Setindeki Sınıflar ve Sayıları



Resim 1: YOLOv4 Etiketleme Mantığı.

Tablo 1: Hastalıkların Etiket ID ve Adetleri

Hastalık İsmi	Etiket ID	Etiket Adedi
Akut Apandisit ile Uyumlu	1	5063
Akut Kolesistit ile Uyumlu	2	4692
Akut Pankreatit ile Uyumlu	3	6430
Böbrek Taşı Üreter Taşı	4	2049
Akut Divertikülit ile Uyumlu	5	1081
Abdominal Aort Anevrizma Abdominal Aort Diseksiyon	6	9767

Proje sunuş raporunda belirtilen Faster R-CNN modelinde beklenilen sonuca ulaşılamadığından yalnızca YOLOv4 algoritması kullanılması yönünde değişiklik yapılmıştır. ‘Sonuçlar ve İnceleme’ bölümünde algoritmalar çıktıları ile daha detaylı bir şekilde incelenmiştir.

2. ÖZGÜNLÜK

Tıbbi görüntüler elde etmek birçok prosedüre dayalı olduğundan ulaşılması oldukça zordur. Görüntüler elde edilmiş olsa dahi üzerinde etiketleme çalışmaları yapılırken tıp doktorlarının uzman girdisine ihtiyaç duyulmaktadır.[1] Tıbbi görüntüler üzerinde veri çoğaltma işlemi yapmak projenin bu aşamasında daha avantajlı olacaktır. Bu nedenle teknofest tarafından tarafımıza iletilen görüntüler üzerinde aşağıda belirtilen yöntemler ile veri çoğaltma işlemi yapılmıştır.

Genelleştirilebilir derin öğrenme modelleri elde etmek için büyük miktarda veriye ihtiyaç vardır. Geometrik düzenlemeler veya deformasyonlar tıbbi görüntü çoğaltılmasında oldukça yaygın olarak kullanılan yöntemlerdir. Derin sinir ağı modellerini eğitmek için örnek sayısını artırmak, veri kümelerinin boyutunu dengelemek ve bunların verimliliğini iyileştirmek için yaygın olarak kullanılmaktadır. Bu yöntemlerden histogram eşitleme, kontrastı veya parlaklığı arttırma, beyaz dengeleme, keskinleştirme ve bulanıklaştırma yöntemleri en popüler olanlardır. Bu yöntemler uygulama kolaylığı ve hızlılığından dolayı diğer veri çoğaltma yöntemlerine göre oldukça kullanışlıdır. Bu nedenle veri büyütme çalışmalarımızda bu yöntemlerden yararlanılmıştır.[2]

Özellikle derin öğrenme bağlamında kullanılan veri büyütme, daha fazla etiketli veri oluşturmanın basit bir yolunu sağlar; genellikle döndürme, çevirme, renk kayması veya görüntünün ve etiketin geçerliliğini koruyan diğer benzer iki boyutlu manipülasyonlar anlamına gelmektedir. Özgünlük adına veri seti üzerinde rastgele kontrast/parlaklık değişimi, gürültü ve bulanıklık, döndürme, perspektif projeksiyonu ve kumaş desenleri uygulanmıştır

2.1. FİLTRELER

Veri seti çoğaltılması yöntemleri arasında filtre uygulamaları da oldukça popülerdir. Yapılan araştırmalar sonucunda görüntüler üzerindeki gürültüyü azaltmak amacıyla gauss, median, wiener ve bilateral filtrelerinin uygulanabilirliği değerlendirilmiştir.[3] Veri seti üzerinde uygulamaları yapılmış ve aşağıda filtrelerin detaylı incelenmesi ve çıktıları sunulmuştur.

Gauss filtresi uygulanan görüntüdeki gürültüyü azaltmakta başarılıdır ancak kenarların korunmaması nedeniyle büyük miktarda ayrıntıyı kaybetme riski bulunmaktadır.[4] (Bkz. Resim 3) Bu sebeplerden dolayı tercih edilmemiştir.

Median filtresi doğrusal olmayan bir yöntemdir. Komşu pikselleri kullanılarak ortalama alınır ve yeni görüntü oluşturulmaktadır. Bu yöntem gürültü azaltılmasına yardımcı olsa da görüntü üzerindeki detayları bozmaktadır.[5] (Bkz. Resim 4) Böylece bulanık ve net olmayan bir görüntü elde edilmektedir. Bu uygulanan filtre neticesinde görüntüler üzerinde ciddi derecede piksel kaybı oluşmaktadır. Belirtilen sebeplerden dolayı bu filtrenin kullanılması tercih edilmemiştir.

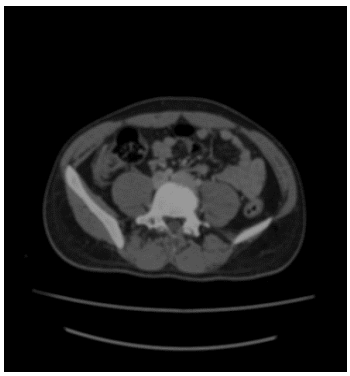
Wiener filtresi ters filtreleme yapmaktadır. Filtrenin çalışma mantığı ortalama kare hatasını en aza indirmektir.[6] Görseller üzerinde gürültüyü azaltma da başarılı olsa da görsel detaylarını olumsuz etkilemektedir. (Bkz. Resim 5) Bu sebeplerden dolayı veri çoğaltma yöntemlerinde kullanılması tercih edilmemiştir.

Bilateral filtresi görselin her bir piksel değerini komşu pikselleri ile ağırlık ortalamasını alarak değiştirmektedir.[7] Filtrenin mantığı, görüntü üzerindeki kenar detaylarını koruyup gürültüyü azaltmayı amaçlamaktadır. (Bkz. Resim 6) Gürültü azaltmada gösterdiği performans incelenmiş olup çalışmalarımıza uygun görülmüştür.

Resim 2: Orijinal Görsel

Resim 3: Gauss Filtresi Uygulanmış Görsel

Resim 4: Median Filtresi Uygulanmış Görsel

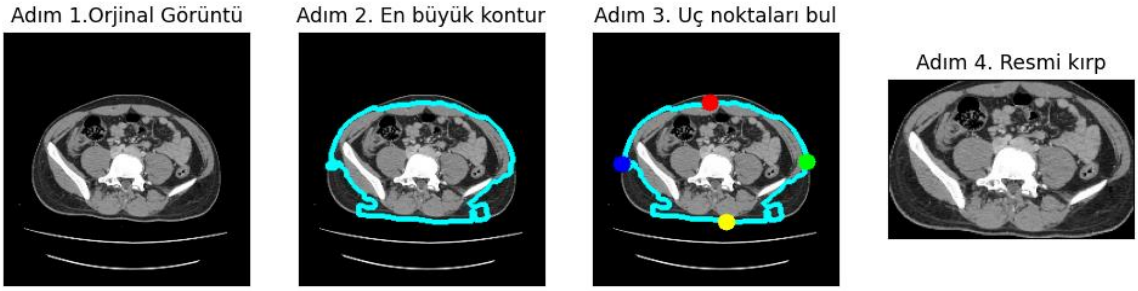


Resim 5: Winer Filtresi Uygulanmış Görsel

Resim 6: Bilateral Filtresi Uygulanmış Görsel

Resim 7: Kontrast İşlemi Uygulanmış Görsel

Veri çoğaltma işlemi yapılırken kullandığımız yöntemlerden bir diğeri de görüntü üzerinden siyah alanların çıkarılmasıdır. Yapılan işlemin amacı veri çeşitlerinin arttırılmasıdır. Bu yöntemde ilk aşama olarak görüntü üzerinde en büyük kontur bulunur. Konturun en uç noktalarını hesaplayarak ilgi alanı elde edilir ve kırpılır. Kırpılmış görüntü kaydedilerek veri sayısı iki katına çıkarılmıştır. (Bkz. Resim 8)



Resim 8: Görsellerden Siyah Alanların Çıkarılması İşlemi

3. SONUÇLAR VE İNCELEME

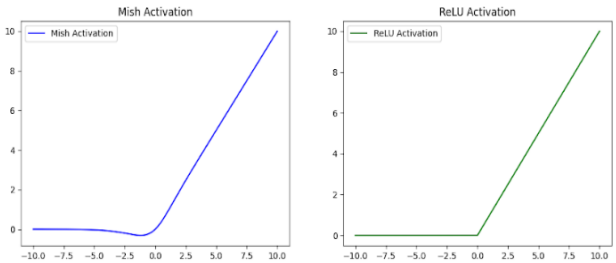
Proje sunuş raporunda belirtildiği gibi çalışmalarımızda YOLOv4, YOLOv3, Faster R-CNN ve SSD algoritmaları denenmiştir. Model incelemeleri ve sonuçları aşağıdaki paragraflarda detaylıca açıklanmış olup kullanılacak olan algoritma belirtilmiştir. Model inceleme yöntemleri ile incelendiğinde de çıkan değerlere tablo 2'de yer verilmiştir.

3.1. ALGORİTMALARIN İNCELENMESİ

SSD algoritmasının arkasındaki temel mantık, sabit boyutlu sınırlayıcı kutu tahmini üretmektir. Bu kutulardaki nesne sınıfı örneklerinin varlığı için puanlar verir ve en iyi tahminleri çıktı olarak veren ileri beslemeli evrişimsel ağ kullanılmaktadır.[8] eklemektedir. SSD, sınıf başına ortalama 8732 tahmin üretirken; YOLO, sınıf başına yalnızca 98 tahmin üretmektedir. Bu durumlardan dolayı nesneler üzerinde SSD algoritmasının başarı oranı düşük çıkmaktadır. Çok tahminde bulduğundan kaynaklı hız açısından olumsuz sonuçlar alınmıştır.[8] SDD modeli bu sebeplerden dolayı tercih edilmemiştir.

Faster R-CNN iki aşamadan oluşan nesne tespiti algoritmasıdır. İlk aşamada bölge teklif önerisiyle nereye bakılacağını bildiren derin ve tam evrişimsel katman bulunmaktadır. İkinci aşamada önerinden bölgeleri kullanan hızlı R-CNN algılayıcısı bulunmaktadır.[9] Faster RCNN algoritmasının çalışma prensibi bölge teklifinde bulunarak nesne tespiti yapmasıdır.[10] Yarışma kapsamında abdomen bölgesinde çalışıldığından dolayı Faster R-CNN algoritması birden fazla teklifte bulunduğu görülmüştür. Bu durum hastalıkların yanlış tespit edilmesine neden olduğu ve doğruluk oranının düştüğü tespit edilmiştir. Bu sebeplerden dolayı çalışmalarda kullanılması tercih edilmemiştir.

YOLOv3 ReLU aktivasyon fonksiyonu kullanmaktadır. Bu durum olumsuz sonuçlanmaktadır. YOLOv4 gelenekselleşen ReLU aktivasyon fonksiyonu yerine Mish fonksiyonu kullanmaktadır.[11] Bu durum modelin kesin kurallar yerine hata payı bırakarak eşleşme değerinin artmasını sağlamaktadır.



Resim 9: Mish ve ReLU Aktivasyon Fonksiyonlarının Garfikleri.

YOLOv4 algoritması 416x416 çözünürlüğünde görseli girdi olarak kullanmaktadır.[12] YOLO'nun nesne tespiti algoritmasının en büyük avantajlarından biri görüntüyü ızgaralara bölerek her ızgara için puanlama yapmaktadır. Sınırlayıcı kutular tahminlerin hesaplanması sonucu oluşmaktadır.[13] Buna göre YOLO her ızgara için ayrı bir tahmin vektörü oluşturmaktadır. Sınıflandırma ile farklı ölçeklere ve boyutlara sahip çoklu sınırlayıcı kutular tahmin edilmektedir. YOLOv4 algoritmasının yukarda belirtilen nedenlerden dolayı yarışma senaryosuna en uygun model olduğu tespit edilmiştir. Bu nedenden dolayı çalışmalarımızda YOLOv4 algoritması kullanılacaktır.

Tarafımıza iletilen veri setinde etiketli olan görseller 29.082 adettir. Özgünlük kısmında bahsedilen veri çoğaltma yöntemleri uygulanarak veri seti zenginleştirilmiştir. Veri çoğaltma yöntemleri sonrasında veri sayımız 8 katına çıkarılarak 232.656 adet veriye ulaşılmıştır. Verilerimizi training, validation ve test olmak üzere 3 ayrı bölüme ayrılmıştır. Verilerin eşit dağılması için algoritma oluşturulmuştur. Bu algoritma ile dağınık bir şekilde bölümlere oluşturulmuştur. Oluşturulan bölümlerin oranları Şekil 3'de belirtilmiştir. Model eğitirken 6500 epoch yapılmıştır. Modelin ezberleme ihtimaline karşı early stop kullanılmıştır.[14]

3.2. MODEL DOĞRULUĞU TEST AŞAMALARI

Değerlendirme kriterleri olarak, karışıklık matrisi (confusion matrix), duyarlılık (recall), hassasiyet (precision), F1 puanı (F1-score) ve kesişim bölgeleri (IoU) kullanılacaktır.[15]

Karışıklık Matrisi: Karışıklık matrisi, bir sınıflandırma algoritmasının performansını tanımlamak için kullanılan tablodur. Karşılık matrisi, bir sınıflandırma algoritmasının performansını görselleştirir ve özetlemektedir. Karışıklık matrisi, sınıflandırıcının ölçüm metriklerini tanımlamak için kullanılan dört temel özellikten oluşmaktadır. Bu özellikler şunlardır:

TP (True Positive/ Gerçek Pozitif): Gerçek değerleri 1 olarak sınıflandırılmaktadır. Gerçek pozitif, 1 olarak sınıflandırılan ve gerçekten 1 olan değerlerin sayısıdır.

TN (True Negative/ Gerçek Negatif): Gerçek olmayan değerleri 0 olarak sınıflandırılmaktadır. 0 olarak sınıflandırılan ve gerçekten de 0 olan değerlerin sayısıdır.

FP (False Positive/ Yanlış Pozitif): 1 olarak sınıflandırılan fakat gerçekte 0 olan değerlerin sayısıdır.

FN (False Negative/ Yanlış Negatif): 0 olarak sınıflandırılan fakat gerçekte 1 olan değerlerin sayısıdır.

Bir algoritmanın performans ölçütleri, yukarıda belirtilen TP, TN, FP ve FN temelinde hesaplanan doğruluk, hassasiyet, duyarlılık ve F1 puanıdır.

Doğruluk, sınıfların ortalama hassasiyet (AP) değerlerinin aritmetik ortalaması (mAP) alınarak bulunmaktadır.[16] Denklem 1'de formülleri gösterilmektedir.

$$\text{AP} = \left(\frac{1}{n}\right)\sum_{Recall_i} Precision(Recall_i)=1 \qquad \text{mAp}=\sum_{c\in classes} \frac{\#TP(c)}{\#TP(c) + \#FP(c)} \quad (1)$$

Duyarlılık, isabet oranı olarak bilinmektedir ve yüksek olması beklenmektedir. Sınırlayıcının ne kadar TP ya da TN değeri doğru tahmin ettiğinin bir ölçüsüdür. Denklem 2'de duyarlılığın nasıl hesaplandığı gösterilmiştir.

$$\text{Duyarlılık} = \frac{TP}{TP+FN} \quad (2)$$

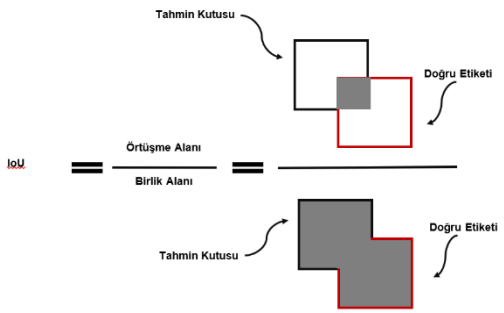
Hassasiyet, pozitif olarak sınıflandırılanlar arasında doğru sınıflandırılmış örneklerin oranını değerlendirir. Duyarlılık gibi hassasiyetin de yüksek olması beklenmektedir. Denklem 3'de hassasiyetin nasıl hesaplandığı gösterilmiştir.

$$\text{Hassasiyet} = \frac{TP}{TP+FP} \quad (3)$$

F-1 puanı, sınıflandırıcının ne kadar iyi performans gösterdiğinin bir ölçüsüdür. Duyarlılık ve hassasiyetin harmonik ortalamasıdır. Denklem 4'de f-1 puanının nasıl hesaplandığı gösterilmiştir.

$$\text{F-1} = 2\ x\ \frac{Hassasiyet\ x\ Duyarlılık}{Hassasiyet+Duyarlılık} \quad (4)$$

IoU, her bir sınırlayıcı kutu için, tahmin edilen sınırlayıcı kutu ile gerçek sınırlayıcı kutu arasındaki örtüşmeyi ölçmek için kullanılır. Şekil 2'de yapı gösterilmektedir.



Şekil 2: IoU Hesaplama Mantığı.

Tablo 3: Model Sonuçlar Tablosu

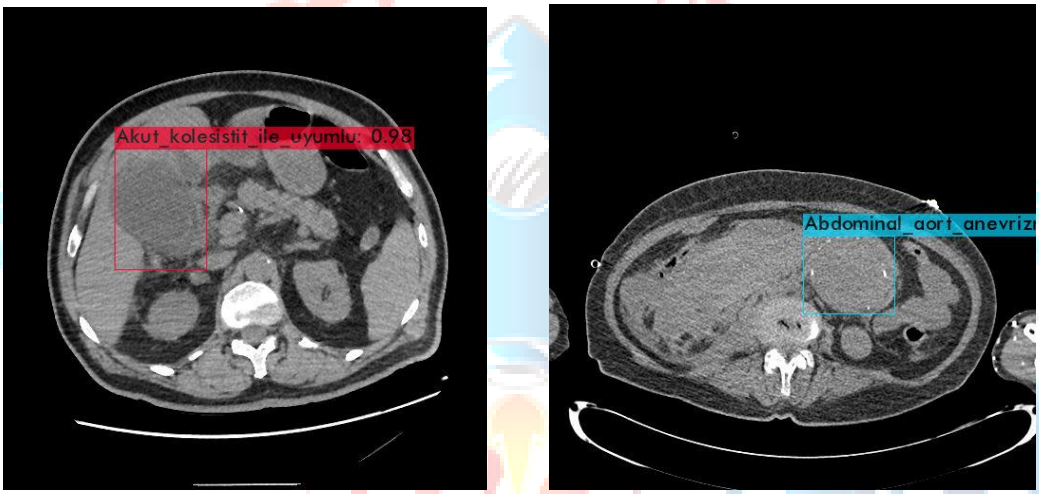
Model	Doğruluk	Doğrulama Doğruluğu	Kayıp	Doğrulama Kaybı
YOLOv4	0.8256	0.8456	0.7664	0.7125
YOLOv3	0.7435	0.8434	0.7046	0.6994
Faster R-CNN	0.7258	0.7562	0.7498	0.8325
SSD	0.7151	0.7895	0.6895	0.7451

3.3. YAŞANILAN SORUNLAR VE ÇÖZÜMLERİ

Yapay zekâ destekli hastalık tespiti çalışmalarında karşılaşılan en büyük sorun veri yetersizliğidir. Bu bağlamda sorunu çözmek amacıyla veri seti arayışına girilmiş ancak "Deney ve Eğitim Aşamalarında Kullanılan Veri Setleri" başlığı altında detaylıca bahsedildiği üzere yeni veri setine ulaşılamamıştır. Veri yetersizliği sorununu çözmek amacıyla özgünlük başlığı altında belirtilen yöntemler ile veri seti büyütülmüştür.

3.4. SONUÇ

Modelin sonuçları ve tablolar incelediğinde %82 doğruluk oranı ile YOLOv4 modelinin başarılı olduğu düşünülmektedir. Modelin test çıktıları bkz. resim 10' da belirtilmiştir. Bu aşamadan sonra model değişikliği yapılmayacaktır. Yarışma zamanına kadar yeni veri çoğaltma teknikleri kullanılarak model geliştirilecek ve test edilecektir. En yüksek doğruluk oranı alınması için çalışmalarımıza devam edilecektir.



Resim 10: Etiketli Örnek Veri

TEKNOFEST İSTANBUL HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

4. DENEY VE EĞİTİM AŞAMALARINDA KULLANILAN VERİ SETLERİ

Yapay zekâ tabanlı hastalık tespiti üzerine makaleler incelenirken birçok veri seti bulunmuştur. Veri setlerinin uygunluğu uzman desteği alınarak değerlendirilmiştir. Yarışma şartnamesinde belirtilen hastalık sınıflarına uygun veri seti bulunamadığından açık kaynak veri seti çalışmalarımızda kullanılmamıştır. Bölgede bulunan hastanelerden doktorlar ile iletişime geçilerek veri seti konusunda talepte bulunulmuştur. Etik onayı vb. prosedürden dolayı istenilen şartlara uygun veri setine ulaşılamamıştır. Bu nedenlerden dolayı teknofest tarafından sağlanan sağlık bakanlığı veri seti üzerinde özgünlük kısmında belirtildiği şekillerde veri çoğaltma işlemleri uygulanmıştır.

Verilerimizi training, validation ve test olmak üzere 3 ayrı bölüme ayrılmıştır. Bölümlerinin oranları şekil 3 belirtilmiştir.

Şekil 3: Veri Dağılımları

5. REFERANSLAR

[1] H. Liu *et al.*, "A new data augmentation method based on local image warping for medical image segmentation," *Medical Physics*, vol. 48, no. 4, pp. 1685–1696, Apr. 2021, doi: 10.1002/MP.14651.

[2] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," *2018 International Interdisciplinary PhD Workshop, IIPhDW 2018*, pp. 117–122, Jun. 2018, doi: 10.1109/IIPHDW.2018.8388338.

[3] S. Senthilraja, P. Suresh, and M. Suganthi, "Noise Reduction in Computed Tomography Image Using WB – Filter," *International Journal of Scientific & Engineering Research*, vol. 5, no. 3, 2014, Accessed: Jun. 28, 2022. [Online]. Available: http://www.ijser.org

[4] R. G. Lopes, D. Yin, B. Poole, J. Gilmer, and E. D. Cubuk, "Improving Robustness Without Sacrificing Accuracy with Patch Gaussian Augmentation," Jun. 2019, doi: 10.48550/arxiv.1906.02611.

[5] S. Luo, A. Peng, H. Zeng, X. Kang, and L. Liu, "Deep Residual Learning Using Data Augmentation for Median Filtering Forensics of Digital Images," *IEEE Access*, vol. 7, pp. 80614–80621, 2019, doi: 10.1109/ACCESS.2019.2923000.

[6] M. Poomani, J. Sutha, and K. R. Soundar, "Wiener filter based deep convolutional network approach for classification of satellite images," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 7, pp. 7343–7351, Jul. 2021, doi: 10.1007/S12652-020-02410-3/FIGURES/10.

[7] Z. Pei, H. Xu, Y. Zhang, M. Guo, and Y. Yee-Hong, "Face Recognition via Deep Learning Using Data Augmentation Based on Orthogonal Experiments," *Electronics 2019, Vol. 8, Page 1088*, vol. 8, no. 10, p. 1088, Sep. 2019, doi: 10.3390/ELECTRONICS8101088.

[8] W. Liu *et al.*, "SSD: Single shot multibox detector," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9905 LNCS, pp. 21–37, 2016, doi: 10.1007/978-3-319-46448-0_2/FIGURES/5.

[9] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9906 LNCS, pp. 443–457, 2016, doi: 10.1007/978-3-319-46475-6_28/TABLES/5.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[11] D. Misra, "Mish: A Self Regularized Non-Monotonic Activation Function," Aug. 2019, doi: 10.48550/arxiv.1908.08681.

[12] A. Ammar, A. Koubaa, M. Ahmed, A. Saad, and B. Benjdira, "Vehicle detection from aerial images using deep learning: A comparative study," *Electronics (Switzerland)*, vol. 10, no. 7, pp. 1–31, 2021, doi: 10.3390/electronics10070820.

[13] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," 2020, [Online]. Available: http://arxiv.org/abs/2004.10934

[14] L. Prechelt, "Early stopping - But when?," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7700 LECTURE NO, pp. 53–67, 2012, doi: 10.1007/978-3-642-35289-8_5/COVER/.

[15] J. L. Borade and M. A. Lakshmi, “Multi-class object detection system using hybrid convolutional neural network architecture,” *Multimedia Tools and Applications*, pp. 1–25, Apr. 2022, doi: 10.1007/S11042-022-13007-7/TABLES/9.

[16] A. M. Roy, R. Bose, and J. Bhaduri, “A fast accurate fine-grain object detection model based on YOLOv4 deep neural network,” *Neural Computing and Applications*, vol. 34, no. 5, pp. 3895–3921, 2022, doi: 10.1007/s00521-021-06651-x.