TEKNOFEST

HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

SAĞLIKTA YAPAY ZEKÂ YARIŞMALARI

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

PROJE DETAY RAPORU

TAKIM ADI: PORTAKAL

TAKIM ID: 342157

İÇİNDEKİLER

1. **Proje Mevcut Durum Değerlendirmesi (20 puan) 1**
2. **Özgünlük (30 puan) 2**
3. **Sonuçlar ve İnceleme (30 puan) 4**
4. **Deney ve eğitim aşamalarında kullanılan veri setleri (15 puan) 5**
5. **Referanslar (5 puan) 6**

1. Proje Mevcut Durum Değerlendirmesi (20 puan)

Proje sunuş raporunda U-Net algoritması ve YOLO algoritması kullanılması planlanmaktaydı. U-Net algoritması segmentasyon oluşturmayı sağlamaktadır [1]. Projede bizden istenen object detection olması sebebiyle ve kısa zamanda hızlı ve doğru sonuç alabilmek için tek algoritma çalıştırma üzerine yoğunlaştık. Projeye en uygun algoritmanın YOLO algoritması olduğunu kararlaştırdık. YOLO algoritması çalıştığı zaman görüntülerdeki ve videolardaki nesneleri ve nesnelerin koordinatlarını eş zamanlı, tek seferde, hızlı bir şekilde tespit edebilir. Tespit edilen nesneleri bir kutu içerisinde gösterir. YOLO algoritması yarışmada bizden istenen tüm özellikleri gerçekleştirebilmemize olanak sağlayan bir kütüphane olması ve hızlı olduğu için kullanılmaya devam edilmiştir [2].

Projede Sağlık Bakanlığı tarafından paylaşılan eğitim seti incelendi. Portakal takımı üyelirinden Radyoloji teknikeri üyemiz veri setindeki bazı tutarsızlıklara rastladı. Hastalık tespit edilen görüntüde ilgili olgunun hiç olmaması ve aslında farklı bir olgu olması durumuna göre güncelleme işlemleri yapıldı. Excel formatında verilen datanın düzenlemesinden sonra sonra YOLOV5 PyTourch TXT eğitim dosyasına çevirlmesi düşünüldü [3],[4]. Ayrıca radyolojik görüntüleme dosya formatı olan dicom uzantısı YOLOV5 de eğitim için kullanılamamaktadır. Bu sebeple projede dicom görüntüyü jpg'ye çevirerek model dosyamızı güncelledik. Program asıl amacı radyolojik görüntülerde hastalık olgularının tespit edilemesidir. Bunun için dicom görüntülerin okunması, işlenmesi ve eğitilen model dosyasına göre bir sonuca varması beklenmektedir. Bundan dolayı pydicom kütüphanesi eklendi. Dicom olarak verilen görüntülerin okunması ve işlenmesi sağlandı (Şekil1),[5].

```
import pydicom as dicom
import numpy as np
from PIL import Image
import os

def convert(directory):
    im = dicom.dcmread('./img/'+directory)
    im = im.pixel_array.astype(float)
    rescaled_image = (np.maximum(im,0)/im.max()) * 255 # float pixel
    final_image = np.uint8(rescaled_image) # integers pixels
    final_image = Image.fromarray(final_image)
    return final_image


path="./img"
ct_images = os.listdir(path)
arr_filename = [x for x in ct_images if x.endswith(".dcm")]

for name in arr_filename:
    image = convert(name)
    image.save('./JPG_IMG/'+name+'.jpg')
```

Şekil 1: Dicom dosyaların JPG olarak işlenmesi

Yolov5 algoritmasının farklı versiyonları bulunmaktadır. Verilen eğitim setinde görüntülerin 512x 512 piksel, derinlik 96 bit olması sebebiyle YOLOv5n ile YOLOv5x arasıda bir veriseti algoritması kullanılması neticesine varıldı [6]. Yolov5x algoritması içerisindeki veri setini eğitebilmek için gerekli threshold, olgu sınıflandırılması, batch ve epoch değelerinin yer aldığı; YOLO veri serileştirmeyi sağlayan .yaml kütüphanemiz revize edildi. Böylece eğitim için bize verilen excel dosyasındaki verilerle, yolov5x algoritmasının anlayabileceği şekilde her fotoğraf için ayrı .xml dosyası oluşturuldu. Oluşturulan bu .xml dosyasında x1, y1, x2, y2, noktalarına göre tespit edilen yükseklik ve genişlik değerleri ile nesnenin kutu içerisine alınması sağlandı. Kutu içine alma durumu her fotoğraf için ayrı ayrı yapılarak modelin eğitilmesi sağlandı (Şekil2) [5].

```
    if img_path != prev_img_path:
        img = cv2.imread(img_path)
        img = cv2.resize(img, (int(img.shape[1] * resize_factor), int(img.shape[0] * r
        cv2.imwrite(img_path, img)
        prev_img_path = img_path
    bbox = f"{(int(annotation[1]) * resize_factor) / (int(annotation[6]) * resize_fact
    x1 = (int(annotation[1]) * resize_factor)
    y1 = (int(annotation[2]) * resize_factor)
    x2 = (int(annotation[3]) * resize_factor)
    y2 = (int(annotation[4]) * resize_factor)
    width = (int(annotation[6]) * resize_factor)
    height = (int(annotation[7]) * resize_factor)
    bbox = f"{((x1 + x2) / 2) / width} {((y1 + y2) / 2) / height} {(x2 - x1) / width}
    label_path = img_path.replace("./images/", "labelsval").replace("jpg", "txt")
    with open(label_path, "a") as file:
        file.write("1 " + bbox + "\n")
def check_sku_imgs():
image_paths = os.listdir("./images/")
for img in image_paths:
    if img.find("train") != -1:
        image = cv2.imread("./images/" + img)
        if image.shape[1] > 640:
            print(img)
convert_sku_to_yolo()
```

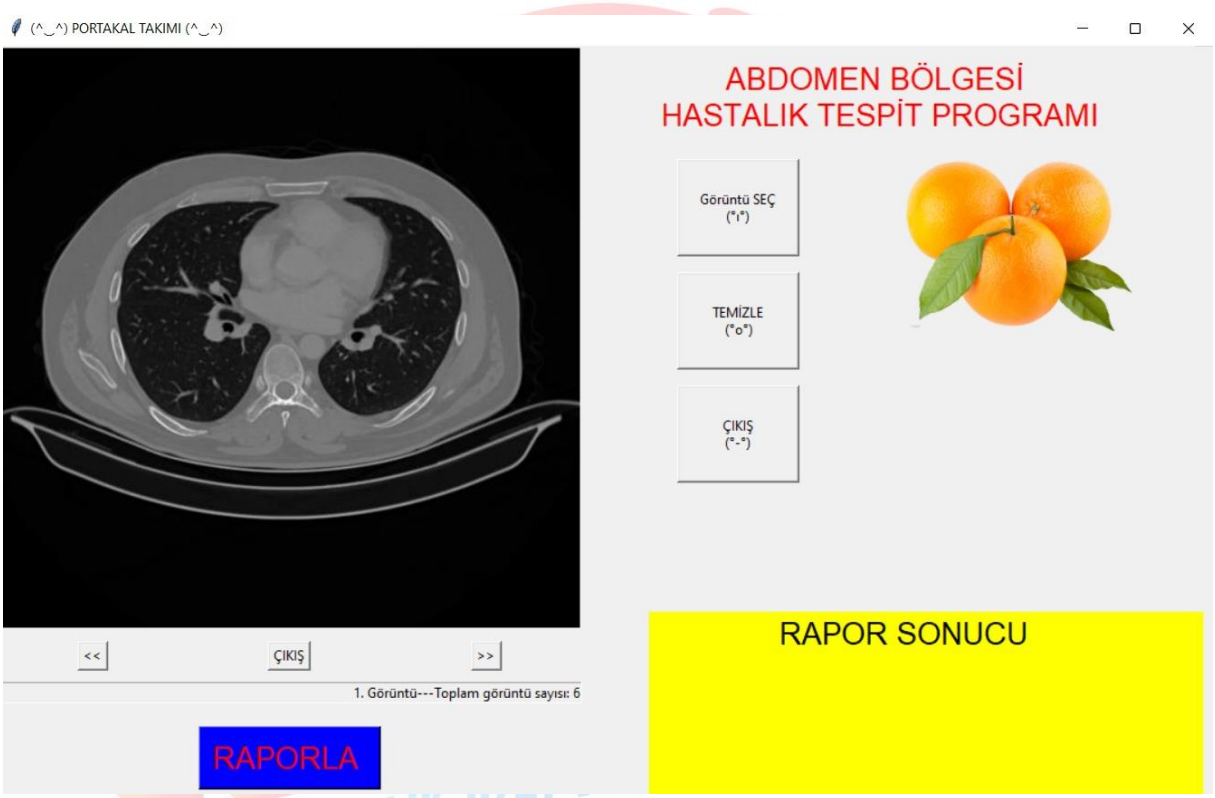
Şekil 2: .yaml dosyasını revize edilmesi

2. Özgünlük (30 puan)

Projede; 0 – Bulgu yok,1 – Apandiks iltihabı (akut apandisit), 2 – Safra kesesi iltihabı (akut kolesistit), 3 – Pankreas iltihabi (akut pankreatit), 4 – Böbrek/üreter taşı, 5 – Bağırsak divertikül iltihabı (akut divertikülit), 6–Karın ana damar balonlaşması/yırtılması (aorta anevrizma/rüptür/diseksiyon) olmak üzere yedi adet durum bizden istenmektedir. Bu Yedi adet duruma ilave olarak 7-Apandikolit, 8- Safra Kesesi taşı, 9-Kalsifiyel divertükül olarak 3 adet ek bulgularda eklenmiştir. Bu bulgular ilk 6 bulguda fizyolojik olarak gelişebilecek olası hastalık seyiri ile ilgi bilgi veren bulgulardır. Program olgu numarası, kesit numarası, tahmin edilen etiket ve etikete ait MBB solÜstNokta (x1,y1), etikete ait MBB sağAltNokta (x2,y2)) şeklinde gösterim sağlanmıştır. Excel de verilen veri setinin ekibimizde bulunan Radyoloji Teknikeri olarak çalışan arkadaşımız ile incelendi bazı olgularda ve kesit alanlarında düzenlemeler yaparak başarım oranını artırdık. Verilerin excelde parse edilmesi ve yolo algoritmasının anlayacağı şekilde revizeedildi. Olası bir durumda veri setimizi başka algoaritmada kulanılabilmesi için convert edebilecek fonksiyonlar yazıldı. Coco Dataset Format, Pascal Voc XML,gibi veri kümeleri için evrensel olacak düzeyde fonksiyonlardır. Yapayzeka analiz sonucunda hem terminal verileri hem de diğer detaylar geçici olarak bir .txt dosyasına aktarılmış. Thkinder kütüphanesi ile yazılan arayüzde ilgili dicom görüntülerin seçilmesi sonrasında arka planda python kodları çalıştırmaktadır. Analiz sonucu ve teknik detay verilerini zaman damgasıyla bir klasöre kaydetmektedir. Arayüz rapor bölümünde de bu veriler görüntü üzerinde ve tablo halinde gösterilmektedir. Tablo verisinde ise olgu numarası başarım oranı, epoch, confidence, IOU değeri yazmaktadır. Görüntü üzerinde ise BoundingBoxda istenilen şekilde gösterilmektedir. Değerlendirme esnasında her bir etiket için eğer olgu numarası-kesit numarası ve etiket değeri doğru ise Birleşimlerin Kesişimi (Intersection of Union- IoU) değerini doğru bir şekilde hesaplamaktadır.

Projede veri seti oluşturmak etiketleme(labeling) yapmak ve veri setinin yolov5x ile çalıştırılması sonrasında arayüz ile raporlama ve görsel üzerinde boundingbox çizilmesi işlemi yapılmış olup bu projede python,php (cli), Nvidia Cuda, Pytourch Tensorflow dil ve kütüphaneleri python arayüz için Tkinter kütüphanesi IDE olarak Pycharm, VsCode, PhpStorm kullanılmıştır. U_Net kullanmamamızın sebebi olarak YOLO kütüphanesinesindeki yolov5 modelinin önceki versiyonlarına göre başarı oranının daha yüksek olmasıdır. Ayrıca U_NET segmentasyon mantığı ile çalışması ile bizden istenilen bir yapıda olmadığı anlaşıldı. YOLOv5

hem başarım oranının yüksek olması ve segmentasyon yerine object detection yapması ile ön plana çıkmıştır. Bize verilen veri setininden dolayı 5x versiyonu kullanılmıştır [7]. Kullandığımız kütüphanelerin ve algoritmaların hızlı ve doğru sonuç verdiği için özgündür. Ayrıca yapmış olduğumuz çalışmaya arayüz ekleyerek programın daha anlaşılır ve kullanılabilir olmasını sağladık. CT yorumlama ve kodlama bilmeyen bireylerinde kolay bir şekilde kullanmasına olanak sağlamaktadır. Bu durum hastanede radyoloji alanında uzman olmayan ve teşhis koymak isteyen sağlık çalışanlarının rahatlıkla programı kullanarak abdomen bölgesinde hastalık tespit edebilmesine yardımcı olmaktadır. Hedefimiz Teknofest alanında masaüstü yazlımı ile göstermekten ziyade mobil veya web arayüzü tasarlayıp insanların erişebileceği bir portal ile ön tespit yaparak ve raporlanmasını sağlamaktır. Böylece toplumun genelinin rahatlıkla kullanabileceği çalışma elde etmektir (Şekil 3).



Şekil 3: Programın arayüz görüntüsü analiz öncesi görüntülemesi

3. Sonuçlar ve İnceleme (30 puan)

Bu çalışmada proje zamanında edindiğimiz MSI gaming bilgisayarmızda hiç bir programlama dili kullanmamamız dolayısıyla yolo kütüphanesini çalıştırmakta zorlandık. Stackoverflow sitesinde ilgili hata kodlarını aratarak bir kaç çözüm önerisi denedik. Sonuç olarak C++ Geliştiri araçlarnın yüklü olmaması nedeniyle Yolo da bulunana Pytourch kütüphanesi sağlıklı çalışmamaktaydı. Metric değerleri ve class_score değerleri düşük ayrıca epoch yükselmesine rağmen verimli değildi. C++ başarım araçlarının yüklenmesiyle kütüphane bileşimlerini kullanmasındaki iç hatalar sona erdi ve başarım metrik değerleri yüklerirken epoch değerinde azalma oldu.

Bilgisayarımızda model eğitimi esnasında ise "Could not load dynamic library 'cudart64_101.dll'; dlerror: cudart64_101.dll not found" hatası almaktadık. Stackoverflow vb siteleri araştırdık Cuda yeniden kurulum işlemi gerçekleştiridik. Fakat Windows değişkenler kısmına Cudanın PATH yolunu eklediğimizde o hata ile ilgili sorunumuzu da çözdük. Pycharm editörü her açtığımızda projede yeniden envoiment oluştuğunu fark ettik. Programın çalışması yavaşlamaktaydı. Python ile ilgili kütüphaneleri her açılışta yeniden yükleme durumu sözkonusu olmaktaydı. Sonuç olarak programdaki hatalarımızdan dolayı projede ilerleyemedik program çok yavaş çalışmaya başladı, projede ilerleme aşamamız durma noktasına gelmişti. Her gün farklı bir hata ile karşılaşıyorduk. Sonra Pycharm'da Python kütüphanelerini global path olarak yükleme yaptık. Yüklememiz sonucunda artık hatalarımızı sistemsel değil, proje kod olarak almaya ve düzenlemeye başladık.

Geliştirdiğimiz sistemde donanımızın sağladığı en büyük değeri batch 20 değerini kullandık. Öncelikle 200 görüntü alınmış ve bu 200 görüntünün 150si eğitim 50 adet test verisi olarak kullanılmıştır. Varsayılan Epoch değeri 300 fakat class score daha da yükselmek için Epoch değeri 600 değeri 1200 şeklinde belirlenmiştir. 60ms içerisinde hızlı bir sonuca ulaşılmıştır. Daha sonra eğitim ve test verisinde bulunmayan farklı 10 adet görüntüde tespit durumu değerlendirilmiştir. Bu aşamada classfication hatası sonucu farklı sınıflandırma ile karşılaşılmıştır. Bu sorunu çözmek için belirlene test verisi artırılarak doruluk payı daha yükseltilmiştir.

Aynı işlem 1000 görüntü üzerinde 750 eğitim ve 250'si test olarak belirlenen bir dataset verisinde işlem yapılmıştır. İlk başta verilen batch değerinin yüksek olması başarı oranı (confidence) düşük hata oranı çok yüksek olmuştur. Daha sonrasında batch değerini yükselmemiş epoch değerinini nin de yüksek olması programın çalışma verim ve performanasını gözle görülür düşürmüş çözüm olarak batch ve epoch(model eğitim tekrarı) yükselttik daha sonra aynı şekilde performans sorunu yaşayınca en uygun ve en uc değerlerde deneme yanılma yaparak toplam verinin batch değerinin bölünmesi ile epoch arasında bir orantı olduğunu tespit ettik. Bu sayede başarım oranımız arttı ayrıca model dosyamızda da gözle görülür bir boyut küçülmesi görüldü. Başarım analizimizde İntel CVAT yazılımını bize sunduğu diğer önermeleri de inceledik. Hepsini uygulayamasakta bazı önerileri ve resim verilerinde threshold değerinde oynamalar yaparak grayskale düzenlemeler yaptık.

Başka bir hata olarak bazı radyolojik görüntülerinde kontrast madddenin taş veya iltihabi bir durum olarak algılanması söz konusuydu. Bunuda radyolojik görüntülerde bululnnan Housfield skalası denilen hematom(kan) doku , su ve hava değerlerini ilgili de ekstra labelleme işlemi yaparak başarım oranımızı daha da artırdık. Wandb.ai aracını kullanarak best model, fit model şeklinde grafik analiz ve önermelerini inceledik. Netice olarak batch değeri yüksek olması sorun oluşturmuş. Bu sorunları çözebilmek için batch değerini düşürmek epoch değerini yükselmek donanımızın izin verdiği ölçüde orantılı bir değişim sağladık. İntel CVAT,

wandb.ai araçlarındaki önermeler ile model dosyamızda düzenlemeler yaptık. Yapılan testler ve araştırmalar sonucunda epoch 600 Batch 20 değerin weight değeri ise 0.6 ile 0.8 arasında tespit ettik. Değer aralıkları ve gerekli düzenlemeler yapıldıktan sonra daha büyük bir veri e eğitim seti oluşturularak modelin eğitilmesi sağlanmıştır.

4. Deney ve eğitim aşamalarında kullanılan veri setleri (15 puan)

Bu çalışmada deney ve eğitim aşamalarında kullanılan veri setleri Sağlık bakanlığı verilerinden alınmıştır. Sağlık Bakanlığı ile takımımız arasında imzalana gizlilik sözleşmesiyle tarafımıza gönderilen bilgiler ile verilere erişim sağlanmıştır.

TEKNOFEST
HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

5. Referanslar (5 puan)

[1] E. DANDIL, M. S. YILDIRIM, A. O. Selvi, and U. Süleyman, "Bilgisayarlı tomografi taramaları üzerinde maskeli bölgesel-evrişimsel sinir ağları ile karaciğerin otomatik bölütlenmesi," *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, vol. 37, no. 1, pp. 29-46, 2022.

[2] R. Huang, J. Pedoeem, and C. Chen, "YOLO-LITE: a real-time object detection algorithm optimized for non-GPU computers," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018: IEEE, pp. 2503-2510.

[3] Y. Su, Q. Liu, W. Xie, and P. Hu, "YOLO-LOGO: A transformer-based YOLO segmentation model for breast mass detection and segmentation in digital mammograms," *Computer Methods and Programs in Biomedicine*, p. 106903, 2022.

[4] H.-h. Cho *et al.*, "Radiomics-guided deep neural networks stratify lung adenocarcinoma prognosis from CT scans," *Communications biology*, vol. 4, no. 1, pp. 1-12, 2021.

[5] M. Mushtaq, M. U. Akram, N. S. Alghamdi, J. Fatima, and R. F. Masood, "Localization and Edge-Based Segmentation of Lumbar Spine Vertebrae to Identify the Deformities Using Deep Learning Models," *Sensors*, vol. 22, no. 4, p. 1547, 2022.

[6] A. Hossain, M. T. Islam, and A. F. Almutairi, "A deep learning model to classify and detect brain abnormalities in portable microwave based imaging system," *Scientific Reports*, vol. 12, no. 1, pp. 1-27, 2022.

[7] M. E. Salman, G. Ç. Çakar, J. Azimjonov, M. Kösem, and İ. H. Cedimoğlu, "Automated prostate cancer grading and diagnosis system using deep learning-based Yolo object detection algorithm," *Expert Systems with Applications*, vol. 201, p. 117148, 2022.

TEKNOFEST
HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ