



7. Geleceğin Mühendisleri Uluslararası Öğrenci Sempozyumu
7th Engineers of Future International Student Symposium
22-23 Haziran/June 2023, Zonguldak, Türkiye

<http://www.efis.beun.edu.tr>

İNSANSIZ HAVA ARAÇLARINDA NESNE TESPİTİ VE TAKİBİ

Abdullah Hayri HASAN^{*1}, Emanuela IBRA², Esmanur CİVİL³, Murtaza CİCİOĞLU⁴

^{1,2,3,4} Bilgisayar Mühendisliği Bölümü, Bursa Uludağ Üniversitesi, 16059 Bursa, Türkiye

ÖZET

Günlük hayatta halihazırda birçok alanda kullanılan yapay zeka ulaşım sektöründe de aktif olarak kullanılmaya başlanmıştır. Günümüz ve geleceğin teknolojisi olan yapay zeka alanında yapılan geliştirmeleri yakından takip etmek bu alanı geliştiren tarafta bulunmak ülkemiz için son derece önemlidir. Bu çalışmada ulaşımda yaşanan ve ileride yaşanabilecek sorumlulara yapay zeka tabanlı çözümler geliştirilmesi amaçlanmaktadır. Bu amaç doğrultusunda insansız hava araçlarında kamера yardımıyla nesne tespiti ve takibi yapabilecek derin öğrenme tabanlı bir model geliştirilmiştir. İnsansız hava araçları için nesne tespit algoritmalarından YoloV5 kullanılmıştır. Veri seti için hem hazır veri setleri kullanılmış hem de drone yardımıyla yeni veri seti oluşturulmuştur. Bu sayede daha gerçekçi bir model geliştirilemeye çalışılmıştır. Çalışmamızda dokuz sınıfı ait farklı nesneler tespit edilmektedir. OpenCV kütüphanesi yardımıyla verilerde düzeltme, çoğaltma gibi işlemler yapılmıştır. Bu sayede sentetik veriler de kullanılarak veri seti genişletilmiş ve optimize hale getirilmiştir. Bu yöntemler ile birlikte yapay zeka modelinde yapılan analizler sonucunda mAP değeri 0.9'lara ulaşırken sınıflara ait loss değeri 0.014 değerlerinde olduğu görülmüştür.

Kelimeler: İnsansız hava araçları, Nesne tespiti, YoloV5, OpenCV

OBJECT DETECTION AND TRACKING IN UNMANNED AERIAL VEHICLES

ABSTRACT

Artificial intelligence, which is already widely used in various aspects of daily life, has also begun to be actively employed in the transportation sector. Keeping a close eye on advancements in artificial intelligence, which is the technology of today and the future, is of utmost importance for our country to be at the forefront of this field. This study aims to develop AI-based solutions for the problems currently encountered in transportation as well as potential issues that may arise in the future. To this end, a deep learning-based model has been developed to enable object detection and tracking using cameras on unmanned aerial vehicles (UAVs). YoloV5, one of the object detection algorithms, has been utilized for UAVs. Both pre-existing datasets and a new dataset created with the assistance of drones have been employed as the training data, aiming to develop a more realistic model. In our study, various objects belonging to nine different classes are detected. Data augmentation and other modifications have been performed on the data using the OpenCV library. In this way, the data set was expanded and optimized by using synthetic data. As a result of the analyzes made in the artificial intelligence model with these methods, it was seen that the mAP value reached 0.9, while the loss value of the classes was 0.014.

Keywords: Unmanned aerial vehicle, Object detection, YoloV5, OpenCV

1. GİRİŞ (INTRODUCTION)

Teknolojik gelişmeler ve endüstriyel ilerlemelerle birlikte insansız hava araçları birçok alanda önemli rol oynamaktadır. Günümüzde özellikle savaş sistemleri ve sivil uygulamalarda sıkılıkla kullanılmaktadır. Bu bağlamda insansız hava araçlarına kameralar yardımı ile nesne tespiti ve takibi yeteneklerinin kazandırılması birçok uygulama alanında kolaylık sağlamlmaktadır. Özellikle yabancı nesne tespiti, silah tespiti, yanım tespiti gibi alanlarda sıkılıkla kullanılmaktadır. Yapılan çalışmada trafikte yaşanan bazı problemlere çözüm olması amacıyla araçların türlerine göre sınıflandırılması ve yayaların tespit edilmesi üzerinde durulmuştur [1]. Nesne tespiti ve takibi bilgisayarlı görü ve derin öğrenme gibi alanların kesişiminde yer alan çok disiplinli bir araştırma alanıdır. Çalışmanın ana hedefi, insansız hava araçlarının nesne tespiti ve takibi yeteneklerini geliştirmek için yenilikçi bir yaklaşım sunmaktadır. Çalışma aynı zamanda yapay zeka teknolojisinin ulaşım sektöründe nasıl kullanılabileceğini gösteren bir örnek teşkil etmektedir. Bunun doğrultusunda nesne tespiti konusunda derin öğrenme yöntemlerinden biri olan Yolov5 algoritması kullanılmıştır. Yolov5, nesne tespiti ve takibi konusunda son derece etkili bir derin öğrenme algoritmasıdır. Yolov5 algoritması gelişmiş bir konvولüsyonel sinir ağının yapısına dayanır ve görüntülerdeki nesneleri tespit etmek ve sınıflandırmak için kapsamlı bir öğrenme sürecinden geçer. Bu süreçte, binlerce etiketlenmiş görüntü algoritma üzerinde eğitilir ve ağırlıkları optimize edilir. YOLOv5, esneklik ve hız açısından RCNN, FasterRCNN vb. çok katmanlı algoritmalarla göre avantaj sağlayarak insansız hava araçlarının nesneleri gerçek zamanlı olarak tespit etme kabiliyetini artıracaktır. OpenCV ise, görüntü işleme ve analiz için kullanılan güçlü bir kütüphanedir. Bu çalışmada, OpenCV'nin sağladığı işlevlerle, nesne tespiti ve takibi yetenekleri optimize edilmiştir. Özellikle ön işleme sürecinde görüntülerin gürültülerinden ayırtılmasında, netleştirme, maskeleme gibi işlemlerde OpenCV kütüphanesi sıkılıkla kullanılmıştır. OpenCV, görüntü işleme algoritmalarını hızlı bir şekilde uygulama, görüntü işleme işlemlerini optimize etme ve görüntü verilerini işlemek için kullanılabilir araçlar sunar. YOLOv5 ve OpenCV gibi teknikler kullanılarak, insansız hava araçlarının çevrelerindeki nesneleri tespit etme ve gerçek zamanlı olarak takip etme yeteneklerinin önemli ölçüde geliştirilmesi amaçlanmıştır. Bu tekniklerin kullanımı, insansız hava araçlarının otonomluğunu artırırken, görevlerini daha etkili bir şekilde yerine getirmesine olanak sağlayacaktır. Bununla birlikte küçük nesnelerin tespit edilmesi konusunda çalışmaya SAHI kütüphanesi entegre edilerek model başarısı artırılmıştır. Çalışma süresince yapılan analizler sonucunda model başarı orası 0.9'lara ulaşırken sınıflara ait kayıp değeri 0.014 değerlerinde olduğu görülmüştür.

2. VERİ VE YÖNTEM (DATA AND METHOD)

Nesne tespitinde yapay zekâ tabanlı çözüm oluşturduğu için ilk etapta bir yapay zeka modeli oluşturulması elzemdir. Yapay zekâ modelini oluşturmadan önce yapılması gereken birtakım aşamalar mevcuttur. Bu aşamalar Şekil 1'de belirtilmiştir. Yapay zekâ modeli oluştururken olmaz noktalardan biri verilerin kullanılabilir nitelikte olmasıdır. Bu amaçla veri temizleme, bulanıklık, gürültü giderme, veri formatı dönüştürme gibi ön işleme (preprocessing) adımları mevcuttur. Veri ön işleme kısaca bir veri kümесini düzenlemeye ve kullanılabilir hale getirme sürecidir. Bu süreç genellikle bir veri kümescini analiz etmek, anlamak ve kullanılabilir hale getirmek için kullanılır. OpenCV, görüntülerdeki nesneleri tanıma, görüntüleri birleştirme ve bölme, görüntüleri filtreleme gibi birçok işlemi gerçekleştirmeye yardımcı olur. Veri ön işleme aşamalarında aktif olarak openCV kütüphanesi kullanılmaktadır. Verilerdeki atık kısımlar giderilerek yapay zeka algoritmasına saf veri girişi sağlanmıştır.



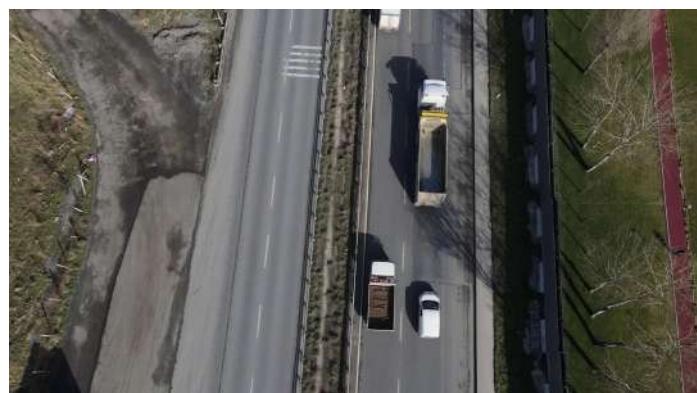
Şekil 1. Model oluşturma aşamaları

2.1 Veri Ön İşleme (Data Preprocessing)

OpenCV ile görüntüleri normalleştirme ve ölçeklendirme yöntemleri kullanılarak verilerin dağılımı düzgün bir şekilde değiştirebilmektedir. Örneğin, OpenCV ile görüntüleri gri tonuna dönüştürerek renklerden arındırabilir veya görüntülerin boyutu değiştirilerek algoritma uygun olarak ölçeklendirilebilmektedir [2]. Görüntüleri normalleştirme için OpenCV fonksiyonlarından biri olan normalize() fonksiyonu kullanıldı. Bu fonksiyon, görüntü verilerini alarak görüntüleri normalleştirme işlemini gerçekleştirir ve normalleştirilmiş görüntü verilerini döndürür. Görüntüleri ölçeklendirme için OpenCV fonksiyonlarından biri olan resize() fonksiyonu kullanıldı. Bu fonksiyon, görüntü verilerini ve istenilen boyutları alarak görüntüleri ölçeklendirme işlemini gerçekleştirir ve ölçeklendirilmiş görüntü verilerini döndürür. OpenCV ile görüntüleri kümemele yöntemleri kullanılarak veriler benzer özelliklere sahip gruplara ayrılmaktadır. Örneğin, OpenCV ile görüntüler k-ortalama yöntemiyle sınıflara ayırlabilir ve her bir sınıf ayrı ayrı incelenilmektedir. Görüntüleri kümemeleme için OpenCV fonksiyonlarından biri olan kmeans() fonksiyonu kullanıldı. Bu fonksiyon, görüntü verilerini ve istenilen sınıf sayısını alarak görüntüleri kümemeleme işlemini gerçekleştirir ve her bir görüntünün hangi kümeye ait olduğunu gösteren bir maske görüntüsü döndürür.

2.2 Veri artırma (Data Augmentation)

Yapay zekâ modeli oluştururken olmazsa olmaz bir diğer nokta veri miktarı ve çeşididir. Bir modelin gerçek hayat başarımının yüksek olabilmesi, farklı senaryolarda başarıyla çalışabilmesi için çok çeşitli ve çok sayıda veriye ihtiyaç duyulmaktadır. Aksi halde aşırı öğrenme(overfitting) problemi ile karşı karşıya kalınabilmektedir [3]. Aşırı öğrenme problemiyle birlikte modelin eğitim sürecindeki başarımı çok yüksek olmasına karşı gerçek dünya verilerinde düşük performans sergilemesi durumuyla karş karşıya kalınmaktadır. Veri seti olarak farklı hava şartları, gece gündüz vb. farklı senaryolar da içermesi ve proje isterlerine uygun olması sebebi ile Visdrone veri seti tercih edilmiştir ve sonradan stok videolardan yararlanmıştır. Ayrıca dron kullanarak farklı alanlarda veri seti toplanmıştır. Bunların dışında veri çeşidini artırmak için sentetik veri üretimi yoluna gidilmiştir. Bu amaçla veri artırma (data augmentation) yapılmaktadır. Veri artırma konusunu kısaca açıklamak gerekirse; modelin başarımını artırmak, overfitting problemine (ezberleme) engel olmak amaçlarıyla daha önceden var olmayan sentetik verilerin üretilmesi işlemidir. Bu amaçla girdi olarak verilen fotoğraflar; belirli açılarla döndürülme, yakınlaşma, uzaklaşılma vb. işlemlere tabi tutularak var olan fotoğrafın farklı versiyonları ortaya çıkarılmaktadır. Aşağıdaki görüntülerde orijinal görüntü (Şekil 2) ve orijinal görüntünden üretilen görüntü çıktıları bulunmaktadır (Şekil 3). Verilerin kullanıma hazır hale getirilmesi ile birlikte veri etiketleme (data annotation) aşamasına gelinmektedir.



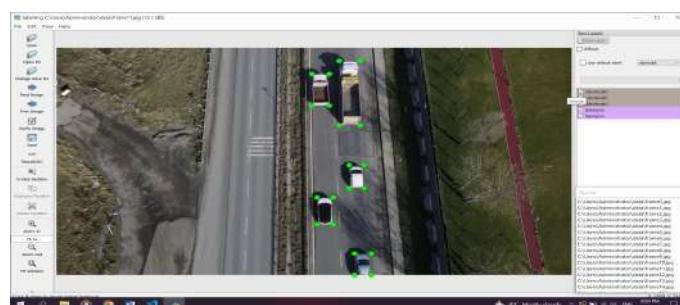
Şekil 2. Orijinal Görüntü



Şekil 3. Veri artırma uygulandıktan sonra çıktı görüntülerı

2.3 Veri etiketleme (Data Annotation)

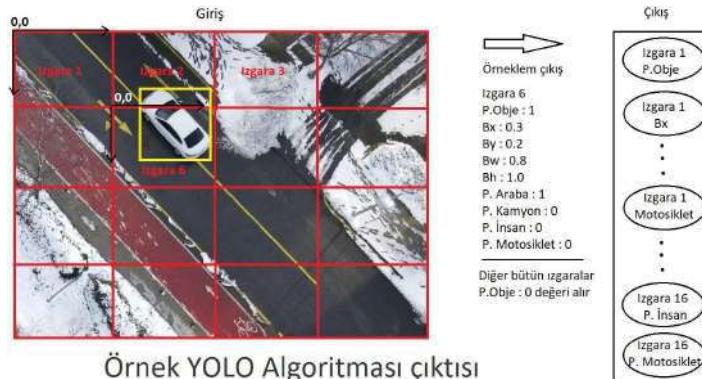
Bir resim üzerinde aranan ya da istenilen objenin koordinatlarını elde etmek ya da gösterme aşamasına veri etiketleme denir [4]. Bu konuya önemsemekte fayda vardır çünkü bir veri seti ne kadar düzgün etiketlenmiş ise daha sonra model başarısının artmasına yardımcı olur. Veri etiketleme sırasında tespit edecek objelerin düzgün bir şekilde sınıflandırması önemlidir. Şekil 4'de gösterilmiş python dili ile yazılan açık kaynak kodlu program olan labelImg programı yardımıyla görüntüler üzerinde etiketleme yapılmaktadır. Önceden yapılmış bir etiketleme üzerinde düzeltmeler yapılabilmektedir. Bir resim üzerinde farklı objeler etiketlenebilir. Daha verimli ve başarımı yüksek sonuçların çıkarılması için yeterince etiketleme yapılması gerekmektedir. Bu işlem çok fazla zaman ve işçilik aldığı için otomatik etiketleme algoritması yapılmıştır. Orta seviyede çalışan bir model eğitilmiştir. Devamında elde edilen model kullanılarak veri setinde bulunan her görüntüdeki insan, araba, kamyon gibi objeler tespit edilerek gerekli olan koordinatlar .txt uzantılı dosyaya yazdırılarak etiketleme işlemi tamamlanmıştır. Daha sonra labelImg uygulaması üzerinden görüntüler üzerinde gereken düzeltmeler sağlanmıştır. Böylelikle veri etiketleme işlemi çok daha az zaman içerisinde yapılmıştır. Verileri nesne tespiti algoritması tarafından kullanılabilir hale getirdikten sonra sıradaki aşama Yolov5 algoritmasının uygulamaya konulmasıdır.



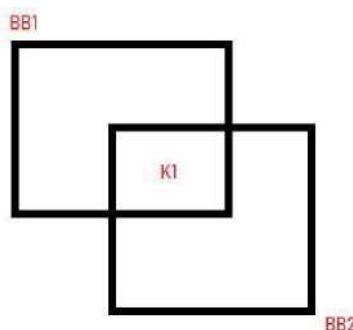
Şekil 4. Etiketleme yapım aşaması

2.4 Yolov5 kullanımı (Yolov5 usage)

Projede modeli eğitmek üzere seçilen algoritmadır ve 200 epoch 64 batch değeri tercih edilmiştir. Temelde diğer nesne tanıma algoritmalarına göre daha hızlı çalışması ile ünlenmiştir. Yolov5 algoritması resmi tek seferde sinir ağına gönderir ve gereken tahminlemeleri yapar. Yolo v5 algoritmasında Görüntü nxn lik ızgaralara bölünür ve o alanlarda nesne var mı varsa sınıfı nedir, aranan nesnenin orta noktası o alanın içinde midir bunları araştırır ve her alana ait bir tahmin vektörü oluşturulmaktadır [5]. Tahmin vektörü ve uygulaması çalışma görüntüsü üzerinde şekil 5'te gösterilmiştir. Nesnenin merkezi hangi ızgaraya denk geliyorsa o nesnenin yükseklik ve genişlik bilgilerine göre sınırlandırıcı kutucuk (bounding box) çizmektedir. Sınırlandırıcı kutucuk lar 5 adet bilgi içerir. x koordinatı,y koordinatı, genişlik, yükseklik, güven skoru. Güven skoru şu anlama gelmektedir: Her sınırlandırıcı kutucuk için bir olasılık hesabı yapılmaktadır. Bu hesaplama yüzde kaç olasılıkla nesnenin tahmin edildiğini bizlere söylemektedir. Fakat Birden fazla merkez tahminlemesi yapılrsa birçok sınırlandırıcı kutucuk çizilebilmektedir. Bunun önüne geçebilmek için non max suppression algoritması kullanılmaktadır. Algoritma şu şekilde çalışmaktadır. Çizilen tüm sınırlandırıcı kutucukların güven skoru sıralanır. Belirli bir eşik değeri belirlenir. Bu eşik değeri altında güven skoruna sahip kutucuklar görüntünden kaldırılır. Kalan sınırlandırıcı kutucuk lar için IOU hesaplaması yapılır. IOU iki kutucuğun birbiri ile örtüşme oranını göstermektedir (denklem 1, denklem 2) IOU için de bir eşik değeri belirlenmektedir. Eğer eşik değeri üzerinde ise güven skoru düşük olan kutucuk silinir. Buradaki amaç birbirine çok yakın mesafede fazladan çizilen sınırlandırıcı kutucukları kaldırmaktır. IOU hesaplaması şekil 6'da gibidir. Böylelikle fazlalık sınırlandırıcı kutucuklar da görüntünden kaldırılmış olmaktadır [6].



Şekil 5. Örnek YOLO algoritmasının çıktısı



Şekil 6. IOU hesaplanması

2.5 Malzemeler ve Hazırlama Teknikleri (Materials and Preparation Techniques)

Elde edilen yazılım aynı ağa bağlı olunan bir bilgisayar ve IP webcam uygulaması yüklü herhangi bir kamera ile çalıştırılabilir. Yapay zekâ modelinin oluşturulup eğitilmesi aşamasında Ubuntu 20.04 işletim sistemi tercih edilmiştir. Deep learning alanında donanım kapasitelerinin artması, paralel işlem gücü, yüksek performans sağlama sebepleri ile birlikte GPU'lar sık sık kullanılmaktadır. Proje ekibi tarafından bilgisayarda bulunan GPU üzerinde eğitim yapılmaktadır. GPU üzerinde işlem yapmayı kolaylaştıran ve derin öğrenme modellerinde

verimli bir şekilde kullanılmakta olan CUDA 11.5 versiyonu ve cuDNN 8.3.3 versiyonları kullanılmaktadır. Veri seti toplamak için dron kullanılmaktadır.

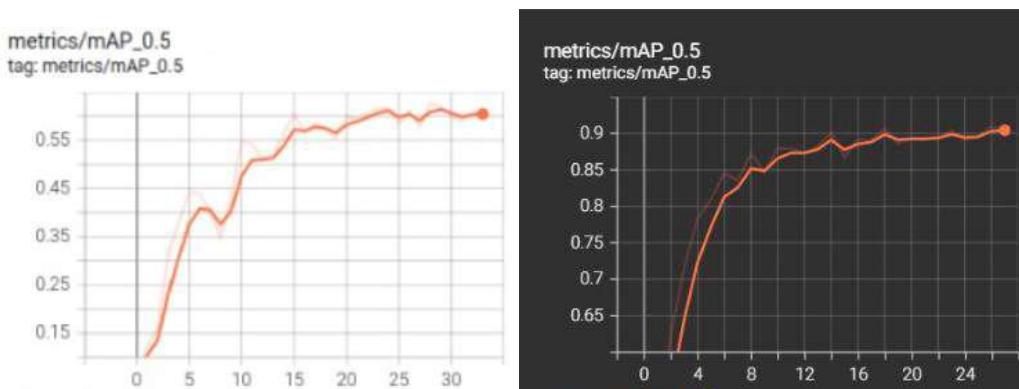
2.5.1 Malzemelerin karakterizasyonu (*Characterization of materials*)

Sistem Özellikleri:

1 Tb SSD
GTX 1650 ekran kartı
16 gb RAM
i7 9. nesil işlemci
6 çekirdekli CPU

3. BULGULAR VE TARTIŞMA (RESULTS AND DISCUSSIONS)

YOLOv5 algoritmasında "mAP" (ortalama hassasiyet) değeri, modelin nesne tespiti performansını ölçmek için kullanılan bir metriktir. "mAP", modelin tespit ettiği nesnelerin doğruluk ve kesinlik derecesini temsil eder. Yüksek bir mAP değeri, modelin gerçek dünyada farklı nesneleri başarılı bir şekilde tespit etme ve sınıflandırma yeteneğine sahip olduğunu gösterir [7]. Veri setinde yapılan düzeltmeler sayede mAP değerinin artışı 0.6'dan 0.9'a kadar yukarıdaki şekil 7'de görülmektedir. Veri setindeki bütün sınıfların etiket sayısını senkronize edilerek aynı zamanda resimlerin boyutu kare şecline çevrilerek daha yüksek mAP değeri elde edilmiştir. Bahsedilmiş mAP değerine karşı obj_loss değerinin düşürülmESİ gerekmektedir. Bu değer, tahmin edilen nesnenin gerçek nesneye ne kadar yakın olduğunu belirler. Değişikler yapıldıktan sonra obj_loss değeri Şekil 8'de görüldüğü gibi düşmektedir.



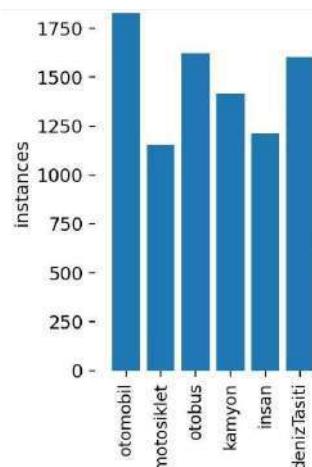
Şekil 7. mAP değeri



Şekil 8. Obj_loss değeri

4. SONUÇLAR (CONCLUSIONS)

Geliştirilen çalışma ile birlikte özellikle insansız hava araçlarında uzaktan alınan görüntüler ile nesne tespiti nasıl en iyilenebilir, sistem en rahat hangi algoritmalar ve yazılımlarla çalışabilir gibi sorulara cevap aranmıştır. Yapay zeka alanında sıkılıkla kullanılan, kaynak tüketimi fazla olmayan, yüksek fps ve yüksek doğruluk oranları sağlayan yolov5 algoritması kullanılmıştır. Geliştirilen model ile birlikte aşağıdaki dağılıma sahip veriler tespit edilebilmektedir. Veri setinde araçlar türlerine göre sınıflandırılmaktadır, aynı zamanda olabildiğince yakın sayıarda veri toplanmıştır(Şekil 9). İlk aşamada görüntülerin boyutlarının aynı ve kare şeklinde olmasına dikkat edilmemiştir. Bu problemin üstesinden gelebilmek amacı ile tüm fotoğrafların kalitesinin gözle görülür şekilde bozulmadan aynı boyuta getirilmesi amacıyla python üzerinden script yazılmıştır. Veri ön işleme adımlarında geliştirilen kısa kod parçacıkları ile birlikte internet üzerinde bulunan veri setleri direkt kullanılabilir hale getirilmiştir. Sistemin, günümüzde kullanılan birçok sisteme göre avantajları olarak genel anlamda veri ön işleme adımları örnek verilebilir. Geliştirilen kod parçacıkları ile birlikte yapılacak birçok işlem insan müdahalesi olmadan, iş yükünü hafifletecek şekilde kullanıma sunulmuştur. İnternet üzerinde kullanılabilen veri setleri direkt olarak indirilerek dosyalarda yapılan işlemler ile kullanıma hazır hale getirilmektedir. Veri setinin proje başlangıç demosuna göre çeşitlendirilmesi, sayıların artırılması ve parametrelerin düzenlenmesi ile birlikte bekleniği üzere mAP değeri artmaktadır. Demo gösteriminde mAP değeri maksimum 0.6 civarında olmakla beraber uygulanan işlemler sonrasında mAP değeri 0.9 seviyelerine ulaşmıştır. Yaklaşık %30 luk bir artış sağlanmıştır.



Şekil 9. Veri Setinde bulunan sınıflar

KAYNAKLAR (REFERENCES)

- Wang, T., Zhao, F., Wan, J., Zhu, Y. (2010). A Novel Hardware Architecture for Rapid Object Detection Based on Adaboost Algorithm. In: , et al. Advances in Visual Computing. ISVC 2010. Lecture Notes in Computer Science, vol 6455. Springer, Berlin, Heidelberg.
- Quanfu Fan, Lisa Brown, and John Smith. 2016. A closer look at Faster R-CNN for vehicle detection. In 2016 IEEE Intelligent Vehicles Symposium (IV). IEEE Press, 124–129.
- Shorten, C., Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. J Big Data 6, 60 (2019).
- T. Zhao, S. Liao and Z. Lei, "Semi-automatic Data Annotation Tool for Person Re-identification Across Multi Cameras," 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 4672-4677, doi: 10.1109/BigData.2018.8622366.

A. M. A. Abdulghani and G. G. Menekşe Dalveren , "Moving Object Detection in Video with Algorithms YOLO and Faster R-CNN in Different Conditions", Avrupa Bilim ve Teknoloji Dergisi, no. 33, pp. 40-54, Jan. 2022,

J. Hosang, R. Benenson and B. Schiele, "Learning Non-maximum Suppression," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 6469-6477, doi: 10.1109/CVPR.2017.685.

Zhao, Z., Yang, X., Zhou, Y. et al. Real-time detection of particleboard surface defects based on improved YOLOv5 target detection. Sci Rep 11, 21777 (2021).

TEŞEKKÜR (THANKS)

Bu çalışma TÜBİTAK 2209-A programı doğrultusunda desteklenmeye hak kazanmıştır. Süreçte maddi manevi desteğini esirgemeyen TÜBİTAK kurumuna teşekkür ederiz.