

# GÜNEŞ PANELİ ANALİZİ

Not: Drone üzerine entegre kamera ile güneş panelleri analiz sistemi

Alper Ceylan  
Yazılım Mühendisliği  
İstanbul Topkapı Üniversitesi  
İstanbul, TR  
alperceylan@stu.topkapi.edu.tr

Emirhan Sertel  
Yazılım Mühendisliği  
İstanbul Topkapı Üniversitesi  
İstanbul, TR  
emirhansertel@stu.topkapi.edu.tr

Ahmet Buğra Kaplan  
Yazılım Mühendisliği  
İstanbul Topkapı Üniversitesi  
İstanbul, TR  
ahmetbugrakaplan@stu.topkapi.edu.tr

**Abstract—** Bu projede, güneş panellerinde oluşan arızaların tespit edilmesi, sınıflandırılması amacıyla derin öğrenme tabanlı bir sistem geliştirilmiştir. YOLO modeli kullanılarak, temiz, tozlu, kuş pislikli, karla kaplı, elektriksel hasarlı ve fiziksel hasarlı olmak üzere çeşitli kategorilere ayrılmıştır. Görüntüler farklı klasörlerde etiketlenmiştir. Ardından ön işleme ve veri artırma teknikleri kullanılarak yaklaşık %95 civarında bir doğruluk oranı almayı başarmıştır. Bu sonuç güneş panellerinde ki arıza tespiti için etkili çözümler olabileceğinin göstergesidir. Panellerin tespiti için OpenCV ve fotoğraf çekilmesi ve aktarım ESP32-CAM kullanılmıştır.

**Keywords—** Yapay sinir ağı, YOLO, Güneş paneli, enerji, OPenCv, ESP32, Sınıflandırma

## I. Giriş

Her geçen gün enerji ihtiyacımız artıyor. Dolayısıyla her geçen gün enerji kaynaklarına olan ilgi artıyor. Sürdürebilir ve çevre dostu olduğu için güneş panelleri yaygın olarak tercih edilmektedir. Ancak güneş panelleri sisteminde meydana gelen arızaların insan gözüyle tespit edilmesi çok zordur. Arıza meydana gelen panel elektrik üretemez ve bu nedenle enerji üretim verimliliği ciddi miktarda azalır. Bu yüzden düzenli kontrol edilmesi ve meydana gelen arızaların en kısa sürede tespit edilip düzeltilmesi gerekir. Bu bağlamda, görüntü işleme ve derin öğrenme gibi yapay zeka teknikleri arıza tespitlerinde önemli bir çözüm sunmaktadır.

Bu projede, sınıflandırma işlemi için farklı klasörlerde toplanan görüntü setleri vasıtasıyla etiklendirilme yapılmış, görüntü üzerinde çeşitli ön işleme teknikleri kullanılmış ve bu vasıtada model eğitime uygun hale getirilmiştir. Bu çalışmanın temel amacı, güneş paneli üzerinde gelecek 5 farklı arıza türünün en kısa sürede tespit edilip kullanıcıya bildirilmesi ve bu sayede enerji verimliliğinden kaybı minimize etmesidir.

## II. Veri Seti ve Veri Toplama

### II.I. Veri Seti Tanıtımı

Veri seti bir sistemin temel yapı taşıdır ve sistemin doğru çalışmasını sağlamak için kaliteli veri setlerine ihtiyaç vardır. Güneş paneli sınıflandırma projesi için kullanılan veri seti 6 farklı (temiz, kirli, kuş dışkısı, elektriksel hasarlı, fiziksel hasarlı, karlı) etiketleme içerir. İlk aşamada açık kaynaklı

platform olan kaggle.com üzerinden Solar “Panel Images - Clean and Faulty” [1] ve solar\_augmented\_dataset [2] başlıklı veri setleri indirilmiştir. Ardından Topkapı Üniversitesi Altunizade Yerleşkesi bahçesinde bulunan farklı sınıflandırmalara elverişli güneş panelleri üzerinde, geliştirdiğimiz drone uçuruldu ve esp32-cam vasıtasıyla fotoğraflarını çekip veri setlerine ekleme yapıldı. Bu iki veri seti detaylı incelenip model üzerinden test yapıldı. Veri setlerinden bir tanesi diğerinin altı katı veri içermesine rağmen kötü sonuç vermesi üzerine aynı sınıfları ve benzer görselleri içeren daha küçük bir veri seti kullanıldı ve çekilen fotoğraflar bu veri setine eklendi. Bu işlemin sonucunda veri setinin içeriğinde ki veri sayısı 1000 adet görsel veriye yaklaştı. Resim 1’de de bu görsellerin bir örneğini görebilirsiniz.



Resim 1 (Temiz Güneş Panelleri)

### II.II. Veri Ön İşleme Adımları

Veri ön işleme, modelin başarı oranını etkileyen en önemli adımlardan biridir. Bu projede, YOLOv8 mimarisine uygun şekilde veri ön işleme adımları titizlikle uygulanmıştır. İlk aşamada, veri setindeki tüm görseller 224x224 piksele yeniden boyutlandırılmıştır. Bu adım, YOLOv8 sınıflandırma modelinin mimari gereksinimlerini karşılamak amacıyla gerçekleştirilmiştir.

Görsel veriler, klasör isimlerine göre otomatik olarak etiketlenmiştir. Her bir alt klasör, ilgili sınıfı temsil etmektedir. Eğitim verisi ve doğrulama verisi, sistem tarafından otomatik olarak %80 eğitim ve %20 doğrulama oranında ayrıştırılmıştır.

Veriler modele verilmeden önce normalize edilmiş, yani piksellerin değerleri 0–1 aralığına getirilmiştir. Ayrıca modelin daha genellenebilir bir hale gelmesi için yatay çevirme (horizontal flip), kırpma (random crop) ve parlaklık kontrast değişimi (brightness/contrast jitter) gibi veri artırma işlemleri eğitim öncesinde otomatik olarak uygulanmıştır.

Bu işlemler sonucunda, veri seti eğitim sürecine uygun hale getirilmiş ve modelin farklı çevresel koşullarda yüksek doğrulukla tahmin yapılabilmesi hedeflenmiştir.

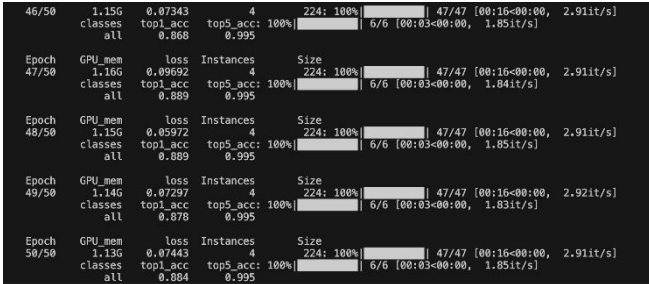
### III. Model Mimarisi

Bu projede, sınıflandırma için derin öğrenme tabanlı YOLOv8 mimarisi kullanılmıştır. Bu mimari hafif bir mimari olup sınıflandırma, nesne tespiti gibi görevlerde çok yüksek başarı oranı vermektedir. Modelin yapısı giriş katmanında 224x224 piksel boyutundaki görselleri alacak şekilde tasarlanmıştır. YOLO mimarisi içerisinde yer alan konvolüsyonel bloklar sabitlenmiş, sınıflandırma katmanları eğitim sürecinde optimize edilmiştir.

### IV. Model eğitimi

Bu projede, sınıflandırma için YOLOv8 mimarisi kullanılmış ve transfer öğrenme yaklaşımı uygulanmıştır. Eğitim sırasında, modelin gövde kısmındaki temel katmanlar korunmuş, son sınıflandırma katmanları veri setine uygun şekilde yeniden eğitilmiştir. Eğitim sürecinde Adam optimizasyon algoritması kullanıldı ve model toplamda 50 epoch boyunca eğitilmiştir. Train ve test verileri %80 e %20 olarak ayrıldı. Eğitim sırasında veri artırma teknikleri kullanıldı.

Eğitim sonunda model, doğrulama verisi üzerinde resim 2'de görüldüğü gibi yaklaşık %95 doğruluk oranı elde etmiştir. Bu sonuç, güneş paneli arıza sınıflandırması gibi hassas görevlerde YOLO mimarisinin etkili bir çözüm sunduğunu göstermektedir.



Epoch	GPU_mem	loss	Instances	Size	top1_acc	top5_acc
46/50	1.15G	0.07343	4	224: 100%	0.868	0.995
47/50	1.16G	0.09692	4	224: 100%	0.889	0.995
48/50	1.15G	0.05972	4	224: 100%	0.889	0.995
49/50	1.14G	0.07297	4	224: 100%	0.878	0.995
50/50	1.13G	0.07443	4	224: 100%	0.884	0.995

Resim 2 (Eğitim Aşaması)

### V. Gerçek Zamanlı Panel Tespiti

Bu çalışmada geliştirilen sistem, gerçek zamanlı olarak çalıştırıldığında güneş paneli içermeyen görsellerin gereksiz yere işlenmesini önleyerek sistem kaynaklarının verimli kullanılmasını amaçlamaktadır. Bu nedenle, sisteme aktarılan tüm görseller öncelikle bir filtreleme sürecinden geçirilmiş ve yalnızca potansiyel olarak güneş paneli içeren görüntüler işlenmeye devam etmiştir.

Panel tespiti amacıyla uygulanan bu ön filtreleme algoritması, OpenCV ve NumPy kütüphaneleri kullanılarak geliştirilmiştir. İşlem adımları şunlardan oluşmaktadır: Görüntü büyütme(Up-sampling), renk tabanlı maskeleme, kenar tespiti(Canny algoritması), kontür analizi.

Elde edilen kontür analizine göre, bir görselde belirli sayıda dikdörtgen benzeri yapı tespit edilmesi durumunda bu görsel "güneş paneli içeriyor" olarak sınıflandırılmıştır. Aksi durumda, görüntü sistemden elenmiş ve işlenmeden "nonsolar" kategorisine aktarılmıştır.

Bu yaklaşım sayesinde sistem, yalnızca anlamlı verilerle işlem yaparak hem performans artışı sağlamış hem de yanlış pozitifleri azaltmıştır.

### VI. Esp-Cam ile Fotoğraf Transferi Ve Çekilen Fotoğrafların Görüntü İşlemeye Aktarılması

Projede, insansız hava aracı (drone) üzerinde entegre edilen ESP-Cam modülü kullanılarak hem görüntü alınmakta hem de bu görüntülerin kablosuz olarak aktarımı sağlanmaktadır. ESP-Cam modülü, üzerindeki kamera sayesinde çevrenin fotoğraflarını çekebilme özelliğine sahiptir. Ayrıca içerisinde bulunan Wifi modülü sayesinde, yerel bir ağ (lokal host) oluşturabilir ve bu ağ üzerinden dışarıya veri aktarımı yapılabilir. Bu işlemin gerçekleştirilmesi için ESP-Cam, drone'un uçuş kontrol kartı olan Pixhawk(2.4.8)'in telem2 portunun 5V ve GND (toprak) pinlerine bağlanır ve böylece modülün çalışması için gerekli enerji sağlanmış olur. ESP-Cam'in açtığı lokal ağa hem bilgisayar üzerinden hem de doğrudan Pixhawk(2.4.8) üzerinden bağlantı kurulabilir. Böylece hem bilgisayardan gönderilen komutlar hem de uçuş kontrol kartı aracılığıyla iletilen komutlar drone'a aktarılabilir.

Drone uçuş halindeyken, ESP-Cam daha önceden ayarlanmış eşit sürelerde otomatik olarak sistem kapanana kadar fotoğraflar çeker. Çekilen bu fotoğraflar, ESP-Cam'in oluşturduğu lokal host üzerinden, API server'a aktarılır. Yani fotoğraflar doğrudan ağ üzerinden, herhangi bir harici depolama aygıtı olmadan bilgisayara iletilir. Sunucuda aktif olarak çalışan Python tabanlı yazılım, bu fotoğrafları belirlenen bir klasörün içine kaydeder ve böylece sonraki adımlar için hazır hale getirir.

Toplanan görüntüler daha sonra iki aşamalı bir filtreleme sürecinden geçer. İlk adımda, her bir fotoğrafta güneş paneli olup olmadığı tespit edilir. Bu işlem için çeşitli görüntü işleme ve derin öğrenme algoritmalarından yararlanılır. Eğer bir görüntüde güneş paneli bulunduğu tespit edilirse, o fotoğraf ayrı bir klasöre taşınır ve diğerlerinden ayrılır. Böylece, yalnızca güneş paneli içeren fotoğraflar üzerinde daha detaylı bir analiz gerçekleştirilir.

Filtrelenen ve güneş paneli içeren bu fotoğraflar, derin öğrenme tabanlı bir model ile incelenir. Bu model, güneş panellerindeki olası arızaları veya sorunları tespit etmek için eğitilmiştir. Görüntülerin analizi sonucunda, panel üzerinde çatlak, kirlenme, gölgeleme ya da başka bir arıza olup olmadığı belirlenir. Böylece, sadece ilgili ve gerekli olan fotoğraflar üzerinde çalışılarak, sistemin doğruluğu ve verimliliği artırılır. Sonuç olarak, bu proje sayesinde drone ile otomatik olarak veri toplanması, işlenmesi ve güneş panellerinin durumunun hızlı ve güvenilir şekilde tespit edilmesi mümkün hale gelmiştir.

### VII. SONUC

Bu proje güneş panellerin arızalarının otomatik tespit etmesi amacıyla derin öğrenme tabanlı sınıflandırma sistemi kullanılarak geliştirilmiştir. YOLOv8 mimarisi kullanılmıştır. YOLO kullanılarak bir model eğitilmiş, eğitilen bu model, temiz, tozlu, kuş pislikli, karlı, elektriksel hasarlı, fiziksel hasarlı güneş paneli görüntüleri sınıflandırılmıştır.

Model eğitilmeden önce veriler etiketlenmiş, tek aşamalı eğitim sürecinde model eğitilmiştir. Bu eğitim sonucunda doğruluk oranı yaklaşık %95'dir.

Proje gerçek zamanlı kullanım için genişletilip, drone üzerinde entegre çalışan ESP-32 CAM yardımıyla uçuş sırasında fotoğraflar çekilmiştir. Bu çekilen fotoğraflar ilk aşamada OpenCV ve NumPy kullanılarak geliştirilen algoritmada içeriğinde sadece güneş panelleri olan fotoğrafları klasörlere ayırmaktadır. Bunun amacı sistem performansını arttırmaktır. Bu işlem sonucunda gökyüzünde uçan drone'un altında olan ESP32-CAM anlık fotoğraf çekiyor, içerisinde panel barındıran fotoğrafları bir klasörde topluyor ve işlenmesi için YOLO mimarisi tarafından eğitilen modele yolluyor.

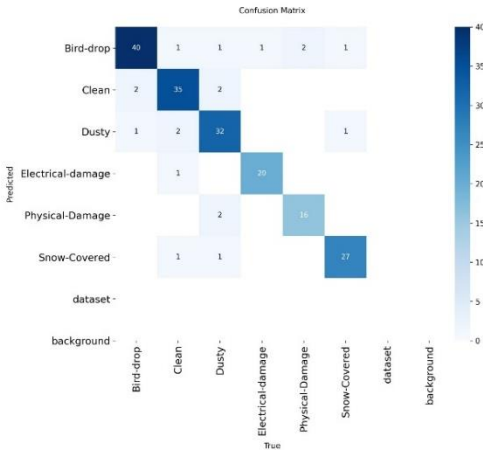
Sonuç olarak, geliştirilen bu sistem, geniş alan kaplayan güneş enerji santrallerinde bakım sürecini hızlandırmak, insan müdahalesini minimuma indirmek ve arızalanan güneş panellerinin en kısa sürede tespit edilip enerji verimliliğinin artırılması açısından önemli katkı sunmaktadır.

Resim 5' te görüldüğü üzere eğitim ve doğrulama kayıpları düzenli azalmış. Bu modelin düzgün öğrendiğinin ve aşırı öğrenme (overfitting) olmadığını göstergesi. Doğruluk (Accuracy) top1 e baktığımızda ise modelin doğru bilme oranı %95 seviyelerine kadar çıkmış. Buda projenin şuan için yeterli seviyede başarılı olduğunun göstergesidir.

## VIII. Kaynakça

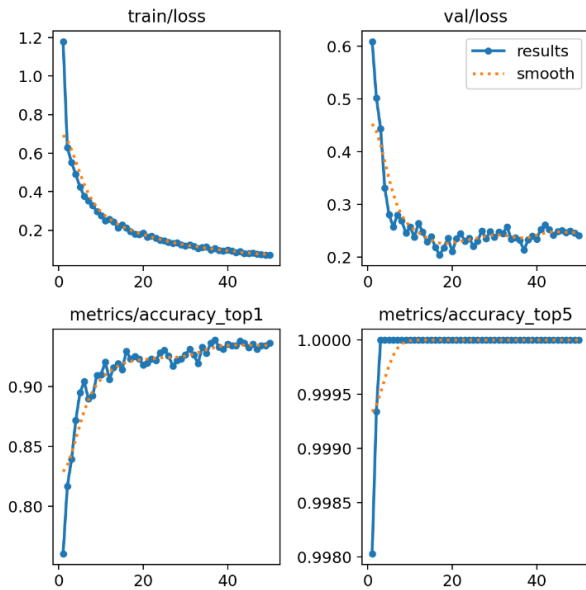
- [1] <https://www.kaggle.com/datasets/lotfiezzeddini/faulty-solar-panel>
- [2] <https://www.kaggle.com/datasets/gitenavnath/solar-augmented-dataset>

<https://www.robotistan.com/Data/EditorFiles/esp32cam.pdf>



Resim 4 (Sonuç Grafiği)

Resim 4' de görüldüğü gibi eğitimiz model Bird-drop, Clean ve Dusty sınıflarında yüksek doğruluk göstermiş. En fazla karışıklığı Dusty-Clean arasında yapmış.



Resim 5 (Eğitim Grafiği)

## CV



Alper Ceylan. -22040301071

İstanbul Topkapı Üniversitesi Mühendislik Fakültesi Yazılım Mühendisliği 3. Sınıf öğrencisi olarak eğitime devam etmekteyim. Python, C, C++ ve C# dillerine orta düzeyde hakimim. Okul kapsamında proje amaçlı yaptığım otopark sistemi, menü uygulaması, personel takip sistemi, dolandırıcılık sistemi ve retina optik disk hastalıklarını tanımlama gibi projelerde bireysel veya grup çalışmam oldu. Siber güvenlik ve kariyer planlama hakkında seminerlere katıldım ve tasarım desenleri üzerine btk akademiden aldığım bir adet sertifikam bulunmakta.

<https://github.com/alperccyln>



Emirhan Sertel. -22040301064

Topkapı Üniversitesi 3.sınıf yazılım mühendisliği öğrencisiyim. C, C++, Python dillerine hakimim. Yazılım mimarisinin SOLID prensiplerini kullanarak Otopark Yönetim Sistemi projesi, veri bilimi için programlama dilleri dersinde kaggle üzerinden dataseti çekerek Python kütüphaneleri üzerinden çalışmalar gerçekleştirdim. Birden çok duygu analizi sınıflandırma vs. içerikli projeler geliştirdim. Ekip arkadaşlarımla el hareketi ile kontrol edilebilen drone tasarlayıp İstanbul Fuar Merkezinde sergiledik. Başlangıç seviyesinde flutter kullanarak mobil uygulamalar üzerinde çalıştım.

<https://github.com/EmirhanSertel>



Ahmet Buğra Kaplan. -22040301028

İstanbul Topkapı Üniversitesi 3.sınıf yazılım mühendisliği öğrencisiyim. Kendimi android geliştirici olarak tanımlıyorum bu alanda bir çok clone proje ve kendi projelerim var. Bunun haricinde java ile design patternleri daha iyi kavramak için oluşturduğum konsol üzerinden savaş oyunu projem ve Otopark yönetim sistemi projem bulunmakta. Android - kotlin dışında java c# python gibi dillere başlangıç seviyesinde hakimim.

<https://github.com/AhmetBugraKaplan>