



YILDIZ TEKNİK ÜNİVERSİTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ  
YAPAY ZEKA DERSİ  
DÖNEM PROJESİ RAPORU

Youtube Linki -

Abdulkadir TÜRE 20011042

[abdulkadir.ture@std.yildiz.edu.tr](mailto:abdulkadir.ture@std.yildiz.edu.tr)

## *İçindekiler*

Proje Hakkında .....	3
Manuel Sürüş Verilerinin Toplanması .....	4
Yapay Sinir Ağı Model Tasarımı .....	5
YSA Modeli Eğitimi Ve Performans Değerlendirmesi .....	6
Nesne Tanıma Modeli Veri Toplama Ve Eğitimi .....	6
Otonom Sürüş Modelinin Oluşturulması ve Performansı .....	7
Sonuç .....	8
Proje Ortamı .....	8
Çalışma Örneği .....	8

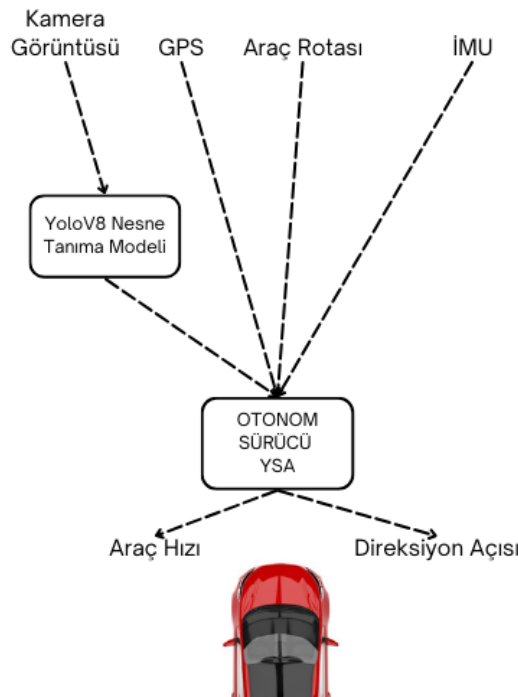
## Proje Hakkında

Bu proje, otonom araçlar için yapay zeka tabanlı bir kontrol modelinin oluşturulmasını ve simülasyon ortamında testlerini kapsamaktadır. Projenin ana amacı, belirli bir hedefe ulaşmak için aracın yönünü ve hızını kontrol edebilen bir yapay sinir ağı modeli geliştirmektir. Bu sinir ağının, hedefe doğru ilerlerken aynı zamanda çevresel etkilere bağlı olarak farklı kararlar verebilen bir yapıda olması amaçlanmıştır.

Projenin temel adımları şunlardır:

- Manuel Sürüş Verileri Toplama: Simülasyon ortamında gerçekleştirilen manuel sürüşler aracılığıyla, aracın davranışını ve çevresini temsil eden veri kümesi toplanmıştır.
- Yapay Sinir Ağı Eğitimi: Toplanan veriler, problemi çözmek için tasarlanan bir derin sinir ağı modelini eğitmek için kullanılmıştır. Bu model, aracın çevresini anlaması ve güvenli bir şekilde hareket etmesi için gerekli olan kararları almasını sağlar.
- Nesne Tespiti İçin Veri Toplama ve Etiketleme: Aracın kamerasından alınan bilgilerin anlamlandırılarak karar verici yapay sinir ağına iletilmesi için bir nesne tespiti modeline ihtiyaç duyulmuştur. Bu aşamada simülasyon ortamındaki trafik ışıkları ve yol tabelalarının görselleri toplanmış ve uygun şekilde etiketlenmiştir.
- Nesne Tespiti: Görsel veri seti YOLOv8 modelini eğitmek için kullanılmıştır.

Projede kullanılan otonom sürüş algoritmasının genel yapısı, şekil 1'de verilmiştir.



Şekil 1 Otonom Sürüş Genel Yapısı

## Manuel Sürüş Verilerinin Toplanması

Projenin başlangıç aşamasında ROS 2 üzerinde vehicle/gps ve vehicle/imu konularına abone olunarak manuel sürüş esnasında gps bilgisi ve aracın yönelim bilgileri toplanmıştır. Belirli bir frekansta toplanan bilgilerle veri setini oluşturmuştur. Veri setinde bulunan sütunlara ait bilgiler aşağıdaki gibidir;

- dx : AraçX – HedefX (Navigasyon tarafından oluşturulan yolda aracın önünden K mesafede bulunan nokta ile aracın bulunduğu noktanın x eksenindeki farkı)
- dy : AraçY – HedefY (Navigasyon tarafından oluşturulan yolda aracın önünden K mesafede bulunan nokta ile aracın bulunduğu noktanın y eksenindeki farkı)
- Araç Yönelimi : (W) (İmu sensöründen alınan ve aracın Z ekseninde yönelimini ifade eder.
- Kırmızı Işık : (0 - 1) (Manuel sürüş esnasında araç kırmızı ışık nedeniyle durduğunda bu değişken 1 değerini alır.)
- atan2(dY/dX) : arctan fonksiyonun özel versiyonu olarak dy ve dx ile hesaplanır.
- Lidar Verisi ÖN : (0 - 1) (Araç önündeki bir engelden dolayı durduysa bu değişken 1 değerini alır.)
- Lidar Verisi SOL : (0 - 1) (Aracın solunda K eşik değerinden yakın bir nesne varsa bu değişken 1 değerini alır.)
- Lidar Verisi SAG : (0 - 1) (Aracın sağında K eşik değerinden yakın bir nesne varsa bu değişken 1 değerini alır.)
- Hız : Aracın anlık hızını ifade eder.
- Direksiyon Açısı: Aracın anlık direksiyon açısını ifade eder.

1	w,dx,dy,atan2_dx_dy,lidar_on,lidar_sol,lidar_sag,trafik_isk,hiz,direksiyon_aci
2	3.141591586398842,-6.1707103757840205,0.9220806360167586,2.9932615754406378,0,0,0,0,6.0,0.1482384204864502
3	3.1413315719866692,-6.155271821312182,0.9218827050510825,2.99292643776546,0,0,0,0,6.0,0.1486777365207672
4	3.137708574960797,-6.073560077239405,0.9132597635714461,2.9923443235127762,0,0,0,0,6.0,0.1475309282541275
5	3.129700918952487,-6.762474800836863,0.9293565268787418,3.0050197000810415,0,0,0,0,6.0,0.12799072265625
6	3.1227830156285017,-6.613126174397273,0.9005226258805621,3.006253081103878,0,0,0,0,6.0,0.11951812356710434
7	3.115617201590665,-6.447139461302292,0.8687554698062669,3.007648936938015,0,0,0,0,6.0,0.11079218983650208
8	3.1108828239467488,-6.280910544144717,0.8375131158205633,3.009031989360352,0,0,0,0,6.0,0.10251308977603912
9	3.1051507364512854,-6.147846204233051,0.8133176951207375,3.0100633135750074,0,0,0,0,6.0,0.09632384777069092
10	3.0992694292845036,-5.998150938094902,0.7870922909036153,3.0111156664336214,0,0,0,0,6.0,0.08987198024988174
11	3.0938289310830522,-6.685526567903757,0.7910677301209788,3.0238148628354375,0,0,0,0,6.0,0.07107403129339218
12	3.089025369339735,-6.5194817664622775,0.7642355729472854,3.0249018418432603,0,0,0,0,6.0,0.06547532975673676
13	3.0848736894569626,-6.403348325900492,0.7465809786835464,3.0255244154648326,0,0,0,0,6.0,0.061362773180007935
14	3.081141962408143,-6.221029997950797,0.719989756107303,3.026370463485541,0,0,0,0,6.0,0.05622958391904831
15	3.0777219689213675,-6.038920398246557,0.6942259001847546,3.0271364759096526,0,0,0,0,6.0,0.05158471688628197
16	3.0748885306724514,-6.776789020597748,0.7108209038024853,3.0370842863251957,0,0,0,0,6.0,0.03903094306588173
17	3.0720037306388708,-6.611395131417618,0.6888576898564196,3.037774771738162,0,0,0,0,6.0,0.035467538982629776
18	3.0701956634767127,-6.479060739214216,0.6722742638967105,3.0382015744644146,0,0,0,0,6.0,0.03310101106762886
19	3.0681696177660958,-6.346700005897439,0.6562532859912622,3.0385581076406143,0,0,0,0,6.0,0.031008001416921616
20	3.066134840758388,-6.114994159728186,0.6290050281351682,3.0390904200379905,0,0,0,0,6.0,0.027768833562731743
21	3.0647716560331872,-5.982546394466208,0.6136585905965717,3.0393753344154257,0,0,0,0,6.0,0.026064066216349602
22	3.0632225458683338,-6.701061528537988,0.6170864375621576,3.049763780907978,0,0,0,0,6.0,0.014208588749170303
23	3.0621955143182684,-6.551931880415646,0.6006631589856042,3.0501709472103915,0,0,0,0,6.0,0.0125508988276124
24	3.0614027769184533,-6.402692568067753,0.5853287096825142,3.0504269118201948,0,0,0,0,6.0,0.01139797456562519

Şekil 2 Manuel Sürüş Veri Seti Örnek Görüntüsü

## Yapay Sinir Ağı Model Tasarımı

Modelin girdi ve çıktıları aşağıdaki gibi belirlenmiştir.

Girdiler;

- Dy
- Dx
- W
- Kırmızı Işık
- $\text{Atan2}(dy/dx)$
- Lidar Ön

Çıktılar;

- Direksiyon Açısı
- Araç Hızı

Gerçekleştirilen deneysel testler sonucunda 2 gizli katmana sahip bir derin sinir ağı modeli oluşturulmuştur. Aktivasyon fonksiyonu olarak reLu, hata hesaplamaları için kareler farkı yöntemi, optimizer olarak Adam optimizer tercih edilmiştir. Modele ait yapı şekil 3’de verilmiştir.

```
class NeuralNetwork(nn.Module):
    def __init__(self):
        super(NeuralNetwork, self).__init__()
        self.layer1 = nn.Linear(6, 32)
        self.layer2 = nn.Linear(32, 16)
        self.output = nn.Linear(16, 2)

    def forward(self, x):
        x = torch.relu(self.layer1(x))
        x = torch.relu(self.layer2(x))
        x = self.output(x)
        return x

model = NeuralNetwork()

criterion = nn.MSELoss()
optimizer = optim.Adam(model.parameters(), lr=0.001)
```

Şekil 3 Yapay Sinir Ağı Modeli

## YSA Modeli Eğitimi Ve Performans Değerlendirmesi

Veri seti test ve train olarak ayrılmış ve Google colab ortamında model train edilmiştir. Train sonuçlarına ilişkin Loss değerleri aşağıdaki gibidir.

- Epoch [10], Loss: 0.0127
- Epoch [20], Loss: 0.0118
- Epoch [30], Loss: 0.0089
- Epoch [40], Loss: 0.0042
- Epoch [50], Loss: 0.0025
- Epoch [60], Loss: 0.0013
- Epoch [70], Loss: 0.0007
- Epoch [80], Loss: 0.0004
- Epoch [90], Loss: 0.0003
- Epoch [100], Loss: 0.0002

Modele ait test sonucu;

Result: 0.002 olarak hesaplanmıştır.

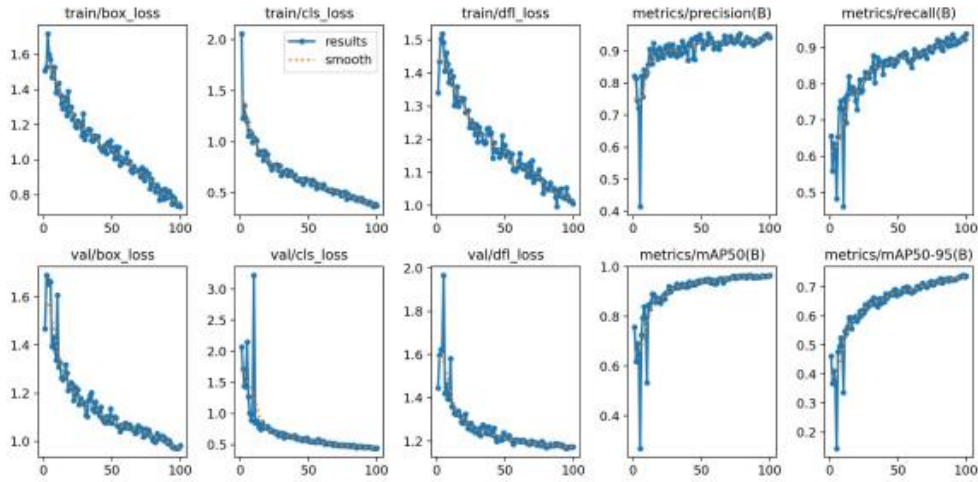
## Nesne Tanıma Modeli Veri Toplama Ve Eğitimi

Nesne tespiti için 4 farklı sınıf belirlenmiş ve bu sınıflara ait toplamda 252 adet görsel etiketlenmiştir. Sınıflar aşağıdaki gibidir;

- dur
- hız-limit
- kırmızı-ışık
- yeşil-ışık

Proje kapsamında aracın hareketi için kırmızı ve yeşil ışık tespitleri kullanılmıştır. Nesne tanıma modeli 4 sınıfı tespit etmesine rağmen, otonom sürüş ysa modeline yalnızca kırmızı ve yeşil ışık nesne tespitine ait bilgiler gönderilmektedir.

Toplanan görseller roboflow üzerinde etiketlendikten dosya YoloV8 modeli kullanılarak eğitilmiştir. Nesne tespit modeline ait performans grafikleri aşağıdaki gibidir.



## *Otonom Sürüş Modelinin Oluşturulması ve Performansı*

Elde edilen 2 yapay zeka modeli birleştirilerek ROS 2 üzerinde otonom kontrolcü yazılmıştır. Kontrolcüye başlangıçta navigasyon tarafından oluşturulan yol noktaları verilmiş ve sonrasında aracın hedefe ulaşma durumu gözlenmiştir. Test senaryolarına göre performans aşağıdaki gibidir.

### 1. Test

Toplam Yol Uzunluğu	348 m
Araç Tarafından Alınan Yol	363 m
Araç Tarafından Yola Bağımlılık (%)	%97
Kırmızı Işığa Bağlı Durma Oranı (%)	%99
Kırmızı Işık Tespit Etme Oranı (%)	%82
Engele Çarpmadan Frenleme Oranı (%)	%76

### 2. Test

Toplam Yol Uzunluğu	189 m
Araç Tarafından Alınan Yol	201 m
Araç Tarafından Yola Bağımlılık (%)	%98
Kırmızı Işığa Bağlı Durma Oranı (%)	%100
Kırmızı Işık Tespit Etme Oranı (%)	%76
Engele Çarpmadan Frenleme Oranı (%)	%82

### 3. Test

Toplam Yol Uzunluğu	512 m
Araç Tarafından Alınan Yol	561 m
Araç Tarafından Yola Bağımlılık (%)	% 95
Kırmızı Işığa Bağlı Durma Oranı (%)	% 99
Kırmızı Işık Tespit Etme Oranı (%)	% 73
Engele Çarpmadan Frenleme Oranı (%)	%71

\* Araç tarafından yola bağımlılık yüzdesi, navigasyon tarafından çizilen yol ile araç tarafından çizilen yolun 2 boyutlu düzlemde birbirine göre farkının yüzde cinsinden hesaplanmasına dayanmaktadır.

\* Kırmızı ışığa bağlı durma oranı, nesne tespit modeli tarafından kırmızı ışık varlığı belirlendikten sonra aracın karar verme sisteminin durma oranını ifade etmektedir.

\* Kırmızı ışık tespit etme oranı, ilerlenen yol üzerinde bulunan kırmızı ışıklardan kaçının doğru tespit edilmesine bağlı olarak hesaplanmıştır.

\* Engele çarpmadan frenleme oranı, aracın karar verme mekanizmasının nesneye çarpmadan durma oranını ifade etmektedir.

## *Sonuç*

Elde edilen sonuçlara göre yaklaşık 2 saatlik manuel sürüş verilerinden eğitilen modelin farklı senaryolarda (ön engel , kırmızı ışık) karar verme oranının kısmen yüksek olduğu görülmüştür. Manuel senaryolarda kırmızı ışıkta durma durumunun örnekleme yüksek olduğundan test senaryolarında bu durum için model yüksek performans göstermiştir. Modelin navigasyon tarafından belirlenen yolu takip etme başarısının ise oldukça yüksek olduğu görülmüştür.

Proje sonucunda elde edilen modelin otonom araçlar için başlangıç niteliğinde bir proje olarak başarılı olduğu, karar vermeden sorumlu sinir ağının geliştirilmesiyle otonom sürüşün farklı senaryolarda da doğru kararları alarak aracı başarılı bir şekilde yönlendirebileceği anlaşılmıştır.

## *Proje Ortamı*

Proje ortamı, Ubuntu 22.04 işletim sistemi, ROS2 humble, Webots simülasyon aracı ve webots\_ros2 paketlerinden oluşmaktadır. Ayrıntılı kurulum için kaynak kod klasöründeki dökümantasyonu inceleyebilirsiniz.

## *Çalışma Örneği*

Simülasyon ortamında çalışma gerçekleştirildiğinden demo videosu, raporun kapak sayfasında linki verilen videoda son kısımda yer almaktadır.



## *Yararlanılan Kaynaklar*

**Amazon Web Services (AWS):** Yapay sinir ağıları hakkında genel bilgiler, derin öğrenme ve sinir ağlarının nasıl çalıştığı hakkında açıklamalar. <https://aws.amazon.com/tr/what-is/neural-network/>

**Cyberbotics:** Webots simülasyon ortamının resmi web sitesi. <https://cyberbotics.com/>

**ROS (Robot Operating System):** ROS'un resmi dökümantasyon sitesi. <https://docs.ros.org/en/humble/index.html>

**Ultralytics:** YOLOv8 nesne tespiti modeli hakkında teknik bilgiler ve eğitim kılavuzları. <https://docs.ultralytics.com/modes/train/>

**Wikipedia - Ackermann Steering Geometry:** Ackermann direksiyon geometrisi hakkında detaylı bilgiler. [https://en.wikipedia.org/wiki/Ackermann\\_steering\\_geometry](https://en.wikipedia.org/wiki/Ackermann_steering_geometry)

**Amazon Web Services (AWS):** Yapay sinir ağıları hakkında genel bilgiler. <https://aws.amazon.com/tr/what-is/neural-network/>