

# TEKNOFEST

## HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

**ROBOTAKSİ – BİNEK OTONOM ARAÇ**

**YARIŞMASI KRİTİK TASARIM RAPORU**

**TAKIM ADI: DUSCART Otonom Takımı**

**TAKIM ID: 468390**

## İçindekiler

1.	Takım Organizasyonu .....	3
2.	Ön Tasarım Raporu Değerlendirmesi .....	5
3.	Araç Mekanik Özellikleri.....	6
3.1.	Direksiyon Sistemi.....	8
3.1.1.	Direksiyon Sistemi Elemanları .....	9
3.1.2.	Salıncak Sistemi Elemanları ve Analiz Sonuçları .....	15
3.2.	Fren Sistemi.....	20
3.3	Elektronik Gaz Sistemi .....	24
4.	Donanım Mimarisi.....	25
4.1.	Yapay Zeka Bilgisayarı.....	25
4.2.	Araç Kontrol Kartı .....	26
4.3.	Batarya Yönetim Sistemi .....	28
4.4.	Kablosuz Haberleşme Sistemi .....	28
4.5.	Sensörler.....	29
4.5.1.	Lidar .....	30
4.5.2.	Kamera .....	34
4.5.3.	GPS .....	39
5.	Yazılım Mimarisi .....	42
5.1.	Nesne Tespiti .....	42
5.2.	Şerit Tespiti ve Takip Algoritması .....	47
5.3.	Parkurdaki Engellerin Tespiti ve Şerit Değiştirme .....	56
6.	Özgün Bileşenler .....	60
7.	Test .....	62
8.	Referanslar .....	68

## 1. Takım Organizasyonu

DUSCART Otonom Takımımız 20 kişiden oluşmaktadır. Bunlar Bilgisayar Mühendisliği, Elektrik Elektronik Mühendisliği, Makine Mühendisliği ve Endüstri Mühendisliği bölümlerinde lisans eğitimlerine devam eden öğrencilerdir.

Takım akademik danışmanı ve kaptanı öncülüğünde kendi çalışma planımızı oluşturarak otonom araç çalışmalarına başladık ve devam etmekteyiz.

Takımımız Bilgisayar Birimi, Elektrik- Elektronik Birimi, Mekanik Birimi ve Sponsorluk Birimlerinden oluşmaktadır.



Tablo 1 Takım Organizasyon Şeması

Yapılacak İş	Mart	Nisan	Mayıs	Haziran	Temmuz
Simülasyon gerçekleştirilmesi					
Veri seti oluşturma					
Özgün nesne tespiti eğitim modeli					
Rapor sunumlarını gerçekleştirmesi ve simülasyon gösterimlerinin yapılması					
Şerit tanıma algoritması					
Araç kontrol ünitesi tasarımı					
Telemetri yazılımı					
Kritik tasarım raporunun hazırlanması					
Direksiyon ve fren sistemi araç üzerinde montajı					
Yazılımın araç üzerinde test edilmesi					
Robotaksi Binek Otonom Araç Yarışı					

Tablo 2 İş Zaman Grafiği

Birimler	Alt İşlevler	İş Tanımı
<b>Bilgisayar</b>	<b>Nesne Tespiti</b>	Trafik tabelaları ve ışıklarını gerçek zamanlı tespit etmek için doğru algoritmaları ve verileri kullanarak model eğitiminden sorumludur.
	<b>Sensör Verilerinin İşlenmesi</b>	Lidardan alınan verilerin ve kameradan alınan görüntülerin aracımızın engel tespiti, şerit takibi ve şerit değiştirme görevlerinin gerçekleştirilmek üzere işlenmesinden sorumludur.
	<b>Şerit Takip Algoritması</b>	Aracın seyir halinde iken kullandığımız kamera aracılığı ile şeritleri çeşitli koşullarda başarılı bir şekilde tespit edip yolu ortayaarak parkuru tamamlamasından sorumludur.
	<b>Simülasyon</b>	Simülasyon ortamında aracı ve ortamı modelleyerek kullanılan algoritmaların doğruluğunun güvenli bir şekilde test edilmesinden sorumludur.
<b>Mekanik</b>	<b>Fren Sistemi</b>	Fren sistemi için kullanılacak motorun araç üzerine entegre edilmesinden sorumludur.
	<b>Direksiyon Sistemi</b>	Direksiyon sistemi için kullanılacak motorun araç üzerine entegre edilmesinden sorumludur.
	<b>Genel Mekanik</b>	Araç üretimi ve yürüs sistemin geliştirilmesinden sorumludur.
<b>Elektrik-Elektronik</b>	<b>Kontrol Kartı Tasarımı</b>	Yapay zeka bilgisaylarından gelen verileri doğru alt sistemlere aktarmak için kullanılacak kartın tasarımından sorumludur.
	<b>Uzaktan Haberleşme</b>	Tasarlanan arayüz ile aracın dışarıdan kontrol edilmesini sağlar.
	<b>Motor Kontrolleri</b>	Sensörlerden gelen verileri yazdığımız algoritmalarla işledikten sonra kontrol kartına doğru komutlar gönderilmesinden ve direksiyon yönü ile fren sisteminin ayarlanması sorumludur.
<b>Sponsorluk</b>	<b>Tanıtım ve Yaygınlaştırma</b>	Takımımızın sosyal medya hesaplarının yönetilmesi ve tanıtım yapılması sorumludur.
	<b>Ürünlerin Temini</b>	Gerekli firmalarla görüşerek ürün desteği sağlanmasından sorumludur.

Tablo 3 Birimler ve İş Tanımları

## 2. Ön Tasarım Raporu Değerlendirmesi

Ön tasarım raporundan sonra devam ettiğimiz çalışmalar esnasında bazı değişikliklerin daha iyi sonuçlar vereceği kararına vardık.

- Otonom aracımızda ortamın haritalandırılması, engellerin tespiti ve mesafe bilgilerinin alınması için ön tasarım raporunda belirttiğimiz RPLIDAR A3 M1 ve Veldoyne Puck sensörlerinden RPLIDAR A3 M1’i kullanmaya daha uygun fiyatlı olması ve temin sürecinin kolaylığından dolayı karar verdik. Şu an seçtiğimiz lidarla da otonom sürüş algoritmalarımızı gerçekleştirebilmemiz için doğru verileri alabildiğimizi gördük.
- Araçta nesne ve şerit tespiti için belirttiğimiz Zed2 ve stereo kameralarından stereo kameranın şu an kullanacağımız sistemi desteklemediği için kamera değişikliğine gittik. Araçta şerit verilerini alacağımız kamerayı Zed mini olarak değiştirdik.
- Aracımızın direksiyon sistemine ES M23480 enkoderli 8NM step motoru entegre etmemi düşünüyorduk ancak yaptığımız çalışmalar sonucunda aracımızın direksiyonu kontrol etmek için Redüktörlü DC Motor kullandık. 5 voltlu ALDP105W röleler kullanarak motor hareketini sağlıyoruz.
- Şerit takip algoritmasında kullandığımız threshold filtresinde istenilen eşik değer için denemelerimiz sonucunda durumumuza daha uygun değerler bulduk. Aynı zamanda kullandığımız canny ve gaussianblur filtrelerinde de daha uygun parametreler olduğuna karar vererek algoritmamızı güncelledik. Bu sayede daha temiz bir görüntü elde ederek daha doğru bir şerit tespitine ulaştık.
- Aracın nesne tespiti algoritmasında öncelikle Yolov3 ve küçük boyutlarda olmak üzere Yolov5, Yolov3-Tiny modelleri test edilmiştir. Fakat Elde edilen sonuçların yetersizlikleri ve düşük doğruluk sonuçları yüzünden veri setine ve kullanılan sisteme göre en uygun olan model Yolov4 kullanılmaya karar verilmiştir. Veri setimizdeki düşük kalitedeki fotoğrafları sildik ve yerlerine daha temizleri eklenerek veri seti arttırlı.
- Aracın otonom sürüşünü sağlaması için yapay zeka algoritmalarını çalıştıracağımız bilgisayarı NVIDIA Jetson AGX Xavier olarak düşünmüştük. Ancak ülkemizdeki stok probleminden dolayı ekran kartı güçlü olan bir dizüstü bilgisayar modelini seçtik. Dizüstü bilgisayarımız da işlem yönünden güçlü bir bilgisayar olduğundan ihtiyacımızı kolaylıkla sağlayıp gelen verileri işleme görevini başarıyla sağlayacaktır.

Ön Tasarım Raporu Bütçe Planı		Kritik Tasarım Raporu Bütçe Planı	
Ürün	Fiyat	Ürün	Fiyat
NVIDIA Jetson AGX Xavier	999 \$	Dizüstü Bilgisayar	35.600 ₺
ZED 2	449 \$	ZED 2	449 \$
SteereoCam	299 \$	ZED Mini	399 \$
Velodyne Puck	~60.000 ₺	RP LIDAR A3M1	11.500 ₺
Ultrasonik Mesafe Sensörü	100 ₺	TF Mini Lidar x3	940 ₺ x3
IMU Sensör	80 ₺	GPS Modül	78 ₺
DRF7020D7 Dorji	796 ₺	DRF7020D7 Dorji	796 ₺
Step Motor ve Sürücü	2000 ₺	HC05 Bluetooth Modül	90 ₺
Lineer Motor	1020 ₺	DC Motor	1000 ₺
		Lineer Motor	1020 ₺
		AKS Komponentler	1300 ₺
		BMS Komponentler	2000 ₺

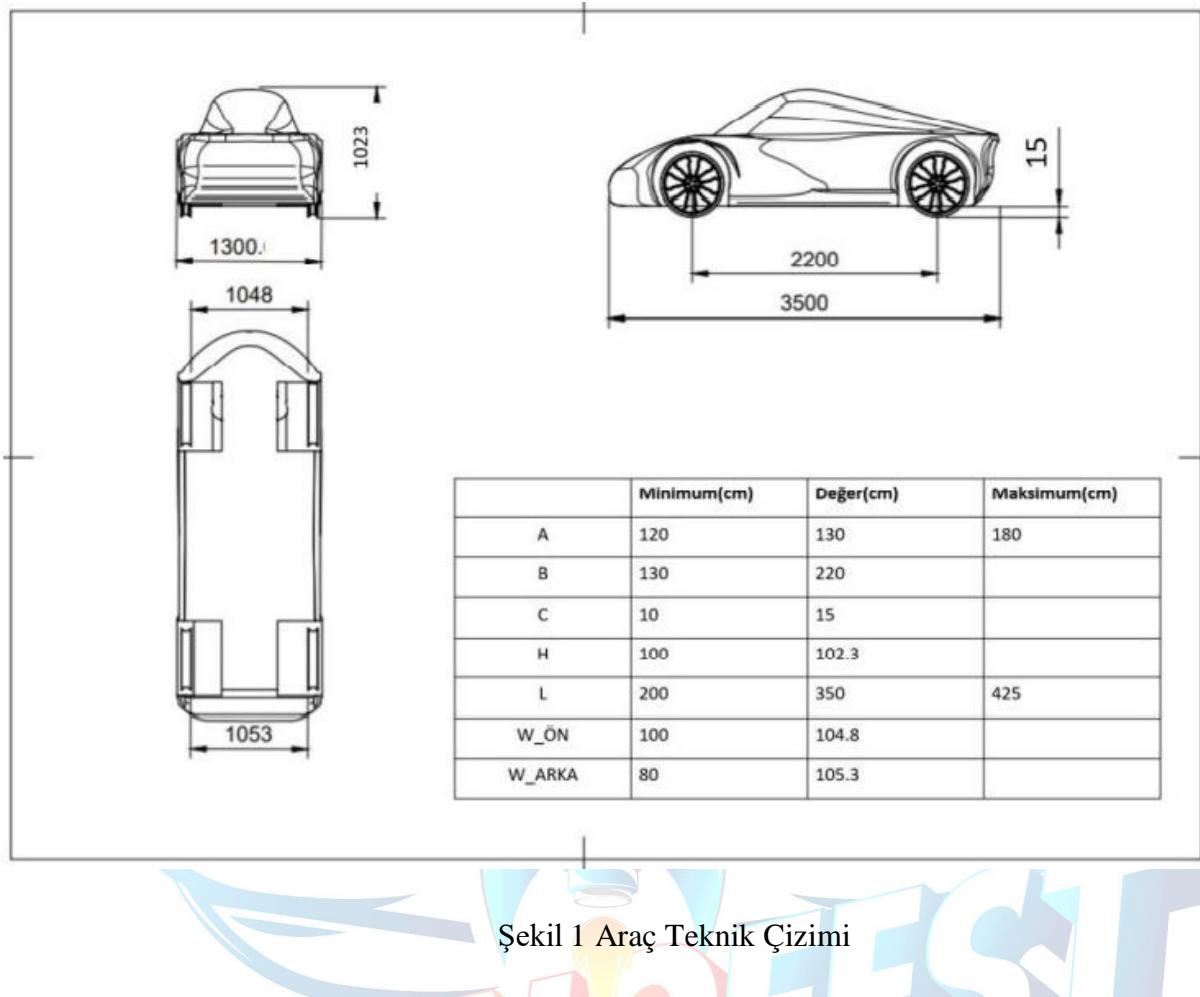
Tablo 4 Bütçe Planı

### 3. Araç Mekanik Özellikleri

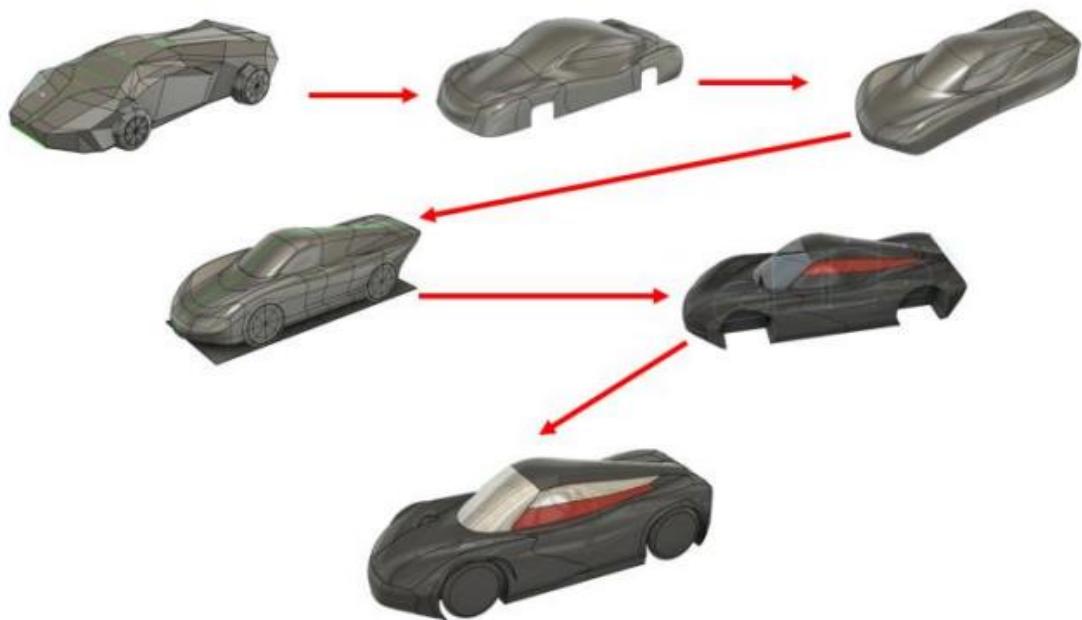
Aracımızın kaportası karbon fiber malzemeden vakum infüzyon yöntemi ile üretilmiştir. Araç şasesi ise karbon fiber malzemeden, sandwich yöntemi ile ara katlara airex destekler atılarak vakum torbalama yöntemi ile üretilmiştir. Aracın yürüyor aksamında AL 7075 malzeme kullanılmış olup, gerekli noktalarda 1040 çeliği kullanılmıştır. Direksiyon ve fren sisteminde st-37 çelik, karbon fiber, 4050 çeliği kullanılmış olup, kontrol için kullanılacak olan motorların montajı gerçekleştirilmiştir.

Özellik	Birim	Değer
Araç Uzunluğu	mm	3500
Araç Genişliği	mm	1300
Araç Yüksekliği	mm	1030
Şasi	Malzeme	Karbon Fiber
Kabuk	Malzeme	Karbon Fiber
Motor	Tip	PMSM
Motor gücü	kW	3
Batarya paketi kapasitesi	Ah	39,9
Batarya paketi maksimum gerilimi	V	109
Batarya paketi enerjisi	Wh	3016,44
Araç Ağırlığı	kg	143
Tekerlek Sayısı	adet	4
Teker Genişliği	mm	72

Tablo 5 Araç Özellikleri Tablosu



Şekil 1 Araç Teknik Çizimi



Şekil 2 Tasarım evrelerinin üst kabuk baz alınarak spesifik yaklaşımlar ile oluşturulması

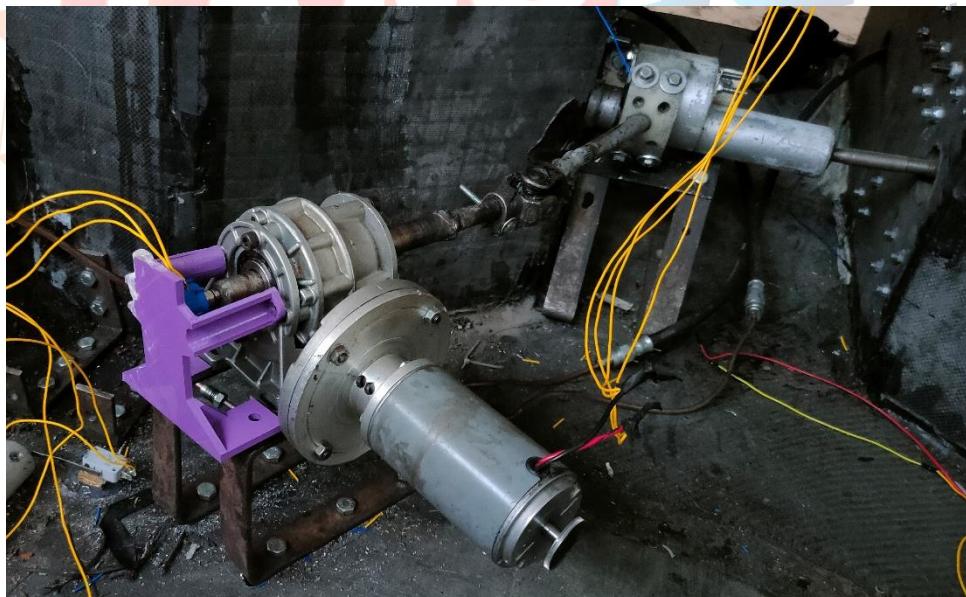
### 3.1. Direksiyon Sistemi

Direksiyon sistemimizde kullandığımız redüktörlü DC motoru çift yönlü ve istenilen açıda döndürmek için 5 voltlu ALDP105W roleler kullanarak H – Bridge devresi oluşturduk bu sayede motorumuzun çift yönlü dönmesini sağladık.

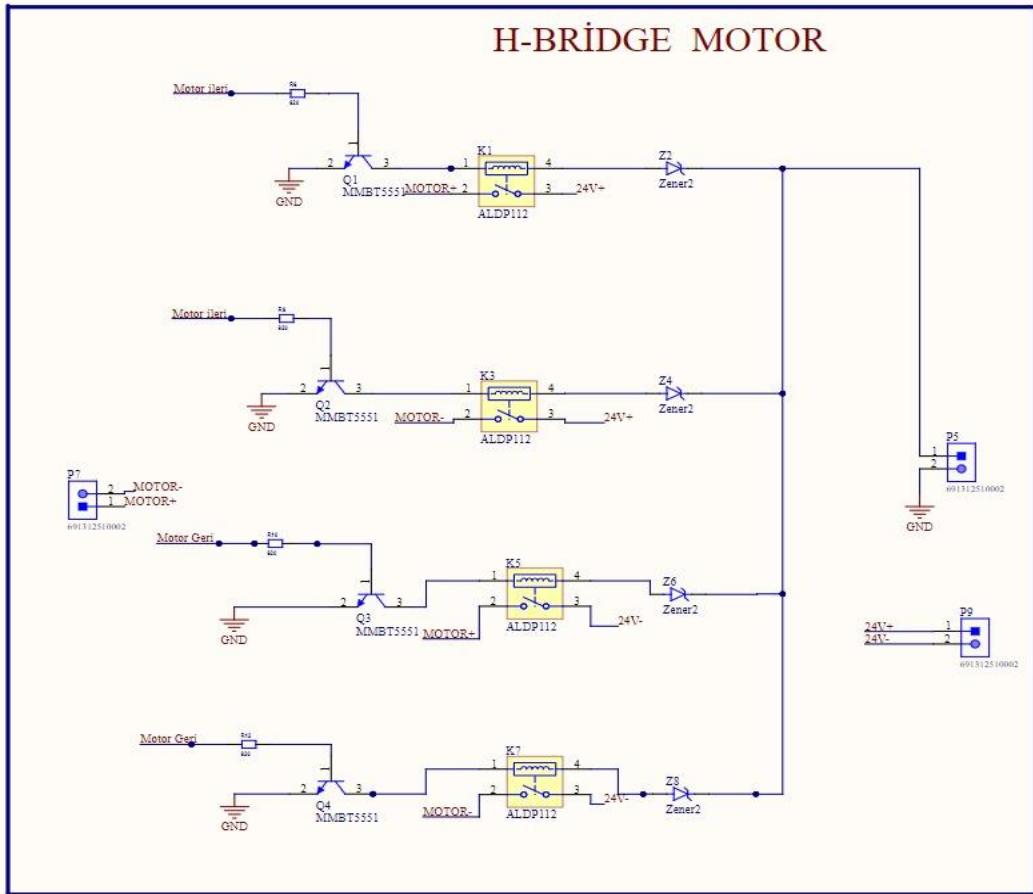
Direksiyon sistemimizi istediğimiz açıda döndürebilmek için direksiyon sistemimize bağlı olan milin ucuna 10K lık çok turlu potansiyometre bağladık. Potansiyometreden gelen açı bilgisi sayesinde direksiyon sistemimizi istediğimiz açıda kontrolünü sağladık.



Şekil 3 Elektrikli Direksiyon Sistemi Tasarımı



Şekil 4 Montajlanmış Direksiyon Sistemi



Şekil 5 Araç Kontrol Kartında Bulunan Direksiyon Motoru Kontrol Devresi Şeması

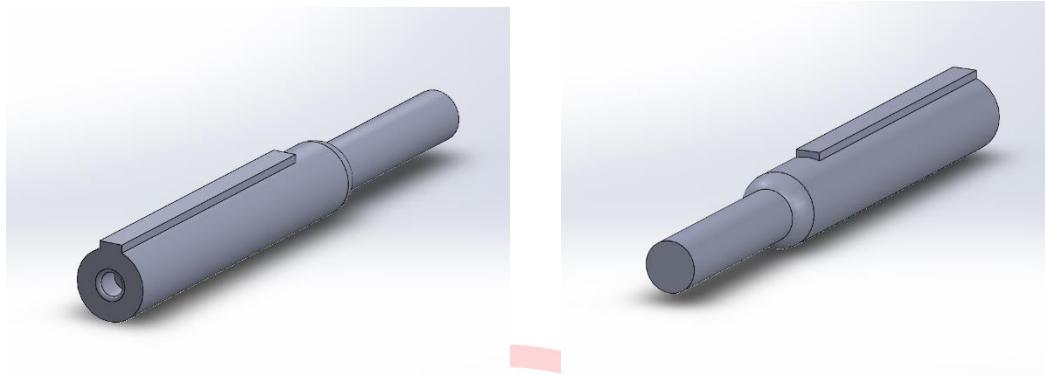
### 3.1.1. Direksiyon Sistemi Elemanları

**DC Motor:** Direksiyon sisteminin ilk elemanı olan direksiyon simidinin çıkarılıp yerine direksiyon hareketini oluşturabilmesi için DC motor entegre edilmiştir. Direksiyon sisteminde tekerlerin x ekseninde pozitif veya negatif yönde hareketinin sağlanabilmesi için DC Motor tercih edilmiştir. DC Motorun içerisinde bulunan sonsuz dişli sayesinde tekerlerde istenilen açıyı ve istenilen açının sağlanabilmesi için gerekli kuvveti DC Motorun karşılayabileceğini belirledik ve montaj yaparak testini gerçekleştirdik.



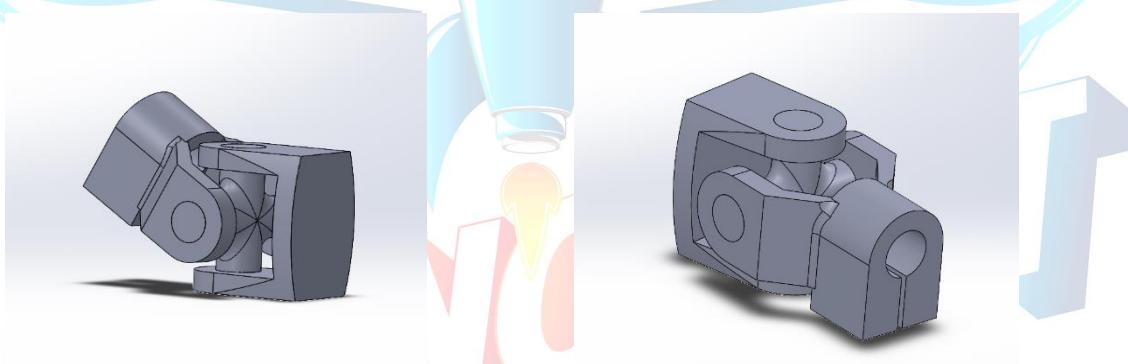
Şekil 6 DC Motor

**Direksiyon Kolonu:** DC motordan çıkan kuvvetin uyguladığı dönme hareketini, direksiyon sisteminin diğer elemanlarına iletilmesini sağlayan silindir şeklindeki parçadır. Direksiyon kolonu alüminyum 6061-T6 serisinden üretilmiştir.



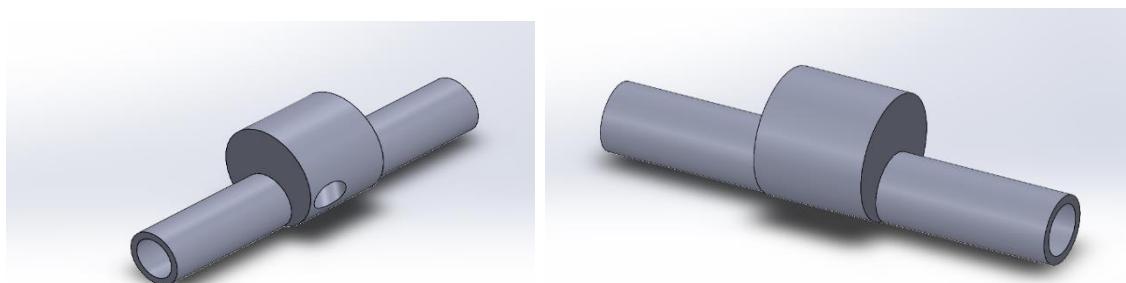
Şekil 7 Direksiyon Kolonu

**Mafsal:** DC motordan direksiyon kolonuna iletilen dairesel kuvvetin helisel pinyon dişliye ulaşana kadar aradaki açı farkından kaynaklanan problemlerin önüne geçebilmek için kullanılan ve iki eksende hareket edebilen hareketli parçadır. Mafsal DIN 808 ST standartıdır.



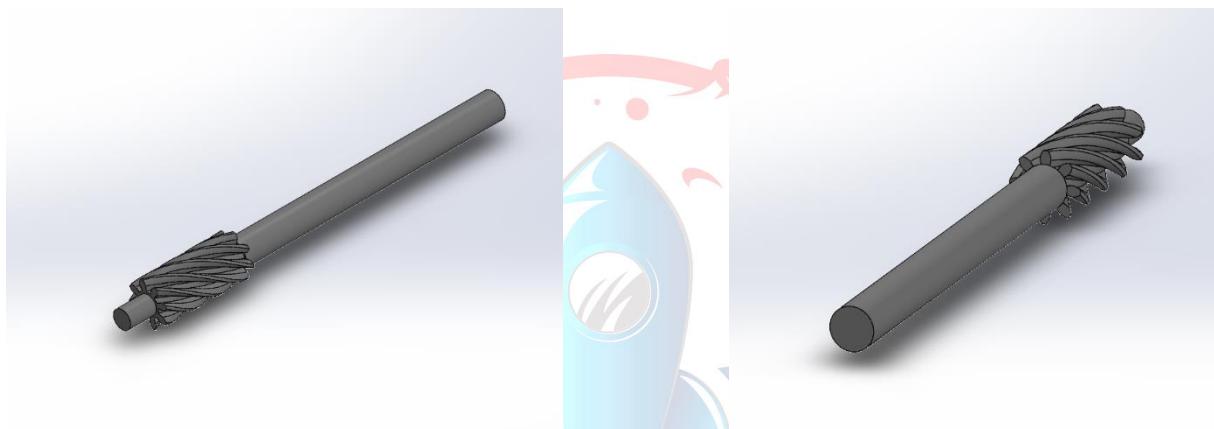
Şekil 8 Mafsal

**Direksiyon Kutusu:** Direksiyon kutusundaki helis pinyon yuvasının açılı olması sayesinde kremayer ve pinyonu doğru açıda ve sabit şekilde tutarak dişlerin birbirine eksiksiz geçmesini sağlar. Direksiyon kutusu hafiflik ve kolay bulunabilirlik açısından poliamit malzemeden üretilmiştir.



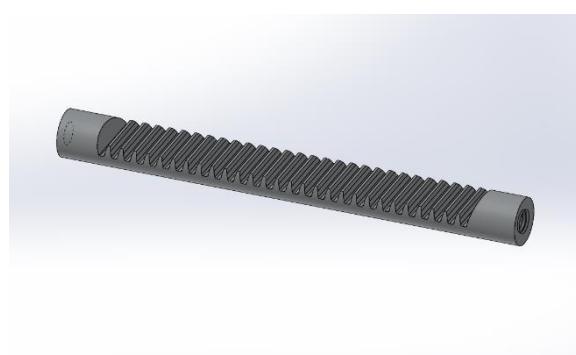
Şekil 9 Direksiyon Kutusu

**Helisel Pinyon Dişli:** DC motorun uyguladığı dairesel döndürme kuvvetinin sırasıyla direksiyon kolonu, mafsal ve helisel pinyon dişliye ulaşmasıyla mafsal ve helisel pinyon dişli birbirine montajlı olduğundan dolayı helisel pinyon dişli dairesel olarak DC motorun uyguladığı döndürme açısı kadar döner. Düz dişli yerine helisel dişli tercih etmemizin sebepleri ise; helisel dişliler düz dişlilere göre daha mukavemetlidir ve düz dişliler açılı hareket iletimine müsait değilken helisel dişliler açılı hareket iletimine müsaittir. Helisel dişlinin düz dişliye kıyasla noktasal değil yüzeysel temasından dolayı kremayerdeki helisel diş yuvalarında daha fazla dişli temas yüzeyi oluşur. Bu da daha fazla tork iletimine olanak sağlar. Bu nedenle de aracın manevra kabiliyeti artmış olur. Helisel pinyon dişli aşınmaya dirençli ve kuvvete karşı mukavemeti yüksek 16MnCr5 (SAE 5115) çeliğinden üretilmiştir.



Şekil 10 Helisel Pinyon Dişli

**Helisel Kremayer Dişli:** Helisel pinyon dişlinin y ekseninden gelen dairesel dönme hareketini x ekseninde direksiyona uygulanan döndürme miktarı kadar sağa-sola gidip gelme şeklinde iletlen dişli çeşididir. Rot kolları da kremayerin uçlarına açılan dişlere sıkılarak sabitlenir. Bağlantılı çalıştığı helisel pinyon dişinden iletilen dairesel kuvveti rot kollarına direkt aktaran en önemli direksiyon sistemi elemanlarından biridir. Helisel kremayer dişli, aracımızın lastiklerinin hareket açısı, araç dönüş çapı, aracın direksiyon dönüş açısı, ackerman prensibi vb. hesaplamalar dikkate alınarak tasarlanmıştır. Aynı zamanda kremayerdeki diş sayısını direksiyon sistemimizde her iki yönde de aşırı yönlemeyi engelleyecek şekilde tasarladık. Helisel kremayer dişli aşınmaya dirençli ve kuvvete karşı mukavemeti yüksek 16MnCr5 (SAE 5115) çeliğinden üretilmiştir.



Şekil 11 Helisel Kremayer Dişli

**Dana Gözü:** Kremayer dişlisi ile rot kolu arasında bağlantı elemanı olan dana gözü, bir ucu mile sabit, diğer ucu içi boş küre şeklinde olan salınım ve dairesel dönme hareketlerini kolaylıkla gerçekleştirebilen, rot kollarının salıncak sistemindeki deveboynundan çıkan yönlendirme kollarına rahatlıkla bağlanabilmesini sağlayan ve aynı zamanda yoldan gelen sarsıntıları sönümleyerek rot kollarını koruyan ara bağlantı elemanıdır.



Şekil 12 Dana Gözü

**Rot Kolu:** Bir ucu dana gözüne bağlı diğer ucu rot başına bağlı olan, aracın sağa-sola yönlendirilmesinde önemli yer tutan bu parça direksiyon sağa veya sola çevrildiğinde tekerleklerin de aynı yönde sağa veya sola dönmesini sağlar. Direksiyon kutusunu solda konumlandırdığımız için sağ rot kolu uzun, sol rot kolu kısa seçilerek montajlanmıştır.



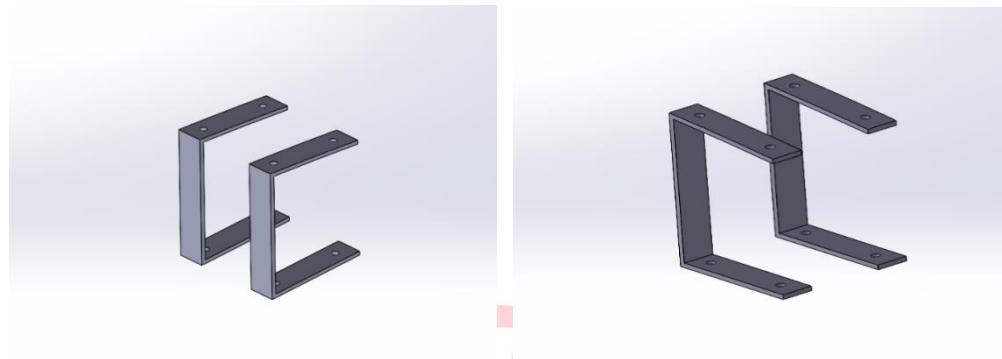
Şekil 13 Rot Kolu

**Rot Başı:** Direksiyon mekanizmasının tekerleğe yön verebilmesi için kullandığı bir dizi parçanın son halkasıdır. Direksiyon kutusundan çıkan yönlendirme hareketini, direksiyon rot kolundan, deveboynundaki yönlendirme koluna oradan da tekerlere iletilmesini sağlayan parçadır. Rot kollarına sıkılıp gevsetilerek toe-in, toe-out açlarını vermemizi sağlar.



Şekil 14 Rot Başı

**Direksiyon Kutusu Sabitleme Ayağı:** Rot kollarımızın şase üzerinde çıkış merkezlerinden dolayı direksiyon kutumuz şaseden yüksekte kalmaktadır. Bu yükseklik farkından dolayı oluşan problemin önüne geçebilmek için direksiyon kutumuzun şaseden yüksekten sabit bir şekilde kalabilmesi amacıyla sabitleme ayakları demir lamadan üretilip montajlanmıştır.



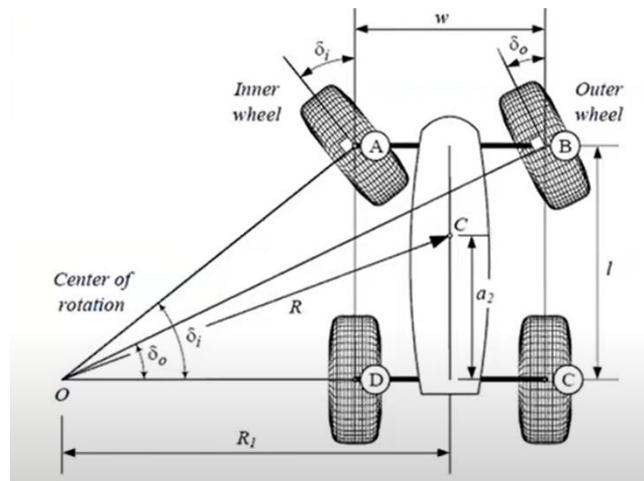
Şekil 15 Direksiyon Kutusu Sabitleme Ayağı

**DC Motor Sabitleme Ayağı:** Direksiyon kutusu ve DC motorun aynı doğrultuda olduğunda dönme kuvvetini en iyi şekilde verdiği düşündük ve buna bağlı olarak direksiyon kutusunu ve DC motorumuzu aynı doğrultuda konumlandırmak için bir sabitleme ayağı demir lamadan üretilip montajlanmıştır.



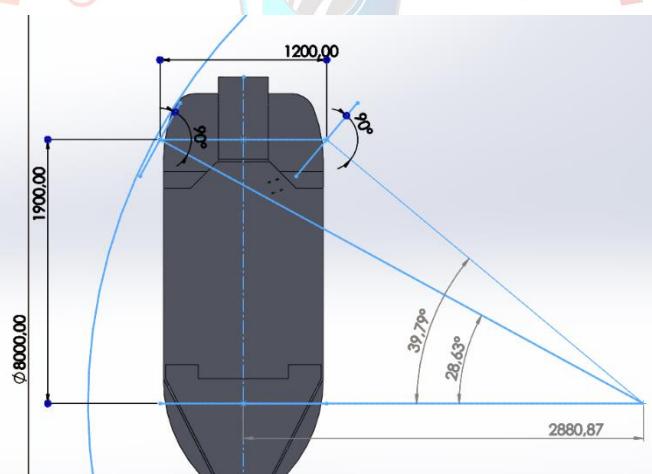
Şekil 16 DC Motor Sabitleme Ayağı

**Ackerman Prensibi:** Aracın dönüşleri iyi bir manevra ile dönmesi için uyguladığımız hesaplar neticesinde ispatlanan yöntemdir. Bu hesaplar ile aracın iç tekerinin dış tekere göre daha fazla açı ile dönmesi sağlanır. Bu prensip uygulanmayan araçlarda dönemeçlerdeki kaza oranı uygulanan araçlara göre daha fazla olduğu gözlemlenmiştir. Tam tur dönüşlerde iç tekerin dış tekere farkı 0-9 derecedir.



Şekil 17 Ackerman Prensibi

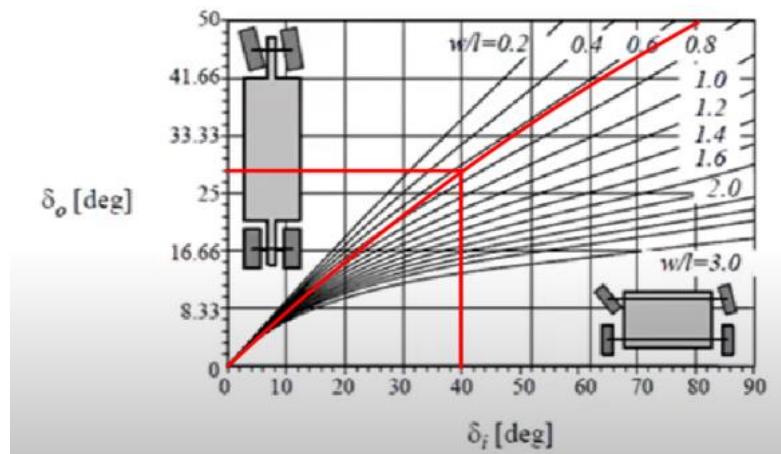
**Ackerman Hesaplamaları:** Aracımızın dönüş yarıçapının 8 metre olmasını istediğimizden W=1200 mm L=1900 mm olduğu için R=2880 mm olarak hesaplanmıştır.



Şekil 18 Ackerman İspatı

OAD ve OBC üçgenlerine göre;

$$\begin{aligned} \tan \delta_o &= \frac{l}{R_1 + \frac{w}{2}} \quad \Rightarrow \quad \tan \delta o = \frac{1900}{2880 + \frac{1200}{2}} \quad \Rightarrow \quad \tan \delta o = 0,54 \quad \Rightarrow \quad \delta o = 28,63 \\ \tan \delta_i &= \frac{l}{R_1 - \frac{w}{2}} \quad \Rightarrow \quad \tan \delta i = \frac{1900}{2880 - \frac{1200}{2}} \quad \Rightarrow \quad \tan \delta i = 0,83 \quad \Rightarrow \quad \delta i = 39,73 \end{aligned}$$



Şekil 19 Araçların  $w/l$  Değerine Göre İç ve Dış Tekerlek Açılarının Belirlenmesi

$R_1$  çekilirse;

$$\begin{aligned} R_1 &= \frac{1}{2}w + \frac{l}{\tan \delta_i} \quad \rightarrow \quad R_1 = \frac{1200}{2} + \frac{1900}{\tan 39,73} \quad \rightarrow \quad R_1 \cong 2880 \\ &= -\frac{1}{2}w + \frac{l}{\tan \delta_o} \quad \rightarrow \quad = -\frac{1200}{2} + \frac{1900}{\tan 28,63} \quad \rightarrow \quad \cong 2880 \end{aligned}$$

$R_1$ 'ler eşitlenirse Ackerman şartı elde edilir

$$\cot \delta_o - \cot \delta_i = \frac{w}{l} \quad \rightarrow \quad \cot 28,63 - \cot 39,73 = \frac{1200}{1900} = 0,63$$

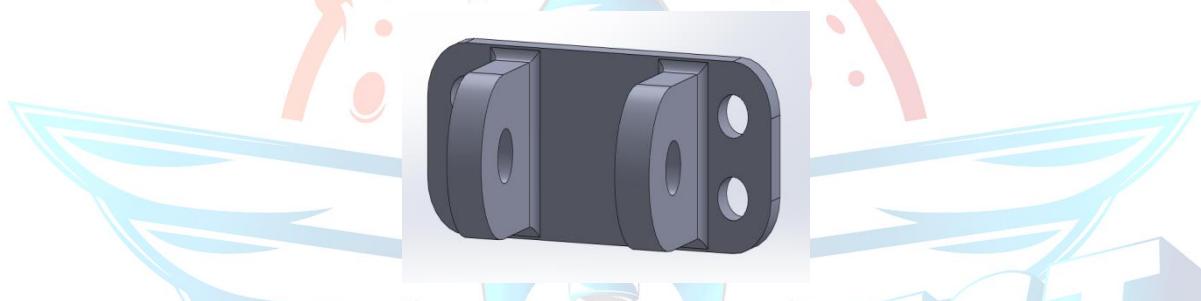
### 3.1.2. Salıncak Sistemi Elemanları ve Analiz Sonuçları



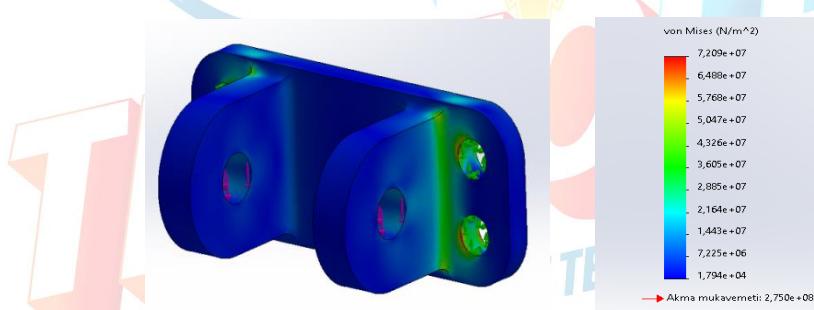
Şekil 20 Salıncak Sistemi

**Kulakçık:** Tasarımını ve üretimini yapmış olduğumuz kulakçık parçaları alüminyum 6061-T6 serisi kullanılarak yapılmıştır. Aracın alt ve üst wishbonunun şase ile olan bağlantısının yapılması için şase ve wishbonelar arasında ara bağlantı elemanı olarak tasarlanıp üretimi yapılmıştır.

- Kulakçık başına gerçek hayatı yaklaşık 12,5kg (125N) yük gelmektedir. Fakat biz analiz sonuçlarının daha garanti olması için 50kg (500N) yayılı yük uyguladık.
- Kulakçılara uygulanan bu analiz sonucunda etkiyen en yüksek gerilme kuvveti  $7,209e+07 \text{ N/m}^2$  dir.
- Kulakçık malzemesi olan 6061-T6 alüminyum合金ının akma mukavemeti  $2,750e+08 \text{ N/m}^2$  dir.
- Malzemenin akma mukavemeti / ulaşılan en yüksek kuvvet ( $27/7$ ) formülü kullanıldığında kulakçıkların emniyet katsayısı 3,85 olarak bulunmaktadır.



Şekil 21 Kulakçık Tasarımı

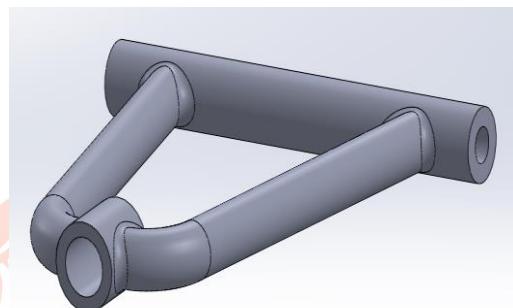


Şekil 22 Kulakçık Analiz Sonuçları

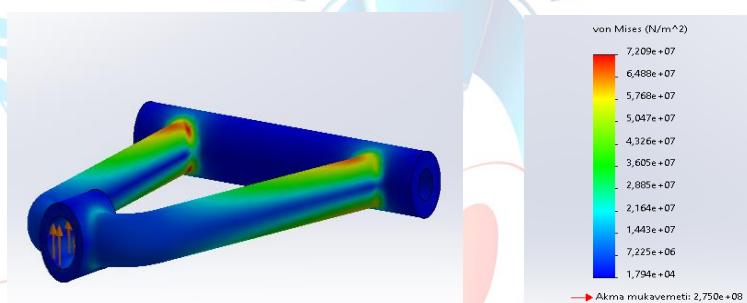
**Üst Wishbone:** Aracımızda bulunan üst wishbone parçası hazır alınarak montajlanmıştır. Üst wishbone aracın deve boynu ile şasesi arasında bağlantı sağlayabilmek ve tekerden gelen titreşimlerin aracın şasesine zarar vermemesi için montajlanmıştır.

- Üst wishbone başına gerçek hayatı yaklaşık 25kg (250N) yük gelmektedir. Fakat analiz yapılırken gerçek hayatı olusabilecek beklenmedik durumlar için 50kg (500N) yayılı yük uygulanmıştır.

- Üst wishbonelara uygulanan bu analiz sonucunda etkiyen en yüksek kuvvet yaklaşık  $2,64e+007 \text{ N/m}^2$ 'dir.
- Üst wishbone malzemesi olan 6061-T6 alüminyum alaşının akma mukavemeti  $2,750e+08 \text{ N/m}^2$ 'dir.
- Malzemenin akma mukavemeti / ulaşılan en yüksek kuvvet ( $275/26,4$ ) formülü kullanıldığında üst wishboneun emniyet katsayısı 10,41 olarak bulunmaktadır.



Şekil 23 Üst Wishbone Tasarımı

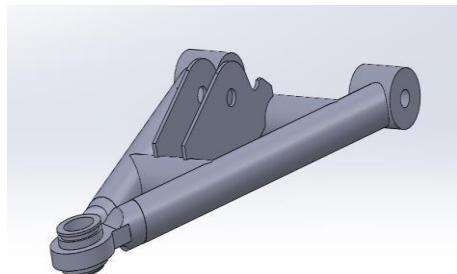


Şekil 24 Üst Wishbone Analiz Sonuçları

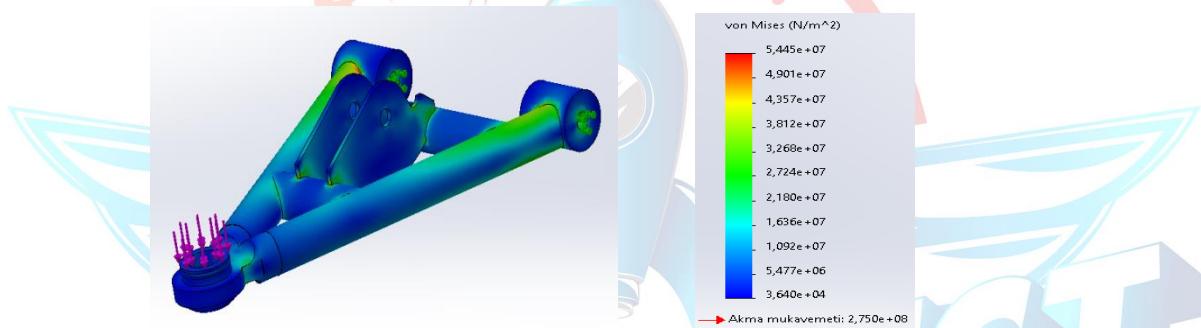
**Alt Wishbone:** Aracımızda bulunan üst wishbone parçası üst wishbone gibi hazır alınarak montajlanmıştır. Alt wishbone üst wishbone gibi titreşimleri engellemenin yanında süspansiyonun bağlantısının şase ve alt wishbone arasında olmasını süspansiyon ile araca olusabilecek titreşimin minimum düzeye indirgenmesi amaçlanarak süspansiyon bağlantısında alt wishbone tercih edilmiştir.

- Alt wishbone başına gerçek hayatı yaklaşık 50kg (500N) yük gelmektedir. Fakat analiz yapılrken gerçek hayatı olusabilecek beklenmedik durumlar için 75kg (750N) yayılı yük uygulanmıştır.
- Alt wishbonea uygulanan bu analiz sonucunda etkiyen en yüksek kuvvet yaklaşık  $5,445e+07 \text{ N/m}^2$ 'dir.

- Alt wishbone malzemesi olan 6061-T6 alüminyum alaşımının akma mukavemeti  $2,750e+08$  N/m<sup>2</sup>'dir.
- Malzemenin akma mukavemeti / ulaşılan en yüksek kuvvet (275/54,45) formülü kullanıldığında alt wishboneun emniyet katsayısı 5,05 olarak bulunmaktadır.

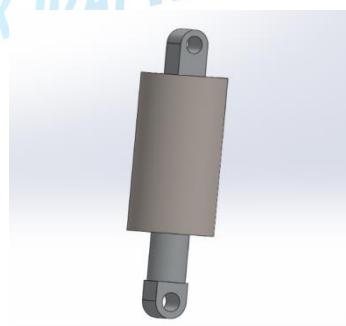


Şekil 25 Alt Wishbone Tasarımı



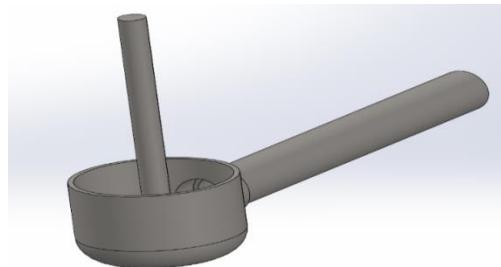
Şekil 26 Alt Wishbone Analiz Sonuçları

**Süspansiyon:** Alt wishbone ile şase arasında köprü oluşturur, tekerin y ekseninde yapacağı yukarı-aşağı hareketlerinin bağlantı elemanlarına zarar vermeden aracın hareket etmesini sağlar. Aynı zamanda yol üzerinden tekere gelen titreşimleri, darbeleri sönümleyerek dengeyi ve aracı korur.



Şekil 27 Süspansiyon Tasarımı

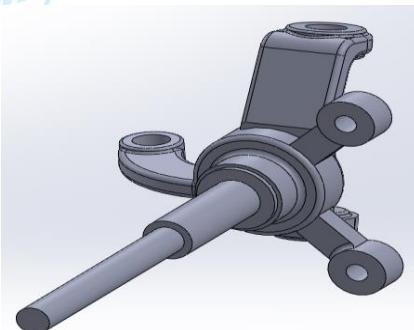
**Rotil:** Wishbonelar ile deveboynunun birbirine bağlanması sağlar, hareket esnasında yoldan lastiğe gelen darbeleri sönümleyerek sistemlerin zarar görmesini engeller, aynı zamanda deveboynunun x ekseninde sağa-sola hareket etmesine olanak sağlar ve tekerlerdeki bazı açıları kamber vs. ayarlamaya yardımcı olur.



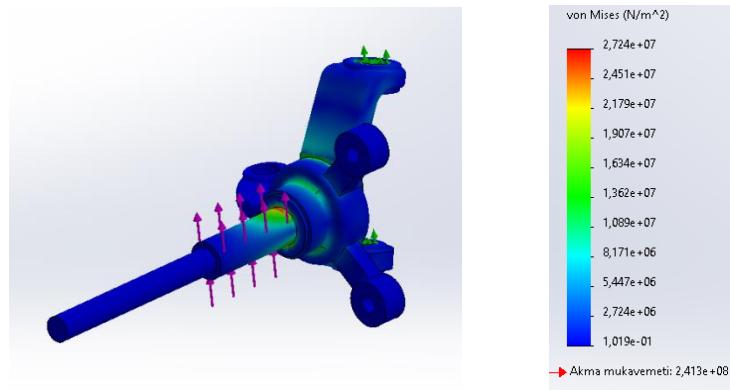
Şekil 28 Rotil Tasarımı

**Deveboynu:** Üst wishbone ve alt wishbone arasında bağlantı kurarak deveboynuna kaynak yöntemiyle montajlanan mil aracılığı ile tekerlerin araca montajlanması sağlayan bağlantı elemanıdır. Direksiyon sisteminden çıkan ve rot kollarında oluşan x eksenindeki git-gel hareketini deveboynundan çıkan yönlendirme kolu yardımıyla tekerleklerde sağa-sola dönme hareketine çevirir. Aynı zamanda fren kaliperinin sabitlenmesi için gerekli parçaya uygun montaj alanı sağlar.

- Deveboynu başına gerçek hayatı yaklaşık 37,5kg (375N) yük gelmektedir. Fakat analiz yapılmışken gerçek hayatı olusabilecek beklenmedik durumlar için 50kg (500N) yayılı yük uygulanmıştır.
- Deveboynuna uygulanan bu analiz sonucunda etkiyen en yüksek kuvvet yaklaşık  $2,724+07 \text{ N/m}^2$ 'dir.
- Deveboynu malzemesi olan dökme demir çelik alaşımının akma mukavemeti  $2,413e+08 \text{ N/m}^2$ 'dir.
- Malzemenin akma mukavemeti / ulaşılan en yüksek kuvvet ( $241,3/27,4$ ) formülü kullanıldığında deveboynunun emniyet katsayıısı 8,80 olarak bulunmaktadır.



Şekil 29 Deveboynu Tasarımı

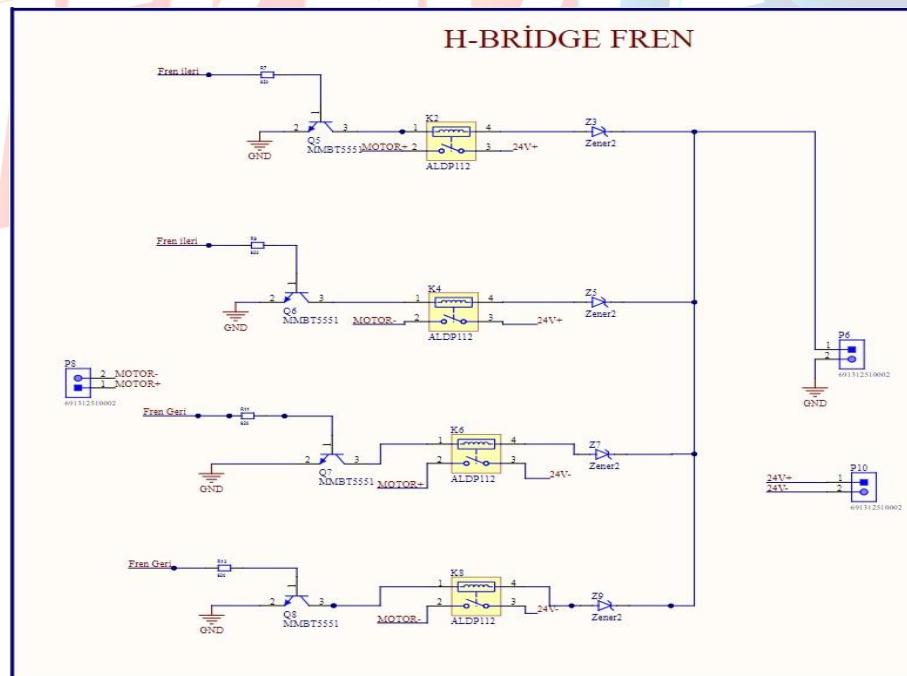


Şekil 30 Deveboynu Analiz Sonuçları

### 3.2. Fren Sistemi

Araçta bulunan fren sistemi hidrolik fren sistemlerinde bulunan fren pedalı yerine lineer motora takılacak olan bağlantı elemanları sayesinde kuvveti iletmesi sonucu fren sisteminin güvenli sürüş için gerekli frenlemeyi yapması planlanmaktadır. Fren sisteminde; fren merkezi, lineer motor, fren diskı, tek pistonlu kaliper, fren hortumları ve fren gücü aktarımı için gerekli kollar bulunmaktadır. Bunun yanı sıra lineer motorun takılacağı ve fren merkezinin sabitleneceği parçaları da kendimiz üretmiş bulunmaktayız.

Elektriksel fren sistemimizi de direksiyon sistemine benzer şekilde araç kontrol kartında roleler kullanılarak H-Bridge devresi tasarlanmıştır. Fren motorunun ileri geri hareketini bu şekilde kontrol ederek gerçekleştirmekteyiz.



Şekil 31 Araç Kontrol Kartında Bulunan Fren Motoru Kontrol Devresi Şematiği



Şekil 32 Montajlanmış Fren Sistemi



Şekil 33 Fren Motoru

Lineer motorun yüksek güvenlik, işlemleri yerine getirmek için basit ve az devre elemanlarından oluşması, uzun mesafeler arasında yüksek hızlı hareket ederken hassasiyete sahip olması ve sık bakım gerektirmemesi gibi avantajları olduğu için aynı zamanda da aracımızda 12V ile çalışan, 1000N yük kapasitesine sahip, hızı 20mm/sn ve 30 cm'ye kadar uzama kapasitesi olan bir lineer motor seçtik.

<b>Çalışma voltajı</b>	12V
<b>Uzunluk</b>	300mm
<b>Akım (Yüksüz)</b>	600mA
<b>Akım (Yük Altında)</b>	850mA
<b>Yük kapasitesi</b>	500n
<b>Hız</b>	20mm/sn
<b>Ağırlık</b>	998gr
<b>Minimum Uzunluk</b>	30 cm
<b>Maksimum Uzunluk</b>	49 cm

Tablo 6 Lineer Motor Teknik Özellikleri

**Fren Hesaplamaları:** Kara Yolları Genel Müdürlüğü'nün sitesindeki tablodan araç hızına göre durma mesafesi belirlenmiştir. Fren hesapları araç 10 km/h hızla giderken durmak istediği bir durum için hesaplanacaktır. Tablodan bakıldığından 10 km/h hızla giden bir aracın toplam 0,7 m gibi bir mesafede tamamen durabilmesi gerekmektedir.

EĞİMSİZ VE KURU ASFALT YOLDA ARAÇLARIN HIZLARINA GÖRE DURMA MESAFELERİ ( f=0,60 )					
ARAÇ HIZI		Reaksiyon Mesafesi	Fren Mesafesi	Durma Mesafesi	Fren Zamanı
km/saat	m/sn	metre	metre	metre	saniye
10	2,77	2,1	0,7	2,8	0,47
15	4,18	3,1	1,5	4,6	0,7

Tablo 7 Araç Hız ve Yol Durumuna Göre Durma Mesafe Tablosu

- **Reaksiyon Süresinin Belirlenmesi:**

Tablodan da görüldüğü gibi 2,1 m gibi bir mesafe sadece reaksiyon mesafesi olarak belirlenmiştir. Bu süre mekanik sistemlerdeki gecikmelerden kaynaklanmaktadır. Yaptığımız Hesap ile bu süre: 0,49 saniye.

- **Fren Süresinin Hesaplanması:**

Araç 10 km/h hızdan 0 km/h hıza x kadarlık bir fren zamanında yavaşlayıp durması gerekmektedir. Bu süreç toplamda 0,7 metrelük bir mesafede gerçekleştiğine göre: Fren süresi: 0,49 saniye.

- **Fren anındaki ivmenin hesabı:**

Fren anında araç 10km/h ten 0 km/h hıza 0,49 saniye de inmektedir. Bu durumda: Araç fren anında  $-5.65 \text{ m/sn}^2$  ivmeye yavaşlamalıdır.

- **Fren Kuvvetinin Hesabı:**

Newton'un 2. Yasası gereği,  $F = m.a$  'dır. (Araç kütlesi 150kg alınırsa) Fren kuvveti  $\Rightarrow F = m. a = 150\text{kg} \times 5.67\text{m/sn}^2 = 850,5\text{N}$  bulunur.

- **Fren Anında Tekerlerdeki Fren Yükünün Dağılımı:**

Sistemde toplamda iki adet fren kullanacağız ve bunlar sadece ön tekerlerde bulunacak. Bu durumda öndeki tekerler 850,5 N fren yükünü karşılayacak.

- **Ön tekerlerdeki fren kuvvetlerinin hesaplanması:**

Ön tekerlerin karşılaşması gereken 850,5N 'luk kuvvet iki tekerlere de eşit dağıldığı varsayılarak, sadece bir tekerin 425,25N 'luk bir fren kuvveti oluşturabilmesi gerekmektedir. Bu kuvvetin oluşturulabilmesi için 220mm olan fren diskimize temas edecek olan balatadaki sürtünme kuvvetinin 1598,9 N olması gerekmektedir. Balata ve fren diskleri arasındaki

sürtünme ise günümüzde 0,3-0,4 arasındadır. (0,35 alınmıştır) Fren diskimize her iki tarafından da balatalar sürecektir. Bu balataların gerekli sürtünme kuvvetini verebilmesi için  $4568,28N/2 = 2284,14N$  'luk bir kuvvetle sıkılması gerekmektedir.

- **Balatalardaki emniyetli yüzey basınclarının bulunması ve balata boyutlarının hesabı:**

Günümüz teknolojisi ile üretilen balatalarda emniyetli yüzey katsayısı maksimum  $800-1200N/cm^2$  dir. Emniyet katsayı: 2,5 alınırsa emniyetli yüzey basıncı  $320 N/cm^2$  olarak alınabilir.  $320N/cm^2$ 'lik yüzey basıncına göre  $4568,28N$ 'luk baskı altında deformasyon olmaması için  $6,86 cm^2$ 'den büyük bir baskı alanı olmak zorundadır. Kısacası fren diskinin sadece bir yüzeyine temas eden alanın  $7 cm^2$ 'den büyük olması gerekmektedir. Araçta kullanmayı planladığımız MONDİAL 125 MH üzerinde kullanılan TL-F305 pabucunun yüzey alanı  $11 cm^2$  cm'dir.

Fren Merkezindeki Pistonu İten Kuvvet:

$$F_s = F_N \cdot \mu$$

$$\mu (\text{fren diskinde oluşan sürtünme}) = 0,35$$

$$1598,9N = F_N \cdot 0,35$$

$$F_N = 4568,28N$$

$$\text{PHİDROLİK} = F_N / A$$

$$A = 13,20 \text{ cm}^2$$

$$\text{PHİDROLİK} = 4568,28N / 1320 \text{ mm}^2 = 3,4608N / \text{mm}^2 = 34,608 \text{ bar}$$

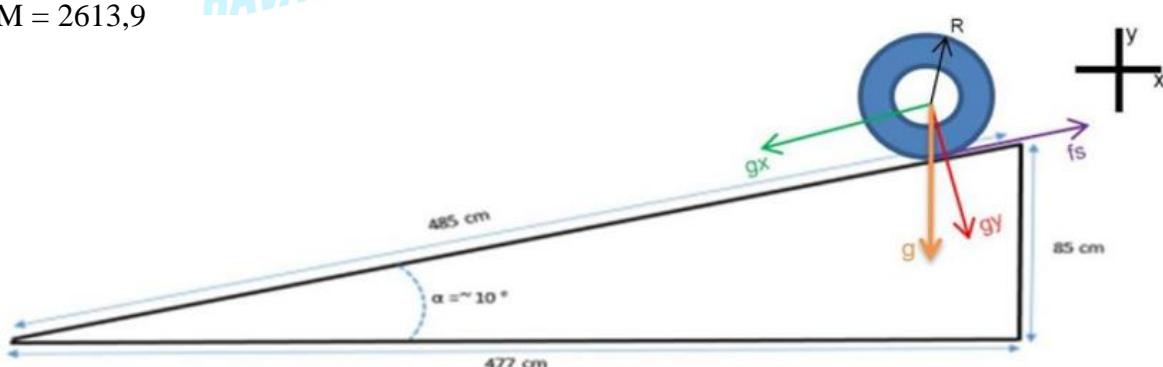
$$\text{PHİDROLİK} = F_M / A_F M (\text{FREN MERKEZİNDEKİ PİSTONU İTEN KUVVET})$$

$$d: \text{Fren Merkezi} = 10\text{mm} \quad r: \text{Fren Merkezi} = 5\text{mm}$$

$$A = \pi r^2 = \pi \cdot (5)^2 = 78,53 \text{ mm}^2$$

$$33,286 \text{ bar} = F_M / 78,53 \text{ mm}^2$$

$$F_M = 2613,9$$



Şekil 34 Frenleme Rampası

“R” harfi tekerleğin yarıçapını, “r” diskin yarıçapını, “g”, “gx” ve “gy” tek tekerlege ve bileşenlerine dağıtılmış ağırlığı,  $F_s$  ise tekerlek ve rampa arasındaki sürtünme kuvvetini simgelemektedir.

Varsayımlar:

Araç ağırlığı 150 kg

Tekerlek ve rampa arasındaki sürtünme katsayısı 0,8

Araç ağırlığı dört tekerlek üzerine homojen dağıtılmıştır.

Kullandığımız fren balataları 0,28 – 0,38 arasında sürtünme katsayısına sahip yarı metalik fren balatalarıdır. Katsayısı 0,35 olarak kabul edildi.

- **Hesaplama:**

Teker başına düşen araç ağırlık Kuvveti:

$$g = (150/4) * 9,81 = 367,875 \text{ N}$$

$$gx = g * \sin(10) = 367,875 * \sin(10) = 63,88 \text{ N}$$

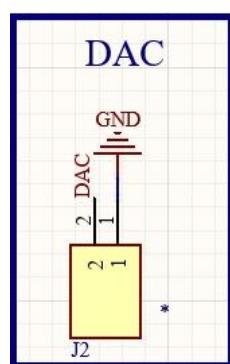
$$gy = g * \cos(10) = 367,875 * \cos(10) = 362,286 \text{ N}$$

$$F_s = gy * (\text{rampa ve tekerlek arasındaki sürtünme katsayısı}) = 362,286 * 0,8 = 289,82 \text{ N}$$

### 3.3 Elektronik Gaz Sistemi

Araç kontrol kartımızda bulunan Arduino UNO ve STM32F103C8T6 mikrodenetleyiciler birbirleri ile haberleşmektedir. Gaz değeri gelen komutlara göre belirlenerek STM32F103' e gönderilir. Mikrodenetleyicinin uygun pini motor sürücüsüne bağlanmıştır. Bu sayede DAC sinyali ile motor sürücüsüne sinyal yollanılarak motorumuzun istediğimiz gaz değerinde çalışması sağlanmaktadır.

PWM sinyali kare dalga ürettiği için motor kontrolü için daha stabil bir yöntem oluçağını düşündüğümüz STM32F103 kullanarak DAC sinyali ile gaz çıkışımızı verdik.



Şekil 35 Araç Kontrol Kartında Bulunan Çıkışın Şematiği

#### 4. Donanım Mimarisi

Çalıştıracağımız otonom sürüs algoritmaları sensörlerimizden gelen verileri bilgisayar sayesinde işleyerek gerekli komutları araç kontrol sistemimize gönderecek. Araç kontrol sistemimiz de aldığı bu komutlar doğrultusunda, direksiyon, fren, motor kontrolünü ve uzaktan haberleşmeyi sağlayacaktır.

Aracımızın donanım mimarisi ile ilgili olan güvenlik önlemleri geliştirilmiştir. Aracımızın üzerinde dışarıdan müdahale edebileceğimiz şekilde 1 adet fiziksel buton bulunmaktadır. Bu buton aracımızın ana elektrik hattına seri olarak bağlanarak tüm hattı kesebilmektedir. Ayrıca aracın önüne ani bir engel çıkması durumunda TF Mini Lidar'dan alınan bilgilerden faydalanaacaktır. Lidardan alınan bilgiler incelendiğinde aracın önünde belirli mesafeden yakın bir nesne tespit edildiğinde araç acil olarak durdurulacaktır. Batarya kutusu aracın tabanında sağlam bir noktaya yerlerinden oynamayacak şekilde sabitlenecektir. Batarya sıcaklığı belirli bir sıcaklık seviyesinin üzerine çıktığında buzzer ötecektir. Aracımızın tesisatında kullanılan çeviricilerimiz elektriksel izolasyondur. Araç üzerinde bulunan elektrik kabloları üzerinden geçeceği akıma göre seçilmiştir. Elektrik tesisatımız, her hattı farklı renk kablo ile yapılmıştır. Aracımızın şerit takibi algoritmasında herhangi bir şerit görmemesi durumunda hızını yavaşlatan ve buna göre eylem gerçekleştiren bir güvenlik görevi oluşturulmuştur.

##### 4.1. Yapay Zeka Bilgisayarı

Yapay zekâ bilgisayarı olarak NVIDIA GeForce RTX 3070 grafik kartlı bir bilgisayar tercih etti. Bu grafik kartını seçerken RTX 2070 ve RTX 3070 arasında kaldık. Sonra hangisinin derin öğrenme ve otonom sürüs algoritmalarımız için en iyisi olduğuna karar vermek adına GPU mimarilerini araştırdık. RTX 3070 grafik kartında Ampere mimarisi kullanılmakta olup RTX 2070 grafik kartında ise Turing mimarisi kullanılmaktadır. Ampere, GDDR6'dan (Turing'de olan) çok daha hızlı bir VRAM olan GDDR6X'i kullanmaktadır. Aynı zamanda CUDA çekirdeği de bilgisayarımız derin öğrenmede iken en iyi şekilde sonuç alabilmek için önemlidir. RTX 3070 grafik kartında 5,888 tane CUDA çekirdeği var iken RTX 2070 de ise bu sayı 4,352' ye düşmektedir. Bu özelliklerinden dolayı RTX 3070 grafik kartlı bir bilgisayar almaya karar verdik.

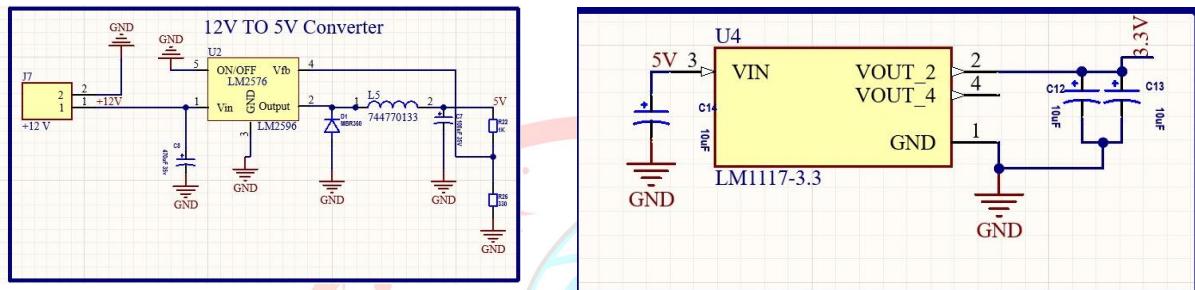
Teknik özellikler	ASUS ROG STRIX G5I3QR
Grafik işlemcisi	NVIDIA GeForce RTX3070 8 GB GDDR6X
İşlemci	AMD RYZEN 9 5900HX
Bellek	16 GB DDR4
Depolama	1 TB NVMe M.2 (PCIe) SSD

Tablo 8 Dizüstü Bilgisayar Teknik Özellikleri

## 4.2. Araç Kontrol Kartı

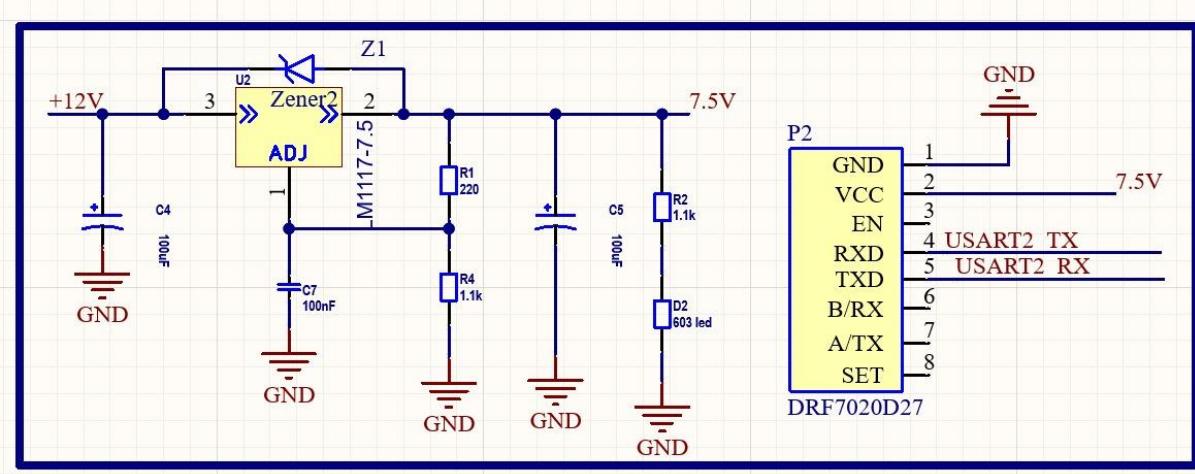
Ekibimiz tarafından tasarlanan araç kontrol kartı ile arabamızın beyni olan bilgisayar seri haberleşerek gerekli algılama, planlama ve kontrol işlemlerini eş zamanlı gerçekleştirecektir.

Kartımız 12 v ile beslenmektedir. Daha sonra Arduino UNO ve STM32F103 kartlarımızın 5v-3.3v ihtiyacı için LM2596 ve LM1117-3.3 devresi kurulmuştur.



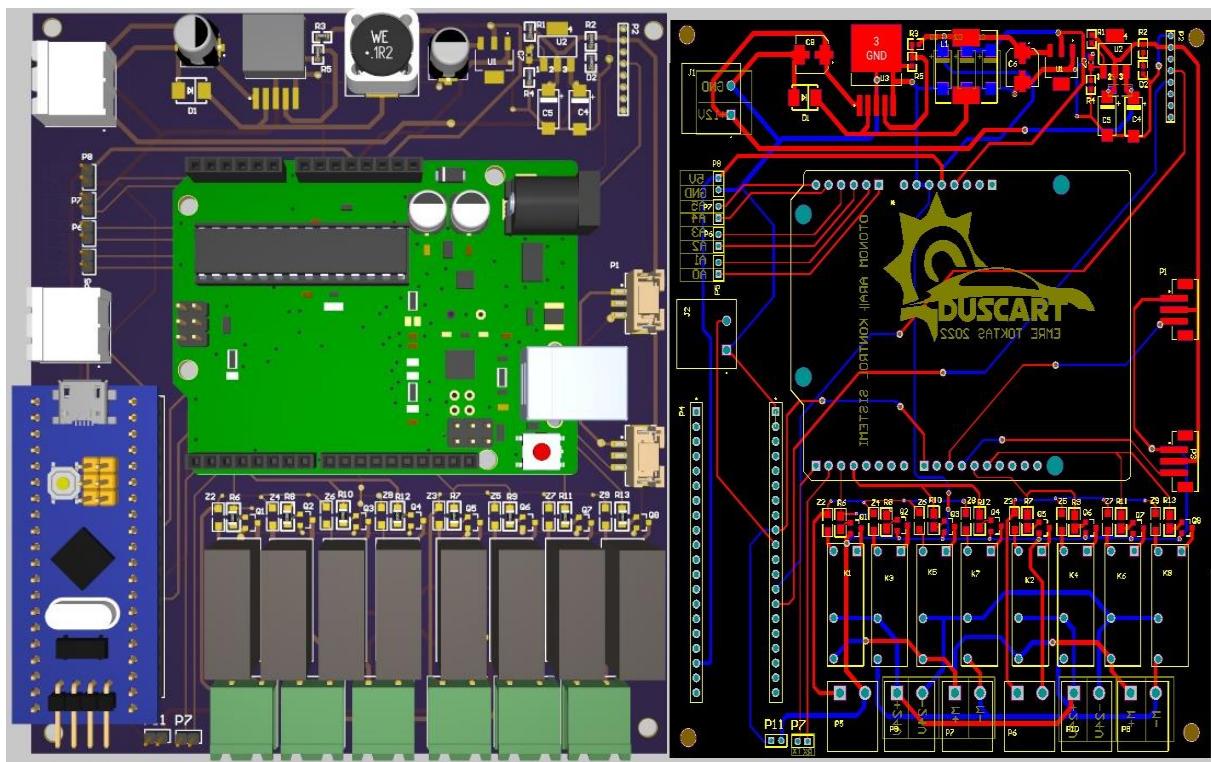
Şekil 36 LM2596 ve LM1117-3.3 Devresi Şematiği

Aracımızın uzaktan haberleşmesi için DRF7020D27 dörtlü sistemi kurulmuştur. Bunun için yine aynı şekilde 12 voltta 7.5 volta dönüştürücü LM1117-adj entegresi ile 7.5 dönüştürücü devresi kurulmuştur.

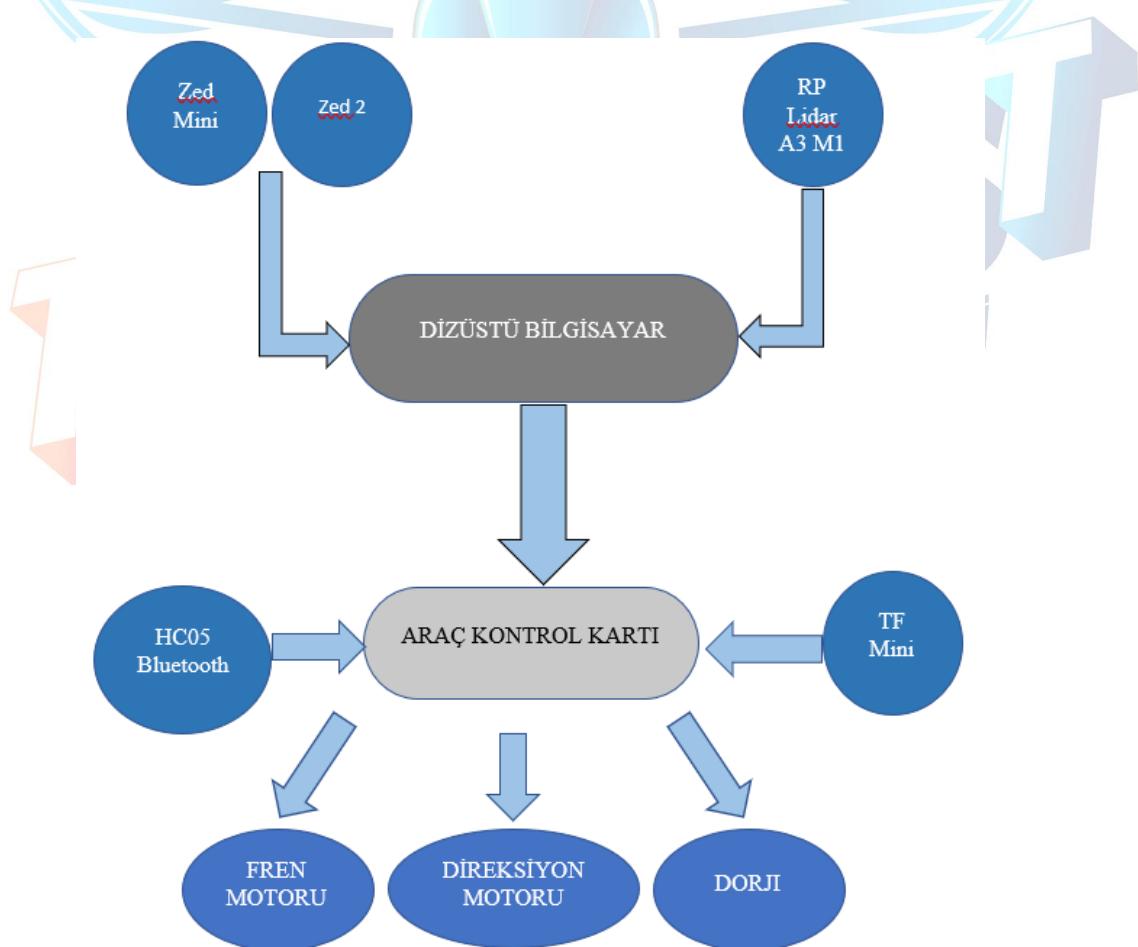


Şekil 37 LM1117-adj Devresi Şematiği

Araç kontrol sistemi kartımızda Arduino ve STM32 UART protokolü ile haberleştirilmektedir. Arduinodan gelen bilgiler doğrultusunda STM32F103 kartımız tarafından motorumuz DAC sinyali gönderilmektedir. Bu sinyal doğrultusunda aracımızın kontrolü sağlanmaktadır. Ayrıca kartımızda ihtiyaç olması halinde kullanılmak üzere analog pinler karttan dışarıya bırakılmıştır.



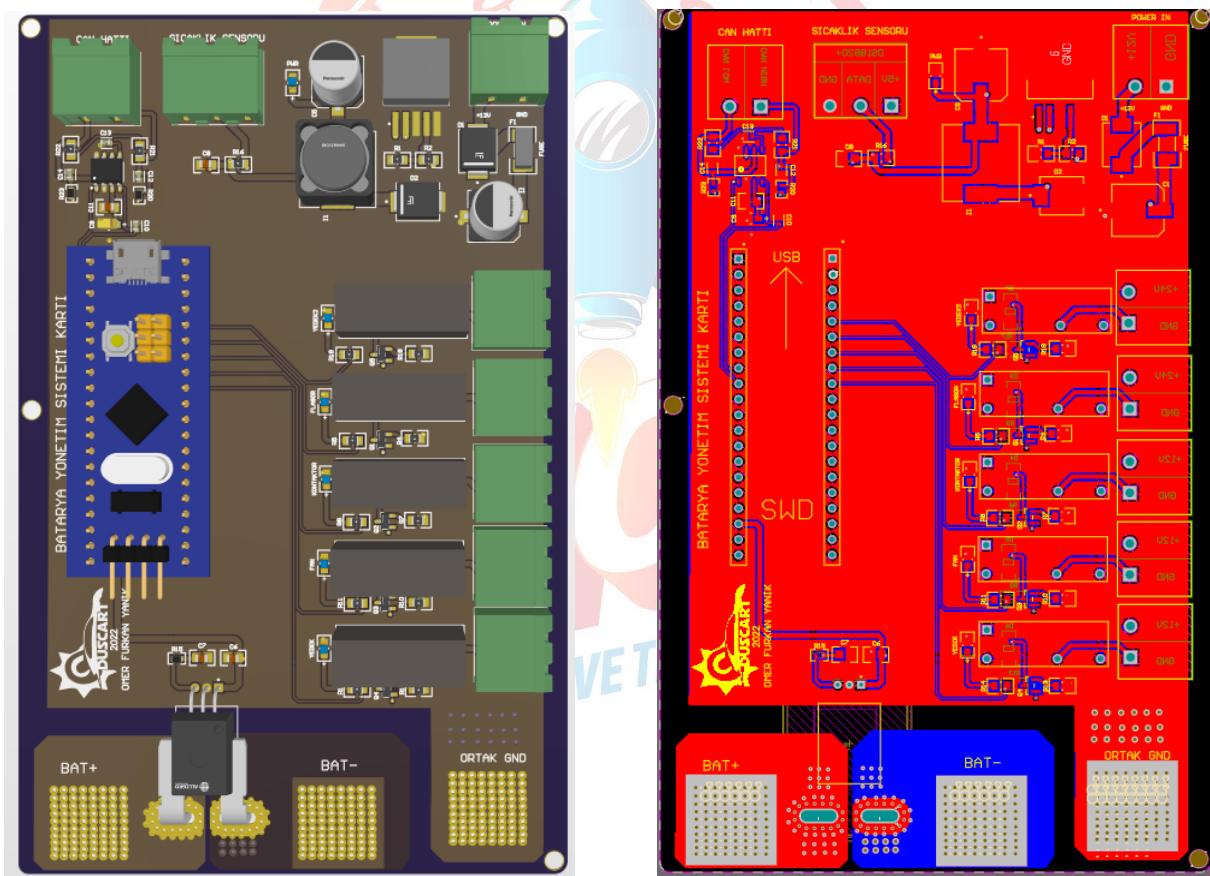
Şekil 38 Araç Kontrol Kartı PCB Tasarımı



Şekil 39 Donanım Mimarisi

### 4.3. Batarya Yönetim Sistemi

Batarya yönetim sistemi kartımız temel olarak 5 fonksiyonu yerine getirecektir. Bunlar hücre sıcaklıklarları, akım , state of charge , bataryanın üstünde bulunan fan kontrolü , kontaktör kontrolü ve flaşör kontrolüdür. Batarya yönetim sistemi kartımız tak çıkarlı olacaktır üzerinde bulunan STM32F103C6 geliştirme kartı ile kontrol edilecektir. Kartın üzerinde bulunan sıcaklık sensörü anlık batarya hücrelerinin sıcaklığını ölçeceğ olup anormal bir sıcaklık yükselişinde ilk aşamada fani açıp bataryayı soğutacaktır ayrıca flaşörü de açıp çevredekileri sesli ve görüntülü bir şekilde uyaracaktır eğer ki sıcaklık kritik çalışma sıcaklığını aşar ise kontaktörü tetikleyip sistemi kapatacaktır. Kart üzerindeki akım sensörümüz ise anlık akımı ölçümü alacaktır bu ölçüm neticesinde coulomb sayma metodu ile batarya yüzdesini hesaplayacaktır bunun yanı sıra anormal derecede akım çekilirse güvenlik önlemi olarak kontaktörü tetikleyip sistemi kapatacaktır.



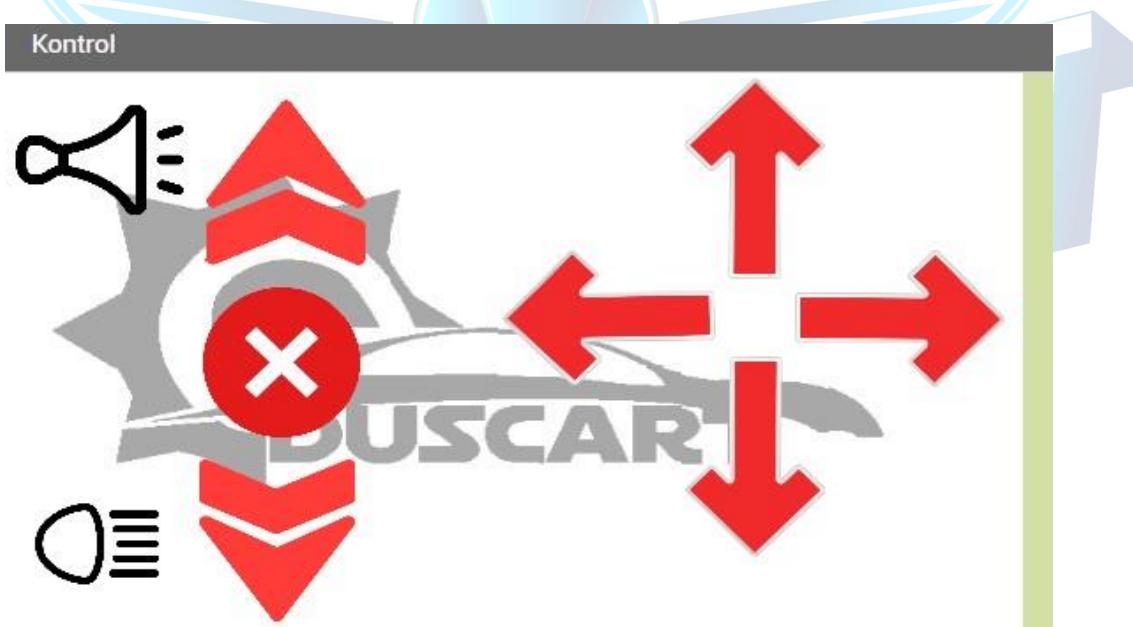
Şekil 40 BYS Kartı PCB Tasarımı

### 4.4. Kablosuz Haberleşme Sistemi

Haberleşme için 433 MHz frekans bandı ve 500 mW RF çıkış gücüne sahip DRF7020D27 DORJI alıcı verici RF haberleşme modülü kullanmayı planlamıştık. Ancak yaptığımız testler sonucunda yakın mesafelerde araca gönderdiğimiz komut mesajlarında karışıklık yaşandığını

fark ettiğimiz araç kontrol kartına yerleştirdiğimiz modül komut mesajı almayacak ancak anlık verileri izleme bilgisayarımıza aktarmak için sistemimizde yine dahil olacaktır.

Karşılaştığımız bu probleme çözüm olarak kontrol kartımıza HC05 bluetooth modülü dahil ettiğim. Aynı zamanda sade bir arayüz tasarımları kullanarak bir mobil uygulama geliştirdik. Programladığımız mobil uygulama sayesinde araç üzerindeki bluetooth modülü ile haberleşmeye sağladık. Gönderdiğimiz komutlara göre uygun kontrol elemanlarının çalışmasıyla aracımızın hareketini sağladık.



#### 4.5. Sensörler

Sensörler, kapsama alanlarındaki olayları veya değişiklikleri tespit etmek ve bilgileri diğer elektronik cihazlara iletmek için kullanılırlar.

Tasarladığımız otonom araç için en önemli veri girdisi kamera ve lidar sensörleri ile

sağlanmaktadır. Bu sensörler sayesinde araç çevresini tanıayıp doğru kurallarla doğru yollardan hareketini sağlayabilmektedir.

#### 4.5.1. Lidar

LIDAR, uzaktaki nesneleri algılamak için odaklanmış ışık demetlerini kullanan sensördür. Sinyali analiz etmek ve anlamlı veriler üretmek için güçlü odaklanmış bir ışık kaynağı, hassas ve hızlı bir alıcı ve hızlı bir işlemci gerektirir. Çalışma prensibi olarak da oldukça basittir. Bir ışık demeti göndererek ve yansyan sinyali ölçerek çalışır. Odaklanmış ışık demeti bir nesneye yönlendirilir ve sensör onun yansımmasını arar. İşin algılanırsa yoğunluğu ve açısı (veya fazı) ölçülür. Bu ölçümler, yansıtıcı nesneye olan mesafenin yanı sıra yansıtıcılığına ilişkin bazı bilgileri hesaplamak için kullanılabilir. Bu değerler daha sonra, yansyan nesnelerin konumunu ve özelliklerini belirlemek için hızlı bir yerleşik bilgisayar tarafından çalıştırılan bir denkleme eklenir. Aracımızda ön tasarım raporunda belirttiğimiz lidarlardan RPLIDAR A3 M1' i kullanmaya karar verdik.



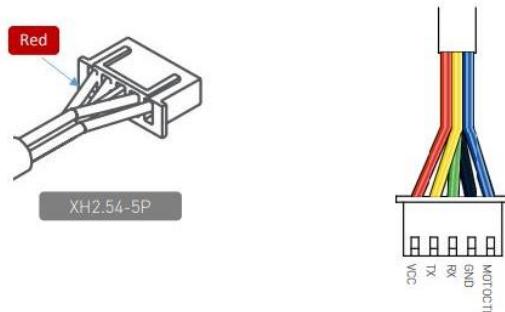
Şekil 43 RPLIDAR A3 M1

RPLIDAR A3M1, 0,5-25 görüş mesafesine sahiptir. Harita oluşturma, yer işaretleme ve nesne/ortam modelleme gibi alanlarda kullanılarak kesin sonuç verir. Mükemmel performansa sahip olmasını sağlayan düşük maliyetli üçgenleme ölçüm sistemini benimser. RPLIDAR A3 M1, varsayılan ayarında 10Hz'de (600 Rpm) çalışır. Bu frekansta çözünürlük 0,9° olur. Tarama frekansını arayüz ile 5-15 Hertz arasında ayarlayabilirsiniz. Lazer çekirdeği maksimum 16000 Rpm hızla döner. Bu saniyede 16000 örnek oluşturabilmesi anlamına gelir.

RPLIDAR A3 M1, bir lazer tarama çekirdeği ve çekirdeğin yüksek hızda dönmesini sağlayan mekanik aksamdan oluşur. Ürün çalışmaya başladığında lazer saat yönünde dönmeye ve taramaya başlayacaktır. Cihazın arayüzü ile Start, Stop ve PWM üzerinden Lazer Dönme Hızı

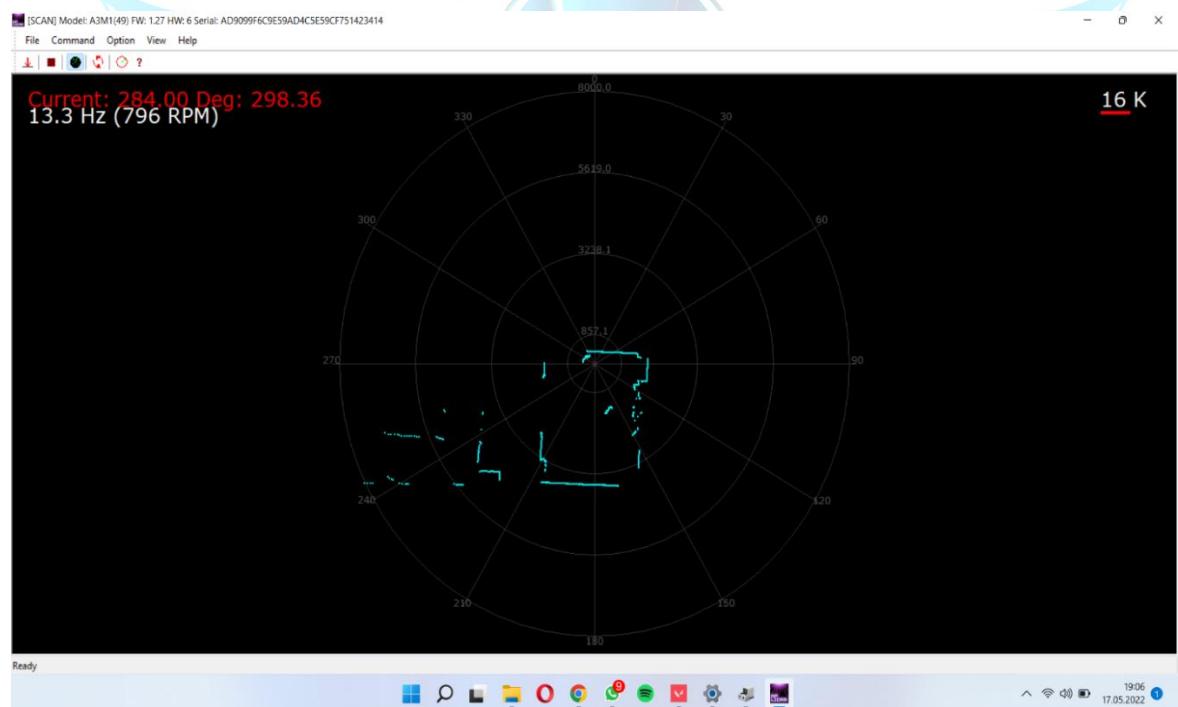
kontrolleri yapılabilir. Cihaz dönüş hızı algılayıcısı ve bağlantı aparatlarıyla birlikte gelir. Kurulumu ve çalıştırması oldukça kolaydır.

Ürün, lazer çekirdeği ve motor sistemi için DC 5V'a gereksinim duyar. Standart XH2.54-5P erkek soket kullanır. Detaylı arayüz tanımı aşağıdaki şekilde gösterilmiştir.



Şekil 44 Lidar Bağlantı Arayüzü

RP lidarımız ile sürekli olarak görüntü taradık ve taradığımız noktaların dinamik bir haritasını görüntüledik.



Şekil 45 Lidardan Alınan Gerçek Zamanlı Veri

## RPLIDAR A3M1- 360 Derece Lazer Tarayıcı Teknik Özellikleri

- Mesafe Aralığı: 0.15- 25 m
- Açısal Aralığı: 0-360 derece
- Mesafe Çözünürlük: <0.5 (0.15 ~ 1,5 metre)
- Mesafenin <% 1'i (Tüm mesafe aralığı)
- Açısal Çözünürlük: 0.9 derece
- Örnekleme süresi: 0.25 milisaniye
- Örnek Frekansı:  $\geq 4000\text{Hz}$
- Tarama Hızı: 10Hz
- Ağırlık: 430g

Aracımızda RP Lidar A3 M1' i desteklemek amacıyla TF Mini Lidar sensörünü de kullanacağız.

TF Mini LiDAR (ToF) Lazer Menzil Sensörü ürünü ToF(time of flight-uçuş zaman aralığı) prensibi ile çalışan mesafe sensörlerindendir. Lazer sensörü, engellere olan mesafeyi tespit etmek için bir mesafe ölçüm aracı veya bir robotun gözü olarak kullanılabilir. Sensör güvenli kaçınma ve rota seçimi için mesafe bilgileri sağlar. Çeşitli makine kontrol senaryolarında sistem otomasyonunu destekler.

ToF Sensörleri diğer mesafe ölçüm sensörlerinden farklı çalışır. Mesafe Ölçüm Sensörleri bir IR gönderici ve IR alıcıdan oluşur. Göndericinin gönderdiği IR ışınları engele çarpıp alıcıya yansır. Alıcıya gelen ışının açısına göre mesafe belirlenir.

ToF yani uçuş zamanı sensörleri ise fotonların havada geçirdiği zamanı hesaplayarak mesafeyi ölçer. Lazer ışığı göndericiden çıkışip alıcıya gelene kadar fotonların ne kadar süre boyunca uçtularını ölçer ve aradaki mesafeyi hesaplar. Böylece diğer sensörlerle göre daha kesin sonuç verir.

TF Mini'nin maksimum algılama mesafesi 12 metredir, 6 metrede 100Hz örneklemeye çözünürlüğünü destekler. Sinyal, parazitlerden etkilenmez ve dış ortamlarda kullanım için uygundur.

Aracımızda aracın ön kaputun ortasında, sağ çaprazında ve sol çaprazında olmak üzere 3 adet TF Mini Lidarları RP Lidar A3 M1' i desteklemek amacıyla kullanacağız. Kullandığımız 3 adet TF Mini Lidar ile engellere olan mesafimizi tespit edip en doğru sonuca ulaşmayı hedefliyoruz. Mini lidarlarımız görüş açılarındaki engellerin mesafelerini tespit edecek ve ilgili komut aracımızın motor ve direksiyon sistemine iletilecektir.



Şekil 46 TF Mini LiDAR

Ardunio üzerinden TFmini lidarımızla mesafe verilerini anlık ve doğru bir şekilde aldık. Bu verilerle aracımızdaki ana lidarımızla verileri kıyaslayarak mesafe bilgileri ile aracımızın otonom sürüşünü destekledik.

COM4

```

228cm strength: 6274
228cm strength: 6273
229cm strength: 6281
228cm strength: 6282
228cm strength: 6273
229cm strength: 6251
229cm strength: 6256
229cm strength: 6253
229cm strength: 6291
229cm strength: 6303
229cm strength: 6296
229cm strength: 6284
229cm strength: 6286
229cm strength: 6296

```

Otomatik Kaydırma  Zaman damgasını göster Yeni Satır 115200 baud Çıkışı temizle

Şekil 47 Sensörden Alınan Gerçek Zamanlı Veri

## TF Mini LiDAR (ToF) Lazer Menzil Sensörü Teknik Özellikleri

- Voltaj Aralığı: 4.5 V-6 V
- İletişim Arayüzü: UART (TTL)
- Çalışma aralığı: 0.3m-12m (kapalı)
- Maksimum Yansıtma Aralığı: % 10- 5m
- Ortalama güç: 0.6w
- Kabul açısı: 2,3°
- Minimum Çözünürlük: 5mm
- Yenileme Frekansı: 100Hz
- Değişen Doğruluk: % 1 (<6m), % 2 (6 ~ 12m)
- Değişen birim: mm
- Band: 850nm
- Boyutu: 42mm x15mm x16mm
- Çalışma Sıcaklığı: 0 ° C / 60 ° C
- Depolama Sıcaklığı: -20 ° C / 75 ° C
- Anti-ışık Ortam: 70Klux
- Ağırlık: 4.7g

### 4.5.2. Kamera

Araç üzerinde nesne tespitinde ZED 2 şerit tespitinde ise ZED Mini olmak üzere 2 kamera kullanılmasına karar verilmiştir. ZED 2 ve ZED Mini araç üzerinde konumlandırılması planlanmaktadır. ZED 2 110° lik bir görüntü elde ederek trafik tabelalarını tanıma ve trafik işaretlerinin tanınması hedeflenmektedir. Yolov4 ile nesne algılama ve mesafe ölçümü yapılacaktır. Kameradan gerçek zamanlı nesne ile araç arası milimetre cinsinden mesafe verisi işlenecektir. ZED ile aynı sensör teknolojisi üzerine inşa edilen ZED Mini, dual yüksek hızlı 2K görüntü sensörlerine ve 110 derecelik görüş alanına (FoV) sahiptir. 63 mm' lik gözler arası uzaklık ile, kamera 0,1 metreden 12 metreye kadar derinlik algılama gerçekleştirmektedir. Yeni nesil görsel ataletsel odometri teknolojisini kullanarak, atalet ölçümleri 800Hz' de stereo kameradan gelen görsel verilerle birleştirilir. ZED Mini, zaman içinde algılama kalitesini koruyan ve fabrika kalibrasyonunu koruyan ışiya ve darbeye dayanıklı alüminyum çerçeveli güzel bir siyah tasarım sunar. Yeni bir USB C Tipi bağlantı noktası, video çıkışını ve gücünü tek bir konektöre entegre ederek 3Gb / s'ye kadar veriyi en düşük gecikmede sunar. ZED Mini, mevcut SDK, ZED için mevcut eklentiler ve örneklerle sorunsuz bir şekilde çalışır. ROS ve OpenCV' den Unity ve Unreal' a kadar, robot ve insan analizlerinden karma gerçekliğe kadar heyecan verici uygulamalar oluşturmak için ZED Mini ile çok çeşitli üçüncü taraf kütüphaneleri kullanılabilmektedir. Gerçek zamanlı ve yüksek çözünürlüklü derinlik haritalama ile sanal objeler gerçek objeler tarafından bloklama özelliği ile de ZED 2 ile özellikleri bakımından birlikte uyumlu oldukları için entegrasyonları da kolaydır.



•Şekil 48 ZED Mini

ZED dünyanın en hızlı derinlik kamerasıdır. 30FPS'de 1080p HD video veya 100FPS'de WVGA çekilebilir. Net ve temiz bir görüntü elde edilir. Ultra keskin 6 elementli tüm cam merceklerle 110 ° geniş açı video ve derinlik yakalanabilir. Işığa duyarlı bir yüzey üzerinde görüntü yakalamasına izin veren küçük bir açılığa sahip kapalı kutulardır. Işığın nasıl düştüğünü kontrol etmek için çeşitli mekanizmaları vardır.

İki kameranın 16 x 9 geniş ekran formatında 4.416 x 1.242 piksel sensörü var. ZED'in sensörleri 120 mm (4,7 inç) aralıklıdır, bu da 1,5 ile 20 m arasında stereo derinlik bilgisi sağlar.



**HAVACILIK, UZAY VE İNŞAAT**  
Şekil 49 ZED 2

Lensinden sensörlerine en son teknoloji ile üretilen ve bu şekilde derinlik algılama ve hareket takibini yeni bir boyuta taşıyan ZED stereo kamerasının diğer özellikleri şöyledir:

- Video Çıkışı:

Video Modu	Saniyede çerçeve oranı (fps)	Çıkış çözünürlüğü
2.2K	15	4416x1242
1080p	30	3840x1080
720p	60	2560x720
WVGA	100	1344x376

- Hareket Algılama: Poz güncelleme oranı: 100 Hz'e kadar
- Teknoloji: Gerçek zamanlı derinlik tabanlı görsel odometri ve SLAM
- Lens tipi: 6 elemanlı tamamen cam çift lens ( $90^\circ$  (H) x  $60^\circ$  (V) x  $100^\circ$  (D) max -- FOV)
- İmaj Sensörü: Büyük 2 mikron piksele sahip dual 4M piksel sensörler(Daha geniş bir yatay görüş alanı için 16: 9 formatı)
- Bağlantı: 1,5 m tümleşik kablolu USB 3.0 bağlantı noktası ve USB 5V / 380mA ile güç bağlantısı
- Boyutlar ve Ağırlık :  $175 \times 30 \times 33$  mm ( $6.89 \times 1.18 \times 1.3''$ ) - 135g
- SDK için Sistem Gereksinimleri:
  - o Çift çekirdek 2.3GHz veya üzeri işlemci
  - o 4 GB veya üzeri RAM
  - o USB 3.0
- **Kamera kalibrasyonu**

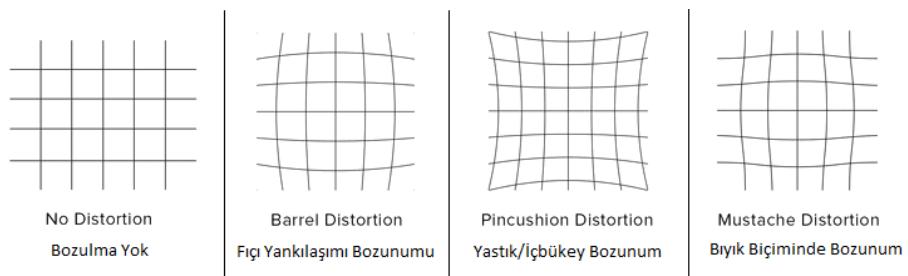
Görüntü işlemini etkileyen kamera parametrelerinin bulunmasını sağlayan işlemidir.

Kalibrasyon sonucunda elde edilen önemli parametreler şunlardır:

Dış parametreler → Kameranın konumu, kameranın açısı

İç parametreler → Odak uzaklığı, optik merkez, lens bozulma katsayısı

Radyal görüntü bozuklukları dış bükey bozulma ve iç bükey bozulma olarak ikiye ayrılır.  
Nedenleri kamera, lens, açı, ışık gibi sebeplerdir.



Şekil 50 Radyal Görüntü Bozuklukları



Şekil 51 Fiçı Bozulumu Örneği



Şekil 52 Yastık Bozulumu Örneği

Kamera kalibrasyonu işlemi, kameranın iç yöneltme parametrelerinin elde edilmesi, görüntü bozukluğunu düzeltmesi için gerekli bazı verilere ulaşmak için yapılır. Kamera kalibrasyonu, bozuk görüntü kaynağı yani kamera üzerinde uygulanır ve bu kalibrasyon verileri kameraya özeldir yani hangi kameradaki görüntü düzeltmek isteniyorsa kalibrasyon işlemi de o kamera üzerinde uygulanmalıdır.

➤ Kalibrasyon tahtası hazırlama:

- Kalibrasyon tahtası çıktıtı düz bir düzeye yapıştırılır.
- Kalibrasyon tahtası karesinin bir kenarını ölçülür.
- Kalibrasyon tahtası ile çekilen 10-20 adet fotoğraf kullanılır.
- PNG gibi kayıpsız formatlardaki görüntüler kullanılır.
- Kalibrasyon tahtasını kamera düzlemine göre 45 dereceden daha az bir açıyla yerleştirilir.

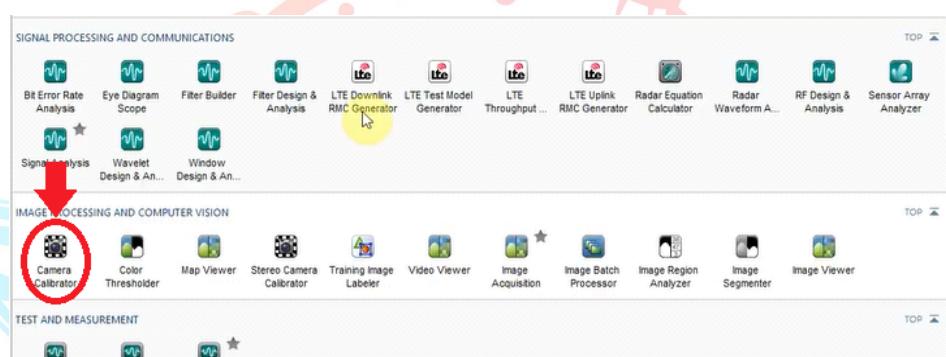
Kalibrasyon tahtasını oluşturduktan sonra kalibrasyonu gerçekleştirmek için matlab uygulaması kullanılmaktadır. Matlab'ın tercih edilme nedeni programda bilgisayarla görmek

İçin kullanılan kamera kalibrasyon uygulaması bulunmasıdır. Bu uygulamada test alanı kalibrasyonu gerçekleştirilmektedir. Test alanı olarak da dama tahtası görüntüsü kullanılmaktadır. Matlab’da kalibrasyon yapma aşamaları şöyledir:

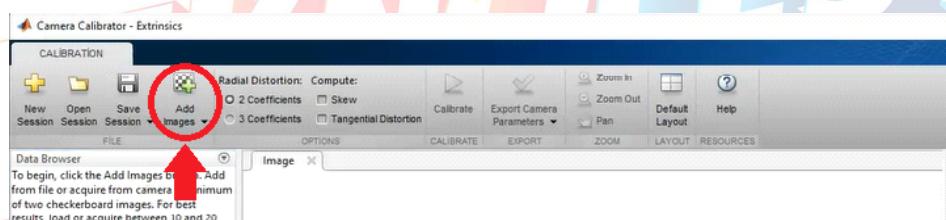
Sayfanın yukarısındaki sekmelelerden “apps” seçilir. Sekmelerde bulunan fotoğrafta da gösterildiği üzere kırmızı daire içerisindeki oka tıklanır.



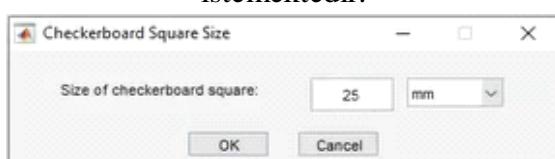
Açılan pencereden "Camera Calibrator"(kırmızı daire içerisindeki) uygulaması seçilir.



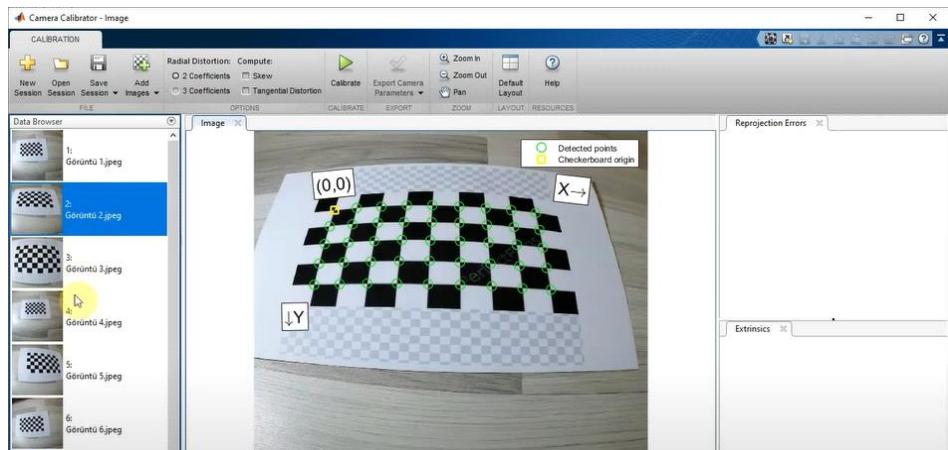
Uygulama açıldığında açılan pencereden "Add Image" seçip kullanılacak dama görselleri seçilir.



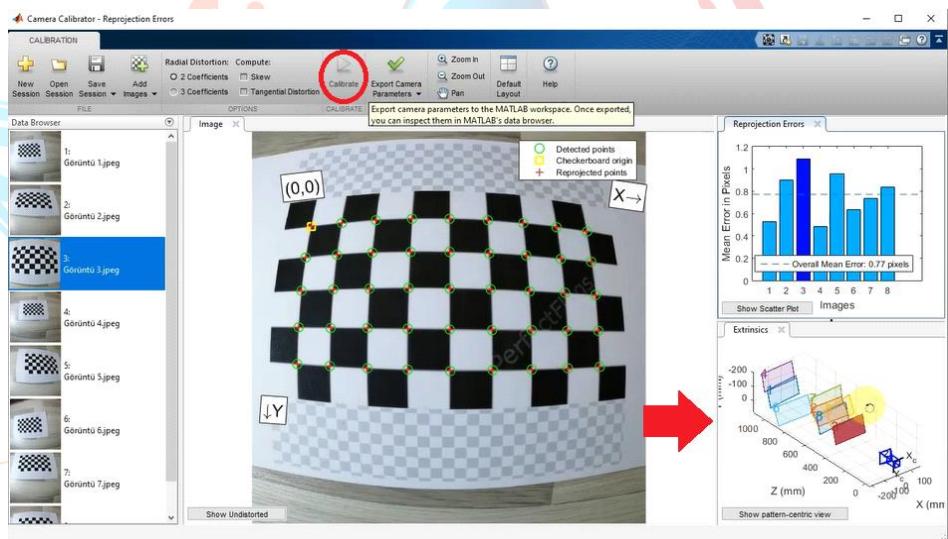
Ardından gelen küçük pencerede her bir dama tahtasının kaç milimetre olduğunu girmemizi istemektedir.



Resimde görüldüğü üzere uygulama tarafından dama tahtası köşeleri bu şekilde belirlenir.



Sonra resimde kırmızı daire içinde gösterilen "Calibrate" butonuna tıklanır ve işlem bittikten sonra okla gösterilen kamera merkezli olan, görüntülerin hangi açıda olduğunu gösteren uygulama tarafından yapılan bir şema açılır.



Son aşamada ise "Export Camera Parameters" butonuna tıklanarak Matlab'a geçilir ve matris hesaplarının yapıldığı sayfaya geçilir ve kamera kalibrasyon işlemi tamamlanmış olur.

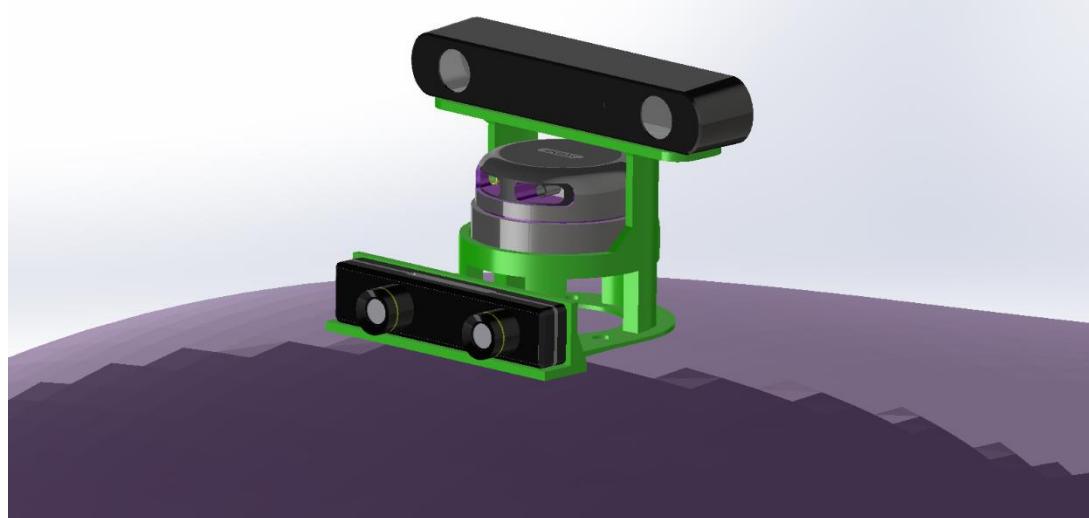
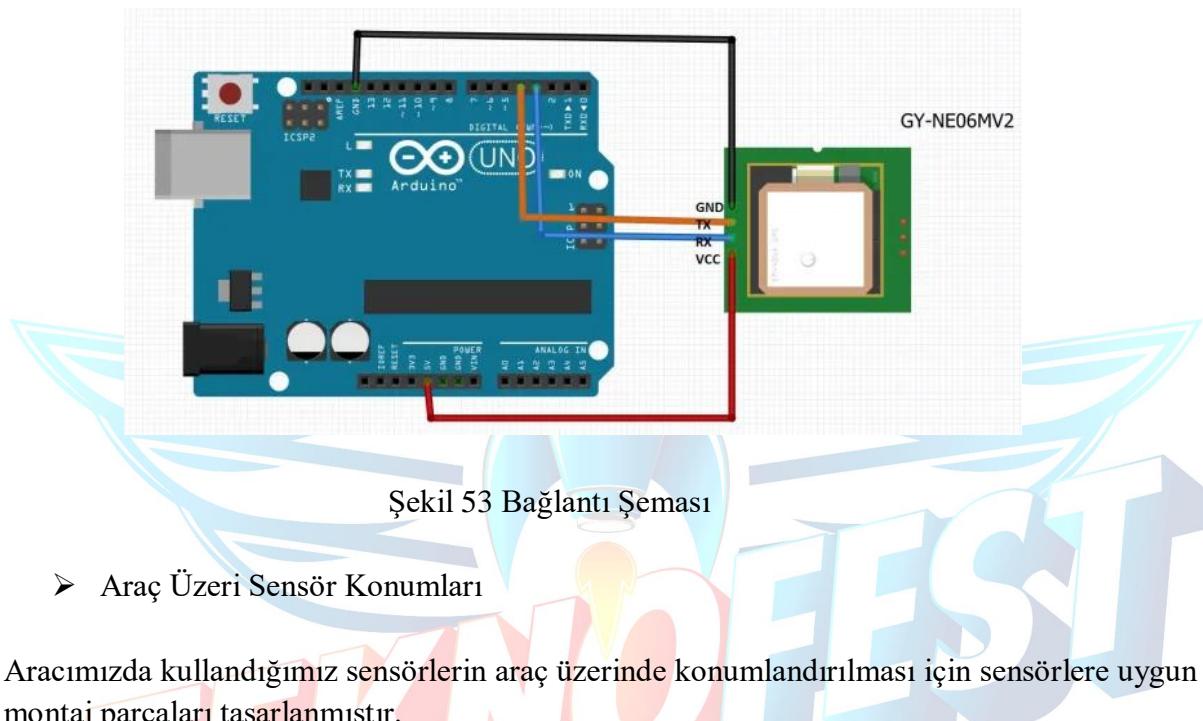
#### 4.5.3. GPS

GPS (Küresel Konum Belirleme Sistemi), uydu sinyallerini kullanarak nokta konum koordinatlarını tam olarak belirler. Bu projede bulunduğuuz yerin konumunu GY-NEO6MV2 GPS kullanarak bulacağız.

Üzerinde NEO-6M-0-001 modülü bulunan bu kart, uçuş kontrol sistemleri başta olmak üzere bir çok projede konum kontrol ve takibi yapmak için kullanılabilen bir ürünüdür. Yüksek hassasiyete sahip olan modül, GPS ile konum bilgisi gerektiren projelerde sıkılıkla kullanılmaktadır. Yaklaşık 5 metrelik bir hassasiyete sahiptir. Modül, Arduino gibi bir çok mikrokontrolcüyle beraber kolaylıkla kullanılabilir.

### Teknik Özellikler:

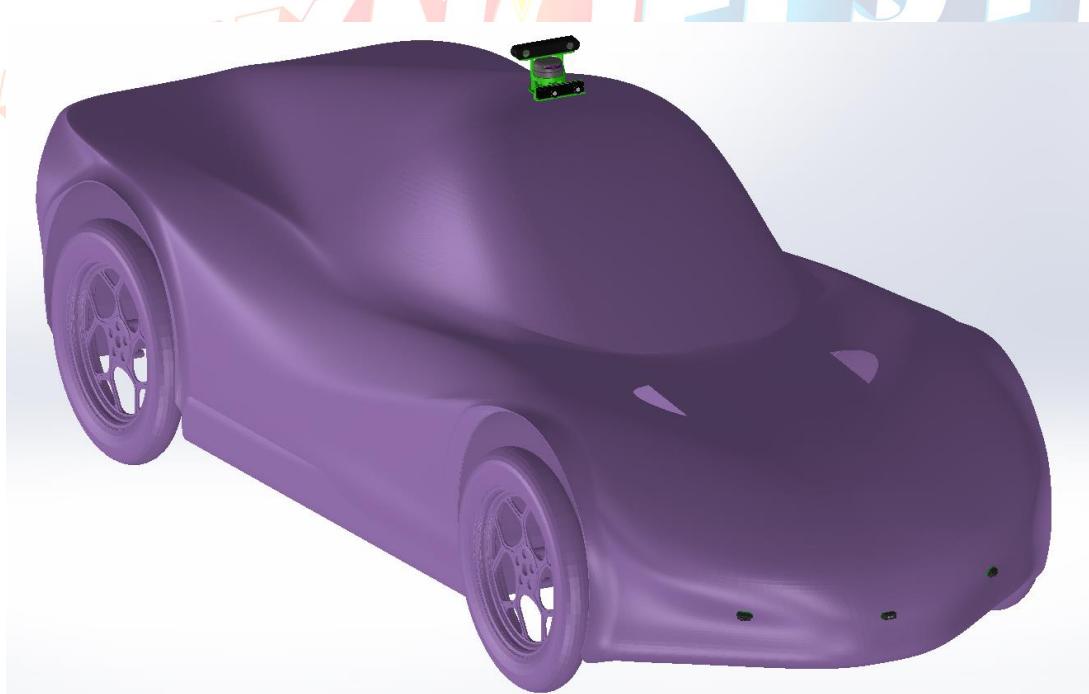
- Çalışma gerilimi : 3.3V-5V
- Haberleşme Birimi: UART (RX-TX)
- 25x25mm Seramik Anten
- Enerji kesildiğinde konfigürasyon ayarlarını EEPROM'da saklama
- EEPROM için kart üzerinde pil bulunmaktadır
- LED ile sinyal uyarı bilgisi
- Default Baud Rate: 9600



Şekil 54 Araç Üzerinde Sensor Konumları Ve Tasarlanan Montaj Parçası



Şekil 55 Araç Üzerindeki Sensör Konumları



Şekil 56 Aracın Farklı Açıdan Görünüşü

## 5. Yazılım Mimarisi

Yazılım mimarımızde kamera ve lidar dahil olmak üzere birçok sensörden veriler alıp bunlarla farklı algoritmalar oluşturup en faydalı ve sağlıklı olanları kullanıyoruz. Kameradan gelen görüntü ve sensörlerden gelen verilerin işlenmesi için şerit tespiti ve engel tanımda python ile görüntü işleme, nesne tespitinde ise derin öğrenme algoritmaları kullanıyoruz. Elimizdeki verileri oluşturduğumuz bu algoritmalar ile anlık işleyip aracın otonom sürüşünü sağlıyoruz. Aynı zamanda aracın otonom sürüşü ve karar mekanizmasını her türlü koşulda en doğru sonucu verecek şekilde geliştirmeyi hedefliyoruz.

Sensörlerden gelen verilerde tutarsızlık yaşanması, yanlış veri gelmesi veya bir şekilde bağlantı sıkıntısı yaşanması durumunda oluşturduğumuz otonom sürüş algoritmasına bu durumları göze alarak eklemeler yaptıktı.

Araçtaki olası bir durumda kamerada bağlantı kopması, kamerada herhangi bir bozulma, şeritlerin olmaması, şeritlerden birinin veya hiçbirinin tespit edilemez halde olması, aracın şerit dışına çıkması durumlarında algoritma uyarı verip lidar sensörden yardım alarak bir süre daha şerit bulmaya çalışır. Hala bulamaz ise motor gücü kesilir ve hızda yavaşlamaya gidilir. Nesne tespitini yapamama durumunda şerit takip algoritmasından yararlanarak aracın güvenli bir şekilde yolda kalması sağlanacaktır.

### 5.1. Nesne Tespiti

Kamrera sensöründen aldığımız verileri aracımızdaki yapay zeka bilgisayarında işlenerek oluşturduğumuz algoritmalarla tespit işlemi gerçekleştirilecektir.

- Tabela ve Trafik Işıklarının Algılanması

Tabela ve trafik ışıklarının tanınmasında kullanabileceğimiz derin öğrenme algoritmalarını araştırmaya devam ederek veri setimize ve kullanacağımız sisteme en uygun modeli bulmayı araştırdık. Yolov4 modelini kullandıkten sonra Yolov5 modelini de kullanmaya karar verdik fakat elde ettiğimiz sonuçlar doğrultusunda, Yolov4 modelinin araştırmaya devam edilmesi kararı alındı. Küçük veri setleri oluşturularak diğer modeller hakkında bilgi toplandı.

Araştırmalarımız ve deneme eğitimlerimiz sonucunda yarışma için kullanılacak modelin Yolov4 üzerinden yapılması kararlaştırıldı.

- Veri setinin hazırlanması

Veri setimizin bir kısmı hazır diğer kısmı takım arkadaşımız ile Google Haritalardan, yolda karşılaştığımız tabelaları çektiğimiz fotoğraflardan ve trafik tabelalarının çeşitli açılardan ve mekanlardan çektiğimiz fotoğraflardan oluşmaktadır.



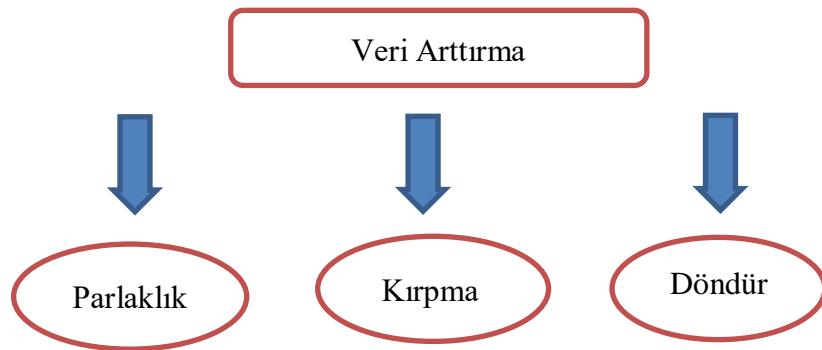
Şekil 57 Veri Setini Oluştururken Çevreden Çektiğimiz Fotograflar

Veri setimiz 13 sınıfından oluşmaktadır.



Şekil 58 Veri Setinde Bulunan Sınıflar

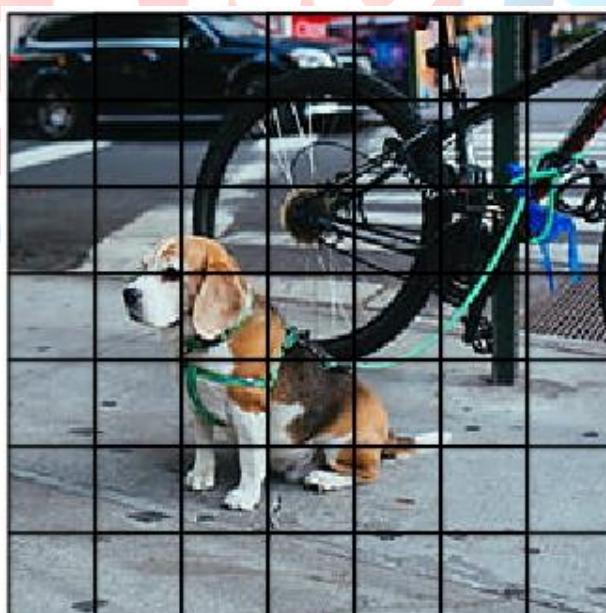
Her sınıfta 2000 fotoğraf bulunmakta olup toplamda 26.000 fotoğraf bulunmaktadır. Coğu fotoğrafta birden fazla tabela olduğu için toplamda 30.000-33.000 arası etiket bulunmaktadır. Veri setimiz ilk veri setimizden farklı olarak birkaç şey değiştirilmiştir. Bunlardan ilki veri setindeki düşük kalitedeki, aşırı karanlık ve aşırı uzaktaki tabela olan fotoğrafların silinmesi ve bunların yerine daha net, yakın ve aydınlichkeit olanlar konulmuştur. Veri setine uyguladığımız diğer işlem veri artırma (Data Augmentation) algoritmasıdır. Öncelikle veri setimizi her sınıf 1000 fotoğraflık olacak şekilde ayarladık geri kalan 1000 fotoğrafı kendi oluşturduğumuz veri artırma algoritmalarıyla 2 katına çıkartarak asıl veri seti oluşturulmuştur.



Şekil 59 Veri Artırımda Kullandığımız Fonksiyonlar

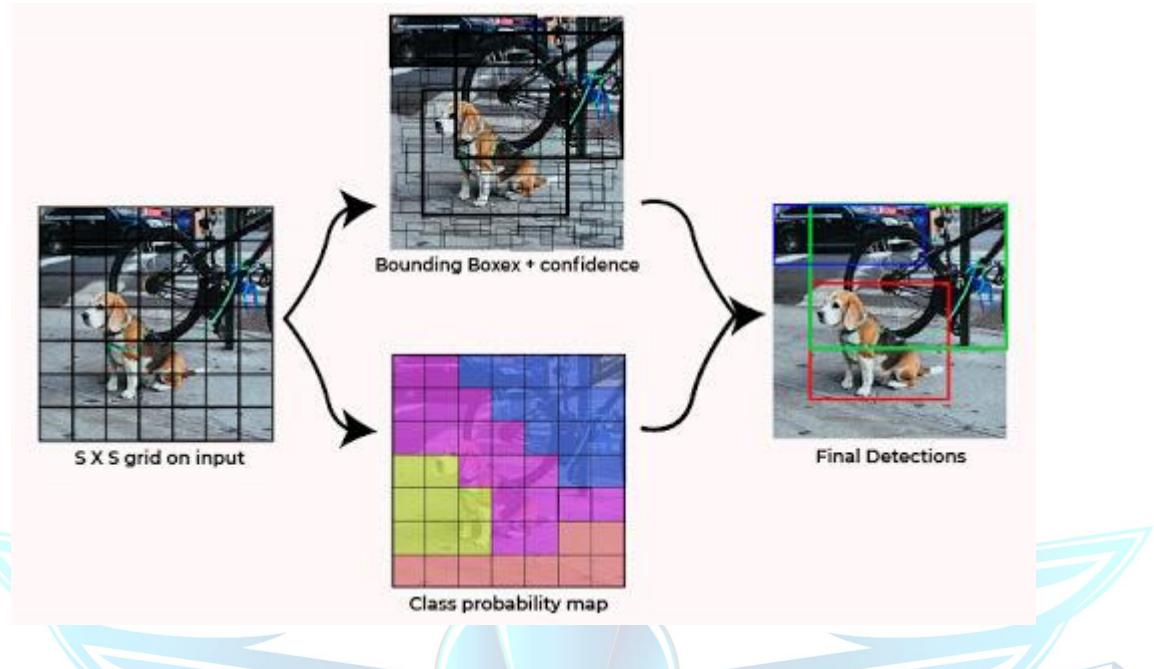
➤ YOLO' nun yapısı ve parametreler

YOLO mimarisi, görüntüyü  $S^*S$  boyutunda bir ızgaraya böler. Nesnenin sınırlayıcı kutusunun merkezi o ızgaradaysa, nesneyi algılamaktan bu ızgara sorumludur. Her ızgara, güven puanıyla sınırlayıcı kutuları tahmin eder. Her güven puanı, tahmin eden sınırlamaların bir nesne içermesinin ne kadar doğru olduğunu ve sınırlayıcı kutu koordinatlarını wrt ne kadar kesin olarak tahmin ettiğini gösterir. Görüntüde mevcut bir nesne varsa, kesinlik ve tahmin kutuları arasındaki güven puanı IoU'ya eşit olmalıdır. Her sınırlayıcı kutu 5 tahminden oluşur:  $(x, y, w, h)$  ve güven puanı.  $(x, y)$  koordinatları, ızgara hücresinin sınırlarına göre kutunun merkezini temsil eder.  $(h, w)$  koordinatları,  $(x, y)$  ile ilgili sınırlama kutusunun yüksekliğini ve genişliğini temsil eder. Güven puanı, sınırlayıcı kutuda bir nesnenin varlığını temsil eder. Her ızgara ayrıca  $C$  koşullu sınıf olasılığını,  $P_{ryi}$  ( $\text{Sınıf } i \mid \text{Nesne}$ ) tahmin eder.



Şekil 60  $S^*S$  Izgarasına Bölünmüş Fotoğraf

Bu olasılık, ızgara hücresinde bir nesnenin varlığında bağlı olarak koşulluydu. Kutuların sayısı ne olursa olsun, her ızgara hücresi yalnızca bir sınıf olasılıkları kümесini tahmin eder. Bu tahminler,  $S^*S^*(5*B+C)$  boyutundaki 3B tensörde kodlanmıştır.



Şekil 61 Yolo Algoritması İşlem Süreci

Yolo algoritmasının hızlı olmasının sebebi fotoğrafları tek seferde işleme yapısından kaynaklanır. Yolov4 Darknet ve Cuda ile yazılan açık kaynaklı bir yapay sinir ağı sistemidir. Nesne tespitinin başarılı olması için geliştirilen Yolov4’ün Darknet kütüphanesini kullandık. Çalışmalarımızda Yolo’nun parametreleri ve değerleri baz alınarak uygun formatta veri seti hazırlanmaya çalışıldı. Yapılandırma dosyasında (.cfg) kullanılan parametreler ve değerler aşağıda açıklanmış olup. **Tablo** ’te dephinildiği üzere tam değerler paylaşılmıştır.

1-) Resim boyutu: Veri setindeki tabelaların işaretlerin boyutları göz önüne alındığında resim boyutu olarak  $416 \times 416$  çözünürlüğü uygun görülmüştür.

2-) İterasyon sayısı: Veri setinin bir tur eğitmek için gerekli parametredir. En az sınıf sayısı  $\times 2000^*$  olacak şeklinde ayarlanabilir.

3-) Subdivision: Girilen batch boyutunu kaç tane mini-batch'e ayıracagını belirten parametredir. Bu sayının kullanılan yapay zeka bilgisayarının sistem özelliklerine göre yükseltilebilir veya düşürtülebilir.

4-) Batch: Veri seti burada girilen değere göre bölünür ve her iterasyonda verilen değer kadar fotoğraf olur. Model eğitimi batch’ta kullanılan değere göre farklılık gösterebilir. Bu sayının kullanılan yapay zeka bilgisayarının sistem özelliklerine göre yükseltilebilir veya düşürtülebilir.

5-) Filtreler: Yapılandırma dosyasında bulunan her yolo katmanından önceki evrişimsel (convolutional) katmanında filters = (sınıf sayısı + 5) x 3 olacak şekilde değiştirilir.

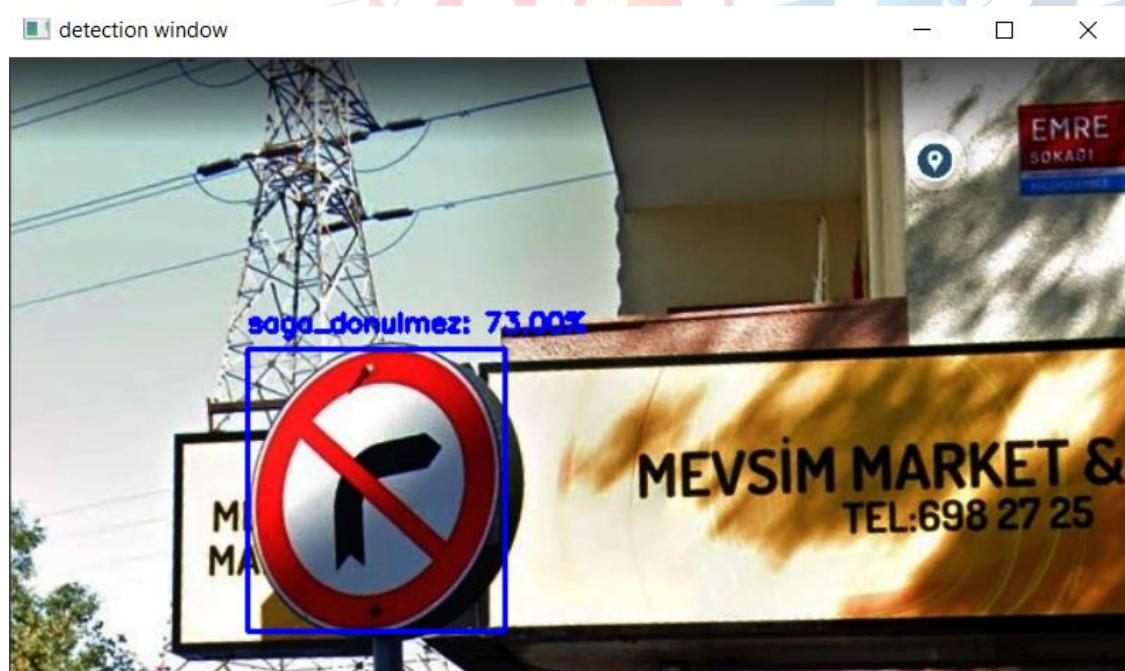
➤ ÖTR Karşılaştırması, Yapılanlar ve Yapılacaklar

Aşağıda karşılaştırma sonuçlarını elde etmiş olup. Yarış zamanına kadar bu verilerin arttırılması ortalama doğruluk değerinin %95-%98 olması hedeflenmektedir. Şu anlık veri seti halen güncellenmekte olup, geliştirilme aşamasında olan sistem ile Yolov4 parametreleri ile daha yüksek bir doğruluk oranı hedeflenmektedir.

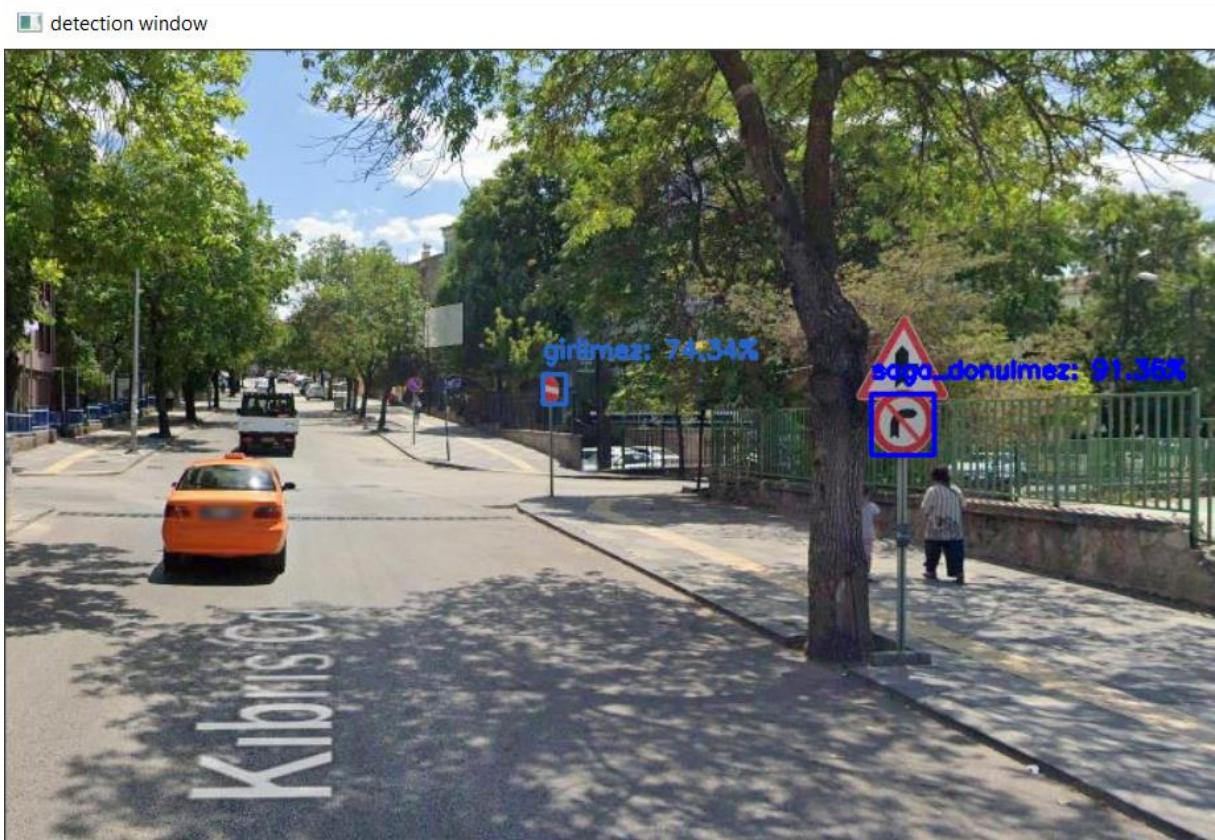
	Eğitim	Eğitim
Fotoğraf Sayısı	15000	26000
Etiket Sayısı	22000-23000	30000-32000
Batch Değeri	64	64
Subdivision Değeri	64	64
Loss Değeri	0.7- 0.75	0.5- 0.7
mAP Değeri	%71,8 ort.	%78 ort
Test ve Train Yüzdeleri	%80- %20	%80- %20

Tablo 9 Eğitim Karşılaştırılması

2. eğitim sonuçları 1. Eğitimimize göre doğruluk oranı çok daha yüksek olup bazı örnekler aşağıda verilmiştir.



Şekil 62 1. Eğitim Sağa Dönülmez Tabela Çıktısı (%73.00)

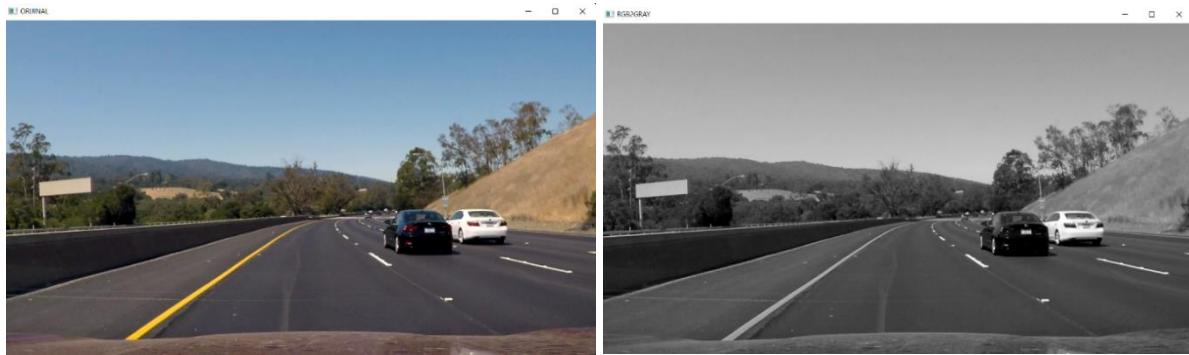


Şekil 63 2. Eğitim Sağa Dönülmez Tabela Çıktısı (%91,36)

## 5.2. Şerit Tespiti ve Takip Algoritması

Şerit takibi algoritmamızı OpenCV, Numpy, Pyplot ve Matplotlib dahil olmak üzere birçok python kütüphanesi kullanarak hazırlamaya devam ediyoruz. Görüntüyü kameradan alındıktan sonra kullanılan kütüphaneler yardımıyla birden fazla filtreleme işleminden geçiyor ve temiz bir görüntü elde ediliyor. Bu görüntü üzerinde fonksiyonlar kullanarak görüntüdeki şeritleri tespit ediyoruz. Şerit takip sistemlerinde sıkça karşılaşılan yöntemlerden biri olan derin öğrenme algoritmaları yerine python görüntü işleme yöntemini daha sağlıklı bulduk. Derin öğrenme yöntemi, algoritmanın kamera ile şeridi tanınması için farklı şartlarda çekilmiş birden fazla yol ve şerit görüntülerini kullanılarak eğitime sokulmasıdır. Bu yöntemde ne kadar ayrıntılı bir eğitim yapılmış olsa da yüzde yüz bir başarıdan söz edilemez. Bu nedenle derin öğrenme algoritması yerine şeritleri gerçek zamanlı bir şekilde tespit etmemizi sağlayan görüntü işleme yöntemini kullanmaya devam ediyoruz. Gerçek zamanlı bir şekilde daha net daha doğru ve kararlı bir şerit tespiti ve takibi yapabilmek için algoritmamızı geliştirdik. Şu anda da halihazır da araştırmaya ve geliştirmeye devam etmekteyiz.

**RGB2GRAYfiltresi:** Kameradan alınan görüntü 3 (RGB) kanallıdır bu nedenle şeritleri diğer renklerden ayırbilmek için gelen kareler 2 kanallı RGB2GRAY formatına çevirdiğimizi böylece siyah ve beyaz piksellerin ayırt edilebilir hale geldiğini belirtmişik. RGB2GRAY formatını kullanmaya devam ediyoruz. COLOR\_RGB2GRAY Filtresi yardımıyla görüntüyü siyah beyaz formata çevirme ile ilgili görseller aşağıda yer almaktadır.



Şekil 64 Orijinal görüntü

Şekil 65 RGB2GRAY filtreli görüntü

**THRESHOLD:** ÖTR raporunda THRESHOLD filtresini kullandığımızı belirtmiştık. THRESHOLD滤镜是我们从报告中使用的。它需要我们指定一个阈值，任何高于该阈值的像素将被设置为白色，而低于或等于该阈值的像素将被设置为黑色。这样，我们就可以将图像转换为二进制格式，从而更容易地进行进一步的处理。在我们的实验中，我们使用了255作为阈值，因为这是白色在RGB颜色空间中的表示。我们还注意到，THRESHOLD滤镜在处理具有不同光照条件的图像时表现得并不理想。因此，我们决定在使用THRESHOLD滤镜之前，先将其转换为灰度图像。这样，我们就可以更好地利用THRESHOLD滤镜来检测道路上的车辆。



Şekil 66 Önceki THRESHOLD kullanılmış görüntü



Şekil 67 Yeni THRESHOLD kullanılmış görüntü

**GAUSSİANBLUR:** Elde edilen görüntü üzerinde GaussianBlur metodu uygulanarak bulanıklaştırma işlemi yapılmaktadır. Bu metodu kullanmamızın sebebi görüntüdeki gürültüden kurtulup daha doğru sonuçlar alma adına karmaşık bir görüntüyle çalışmak istemememizdir. Bu sayede kenar tespiti sırasında daha temiz bir görüntü elde edilir. GaussianBlur bizden 2 parametre istemektedir. Biz araştırmalar ve denemeler sonucu bu parametrelerde değişikliğe giderek daha da temiz ve karmaşık olmayan aynı zamanda da şeritlerin bulanıklaşmadan en az etkilendiği bir görüntüye ulaştığımızı gördük ve yazılımımızı buna göre güncelledik.

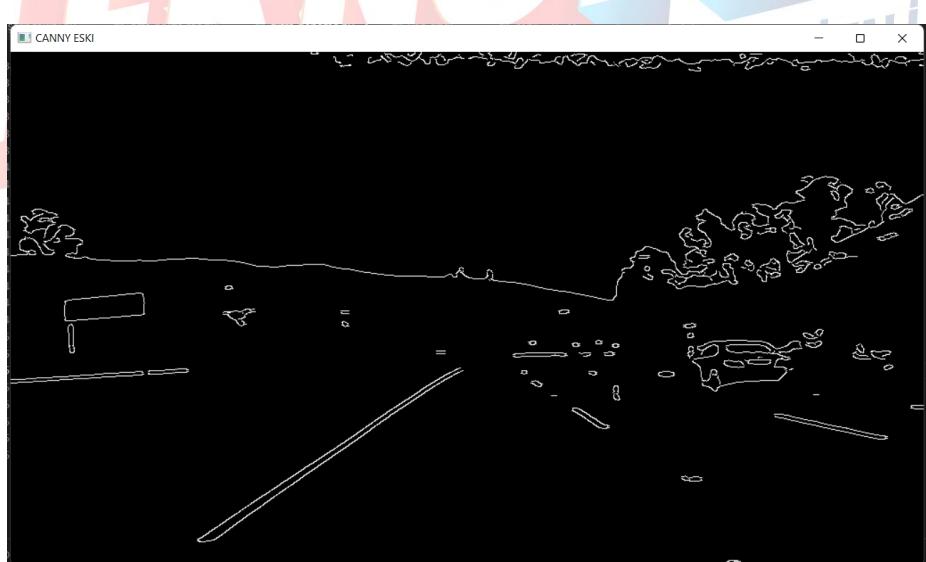


Şekil 68 Eski GAUSSİANBLUR uygulanmış görüntü

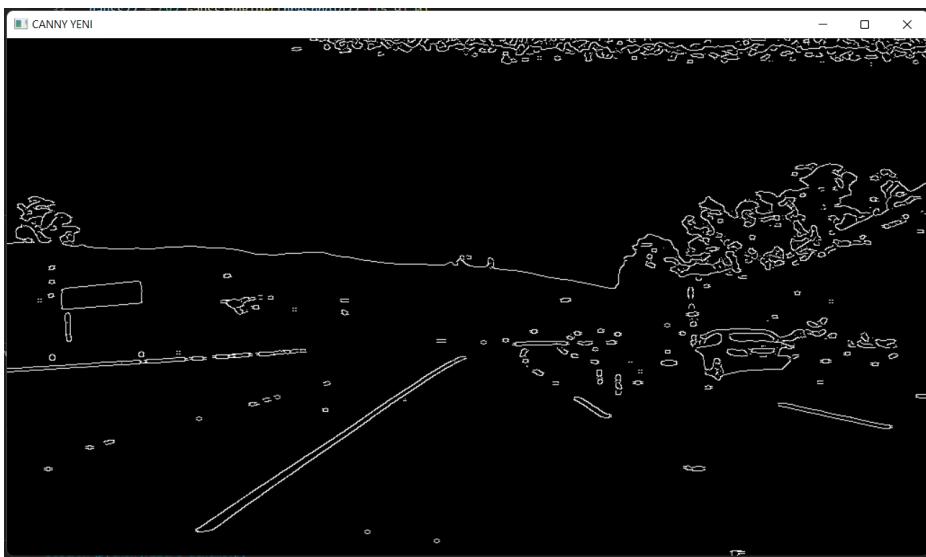


Şekil 69 Yeni GAUSSİANBLUR uygulanmış görüntü

**CANNY:** Yukarı da belirtilen filtrelemelerdeki değişimler sonucu görüntümüz artık şerit algılama için daha da uygun hale gelmiştir. Bunların üzerine canny filtresini uygulayarak kenar tespiti yapılır ve görüntüde kenar tespiti yapılan çizgiler üzerine şerit çekilebilir hale gelir. Cannyfiltresi de bizden belli başlı parametreler istemektedir. İlk çalışmalarımızda bu parametrelere klasik değerler verilmişken yine çeşitli araştırmalar ve denemeler sonucu daha stabil sonuçlar verdiği gördüğümüz parametrelere ulaştık. Bu parametrelerdeki değişiklikler sonucu daha hassas şerit tespiti yapabiliyoruz. Bu sayede algoritmamız günlük hayatı daha ekstrem durumlar (şeritlerin hafif silinmesi vs.) için hazır hale geldi. Sabit kalmayıp daha fazla çalışarak kendimizi güncellemeye ve geliştirmeye devam ediyoruz. Yukarıda anlatmak istediğimiz değişiklik, ulaştığımız yeni gelişmeler ve sonuçlar aşağıdaki görsellerde yer almaktadır.



Şekil 70 Canny ile kenar tespiti yapılmış görüntü



Şekil 71 Canny ile kenar tespiti yapılmış yeni görüntü

Filtrelerdeki tüm bu değişikliklerin ardından artık görüntüdeki nesnelerin kenar çizgilerinin çekilipli tespit edilmesinde daha verimli sonuçlar almaktayız. Bu kenar tespitini geliştirdiğimiz nesneleri ve noktaları verilen parametrelere göre birleştirerek yeni çizgiler üretmek için kullanacağız. Bu yeni üretilecek çizgiler yine şerit üzerinde toplanan noktalardan oluşur. Noktaların birleşimi ile oluşan çizgiler üzerine oluşturmuş olduğumuz `draw_the_lines` fonksiyonu ile boyama işlemi yapılıyor. Ana şeritler ve görüntüdeki yoğun gürültüler dışında az sayıdaki noktalar çizgi çizmeye yetmeyeceği için algoritma onları göz ardı ederek tespit edilen şeritlerin dışında tutacaktır. Bu boyamanın rengi, kalınlığı, saydamlığı isteğe bağlı bir şekilde algoritmamız tarafından değiştirilebilir.

Değiştirdiğimiz parametreler sonucu şerit tespitini daha detaylı bir şekilde yapabiliyoruz. Fakat görüntünün her yeri daha detaylı şerit tespiti yapabilmek adına filtrelendiği için görüntümüzde şeritler dışında da birçok gürültü görebiliyoruz. Tabii ki filtrasyon işlemleri sayesinde bu gürültü en aza indirildi. Yine de daha detaylı ve günlük hayatı uygun bir şerit tespiti için bu gürültüler az da olsa görüntümüzde hala mevcut. Bu sebepler ve şeritlerin sağ ve sol kenarlarında belirebilecek olası nesne karmaşıklıklarını engellemek adına gereksiz bölgeleri kırpmaya devam etmekte kararlıyız. Görüntümüz perspektif olduğu için bu bölgeleri üçgen şeklinde kırıyoruz. Gereksiz bölgeleri kırmada kullanacağımız üçgenin köşe noktalarını aracımız için kamerasının bağlandığı konuma ve kamerasının bakış açısına uyarlayarak en verimli olacak şekilde seçtik. Kamerasının konumu için daha verimli bir nokta bulunması durumunda bu üçgenin köşe noktaları değiştirilebilmektedir.



Şekil 72 Kırpma işlemi yapılmamış görüntü

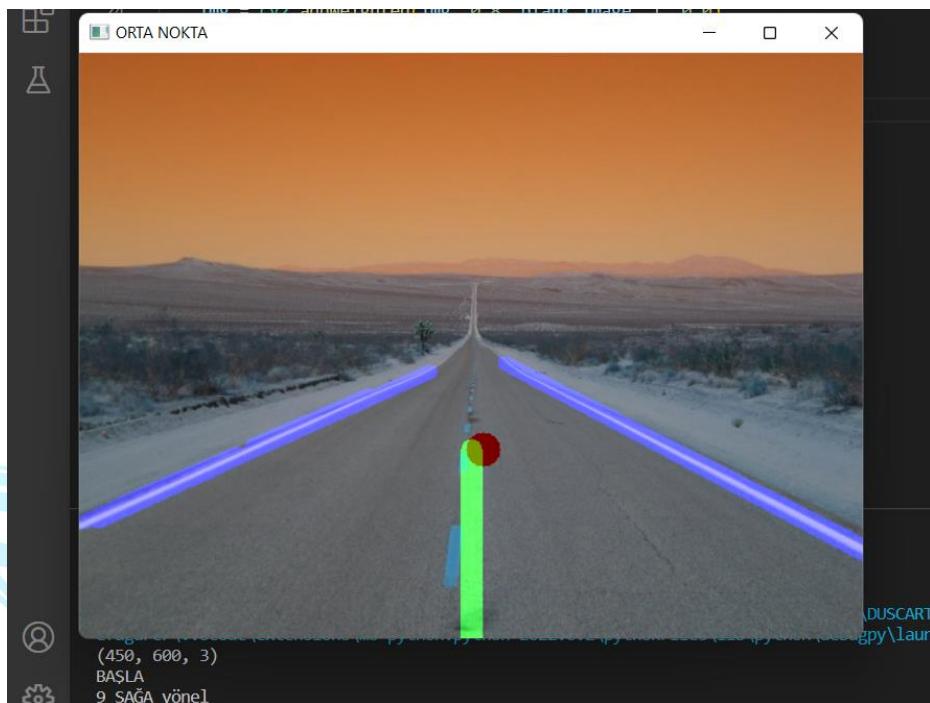


Şekil 73 Kırpma işlemi yapılmış görüntü

#### ➤ Orta Nokta Tespiti ve Araç Hareketi

Ön tasarım raporunda görüntümüzdeki şeritlerin orta noktasını bulabilmek için img rolls ve cows değerlerine göre gelen görüntünün orta noktasını renkli bir çizgi kullanarak işaretlediğimizi belirtmişik ancak yapılan araştırmalar ve fikir alışverişleri sonucu moments metodu ile şeritlerin ağırlık merkezini bulmanın daha verimli olduğuna karar verdik. Algoritmamızda alınan görüntüde aracın tam ortasına kameranın konumlandırıldığı noktaya bağlı olarak değişiklik gösterecek şekilde uzunluk, kalınlık ve RGB renk değerleri değiştirilebilir bir şerit çekilmektedir. Moments metoduyla şeritlerin ağırlık merkezinin konumu tespit edilip alınan görüntüde bu konuma yarı çap ve RGB renk kodları değiştirilebilir bir nokta koyulmuştur. Algoritma tespit edilen şeritlerin ağırlık merkezi ile

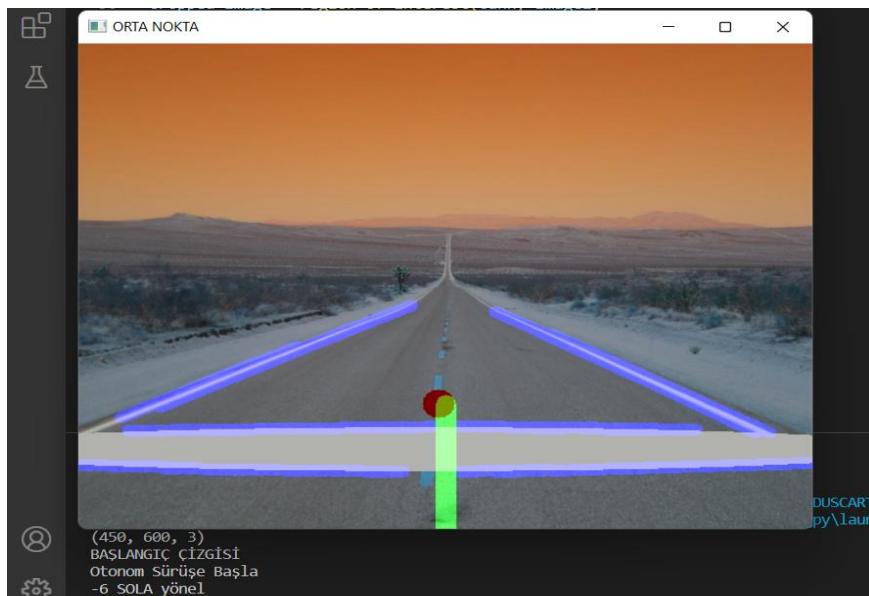
aracın konumu arsındaki mesafeyi ölçerek direksiyon kontrolü için kullandığımız motora hangi yöne ne kadar dönülmesi gerektiği bilgisini verir ve aracın şerit takibi tam anlamıyla gerçekleşir. Ayrıca terminalde bu mesafeyi ve hangi yöne yöneltinmesi gerektiğini belirtir. Orta nokta tespiti ve aracın konumuna göre değiştirilmesi geren konumun terminalde gösterildiği görüntüler aşağıda yer almaktadır. Bordo renkli nokta şeritlerin ağırlık merkezi olup yeşil çizgi aracın konumu ve mavi çizgiler ise yol şeritleridir.



Şekil 74 Orta nokta tespit edilmiş görüntü

#### ➤ Otonom Sürüs Başlangıç Noktası Tespit Algoritması

Tasarlanan otonom aracın otonom sürüs görevinin başlaması için başlangıç şeridini başarılı bir şekilde tespit etmesi gerekmektedir. Bu noktanın tespiti için özgün bir algoritma düşünüldü. Uyguladığımız yöntem şöyledir: kameradan alınan görüntüde başlangıç çizgisinin bulunması durumunda tespit edilen şeritlerin arasına başlangıç çizgisi de dahil olmuş olacak. Bundan dolayı tespit edilen şeritlerin ağırlık merkezi aşağıya kayacaktır. Bizde bundan yararlanmak adına görüntüde, kamera konumuna ve açısına göre değişeceğ bir nokta belirledik. Ağırlık merkezinin bu belirlenen sınır noktasını geçmesi durumunda araç otonom sürüse başlayacaktır. Bu oluşturduğumuz algoritmanın doğruluğu yapılan farklı testler ile de doğrulanmıştır.



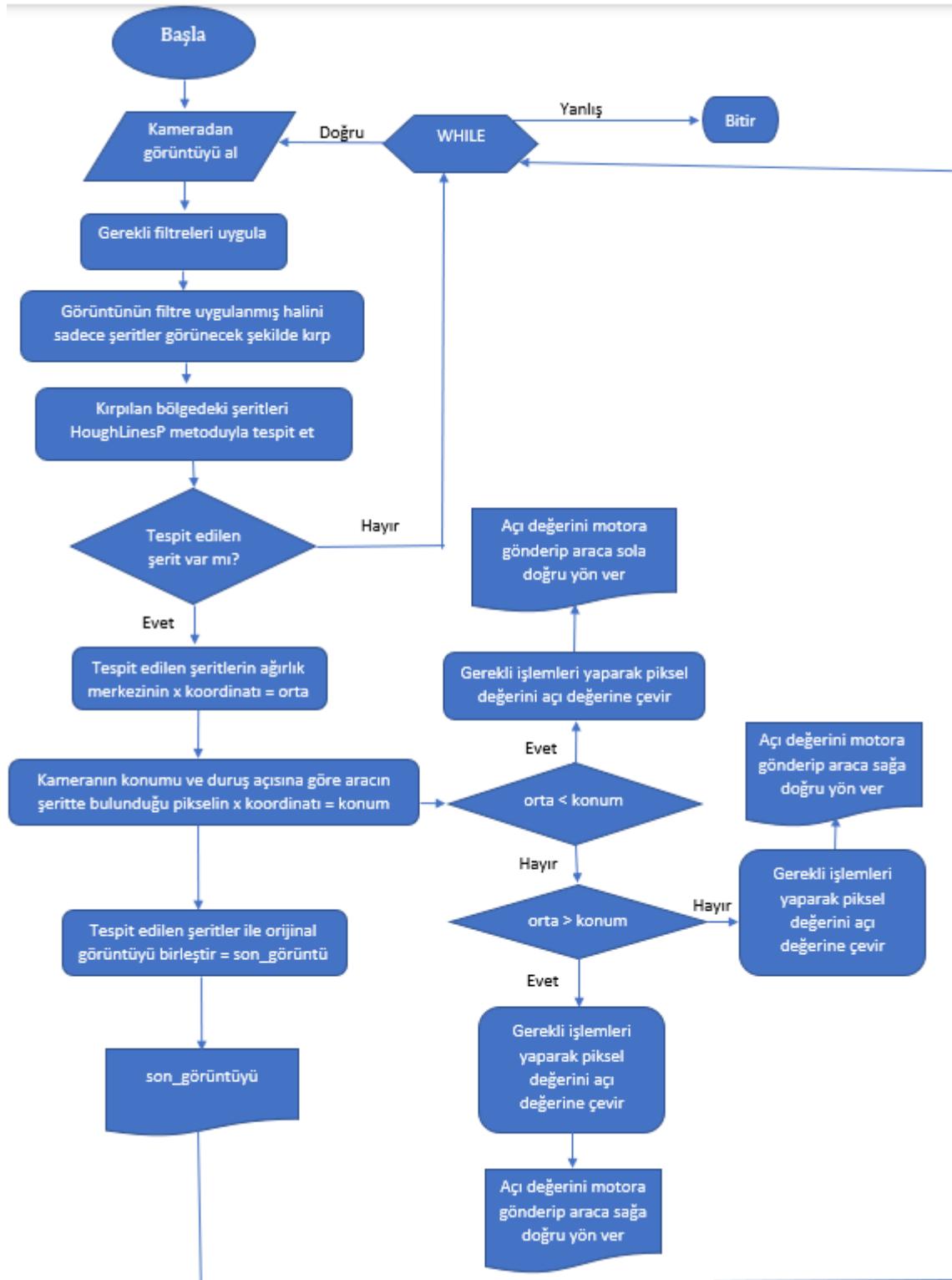
Şekil 75 Başlangıç Şeridi Tespiti

#### ➤ Otonom Sürüş Bitiş Noktası Tespit Algoritması

Tasarlanan otonom aracın otonom sürüşünün tam anlamıyla gerçekleşmesi için başlangıç şeridinin tespitinin yanı sıra bitiş şeridinin başarılı bir şekilde tespit edilmesi de önemlidir. Bitiş şeridinin tespiti için başlangıç şeridinin tespitinde kullanılan algoritmayı kullandık. Aynı şekilde bitiş şeridi görüntünün üst tarafından kamera karesine girmesi ve şerit tespit algoritmasının bitiş çizgisini tespit etmesi durumunda algılanan tüm şeritlerin ağırlık merkezi yukarıya kayacaktır. Bizde bitiş çizgisi için tipki başlangıç şerit tespitinde olduğu gibi bir nokta belirledik. Ağırlık merkezinin bu belirlenen sınır noktasını geçmesi durumunda araç bunu algılayacak ve otonom sürüsür duracaktır. Bitiş çizgisi tespiti için uyguladığımız bu yöntemin doğruluğu da yapılan farklı testler ile doğrulanmıştır. Bunun ile ilgili görseller aşağıda yer almaktadır.



Şekil 76 Bitiş Şeridi Tespiti



Şekil 77 Şerit Takip Algoritması Akış Diyagramı

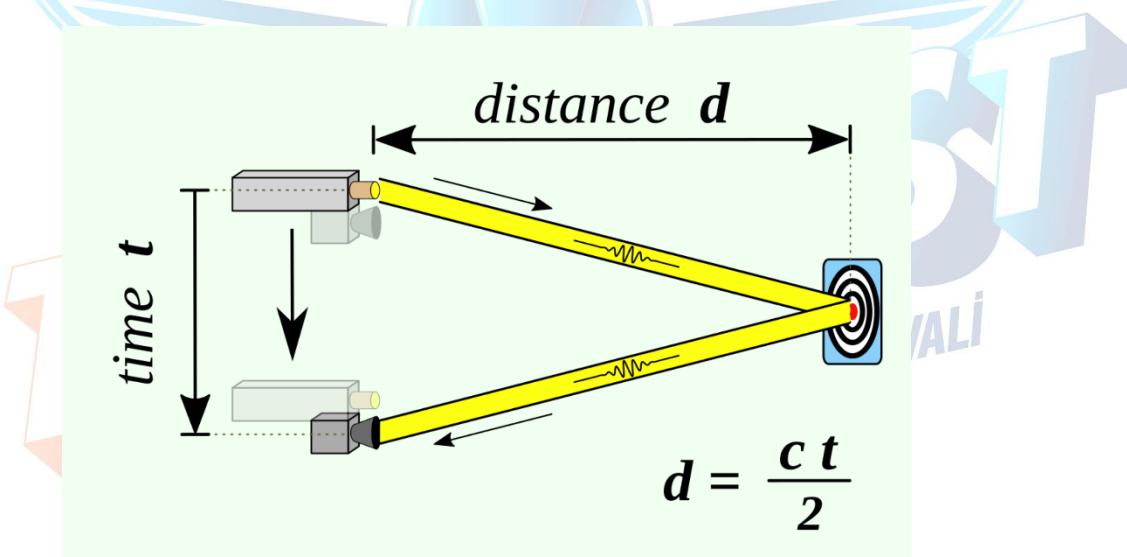
### 5.3. Parkurdaki Engellerin Tespiti ve Şerit Değiştirme

#### ➤ Parkurdaki Engellerin Tespiti

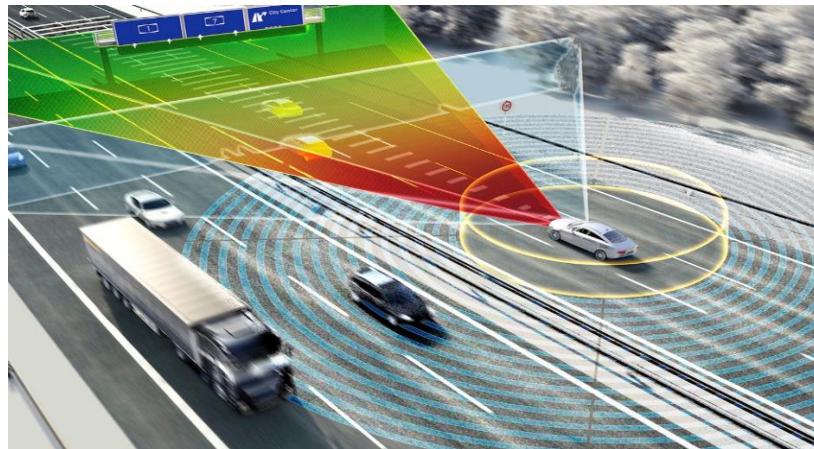
Parkurda aracımızın karşısına çıkabilecek engellerin tespiti, lidar tarafından sağlanan 2 boyutlu harita verisi ve aracın görüş açısını gören kamera tarafından sağlanan 2 boyutlu kamera görüntü verisinin eşzamanlı olarak alınmasıyla sağlanacaktır. Bunun için kullandığımız sensörlerimiz RPLIDAR A3M1 ve TF Mini LİDAR lazer menzil sensörüdür. Kullandığımız kamera ZED2 Stereo kameradır. Lidarımız 0.15 m'den 25 m'ye kadar mesafe ölçümü yapabilmektedir.

Lidar cihazı nesneye gönderilen lazer ışınlarının algılayıcıya geri dönüş süresini kullanarak engele olan mesafeyi hesaplar. Nesneye gönderilen lazer ışınlarının geri dönüş süresi ışık hızında olduğundan zaman bilgisini de kullanarak mesafe bilgisini elde edebiliriz. Lazer ışınları bir kez nesneye çarpmaya ve bir kez de tekrardan algılayıcıya dönmüşinden dolayı 2 kez etkileşime girmektedir. İşinların algılayıcıdan çıkışın tekrar algılayıcıya dönmesine kadar geçen süre uçuş süresi olarak adlandırılır. Biz ise, sadece nesnenin uzaklığını bulmak istediğimiz için ışık hızı ile aracın kat ettiği uçuş süresini çarptıktan sonra bu sonucu ikiye böleriz ve şöyle bir formülle karşılaşmış oluruz:

$$\text{Ölçülen nesnenin uzaklığı} = (\text{Işık hızı} \times \text{Uçuş süresi}) / 2$$



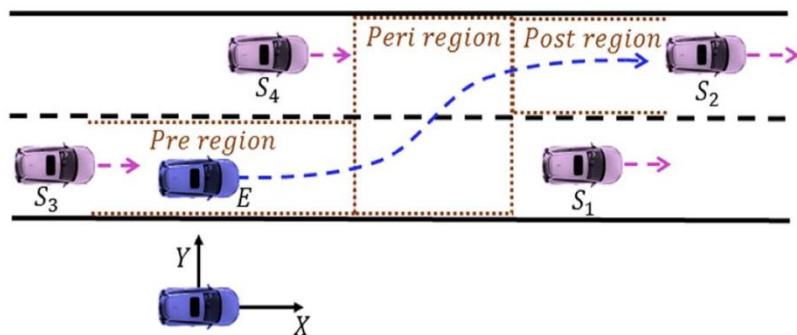
Şekil 78 Lidarla Mesafe Hesaplama Formülü



Şekil 79 Lidar Mesafe Ölçeklendirmesi

TF Mini LİDAR ve RPLIDAR A3M1 birbirlerini destekleyecek şekilde mesafe ölçümünü gerçekleştirecektir. Engelimizin RPLIDARIN görüş açısına takılmadığı noktalarda TF Mini Lidar desteği sayesinde engeli gözden kaçırma durumunun önüne geçilmiş olunacaktır. ZED2 Stereo Kamera kullanarak ise şerit değişikliği esnasında şerit tespit edilebilecek ve bu sayede şerit orta noktası doğru hesaplanacaktır.

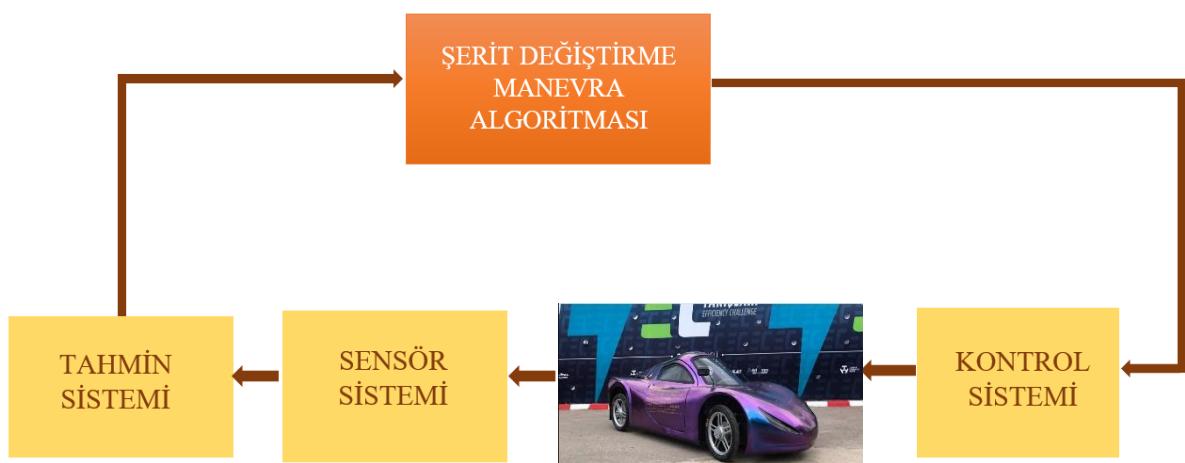
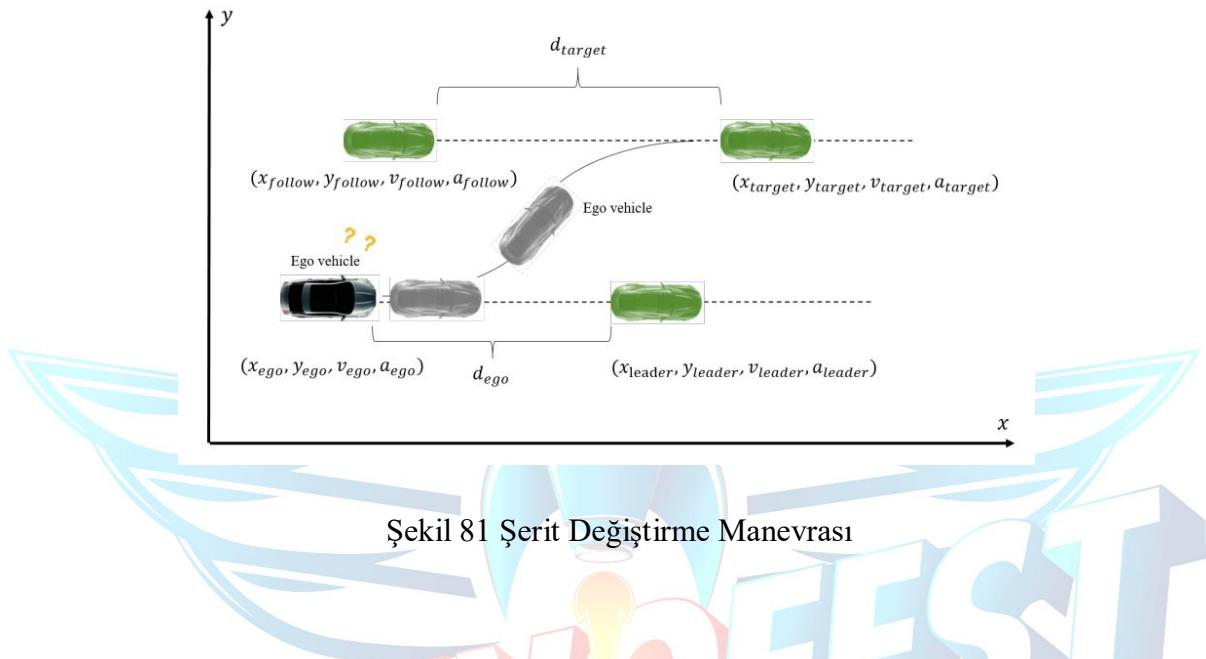
Parkur üzerinde bir engelle karşılaşlığımızda lidarımızdan aldığımız mesafe bilgisini kullanarak engele 25 m'den az bir mesafe kala aracımız kameradan aldığı gerçek zamanlı görüntü işleme verileri aracılığıyla servo motoru harekete geçirerek tekerlekleri engelin olmadığı şeride doğru çevirerek diğer şeride geçmek için yön değiştirmeye başlayacaktır. Aracımız hangi şeride geçeceğine Lidar ile yol kenarlarındaki bariyerlere olan mesafeyi ölçerek karar verecek ve kendine daha uzak olan bariyerin bulunduğu şeride geçecektir. Yeni şeride geçtiğinde yine gerçek zamanlı görüntü işleme verilerini kullanarak şerit orta noktasını tespit edecek ve yoluna devam edecektir.



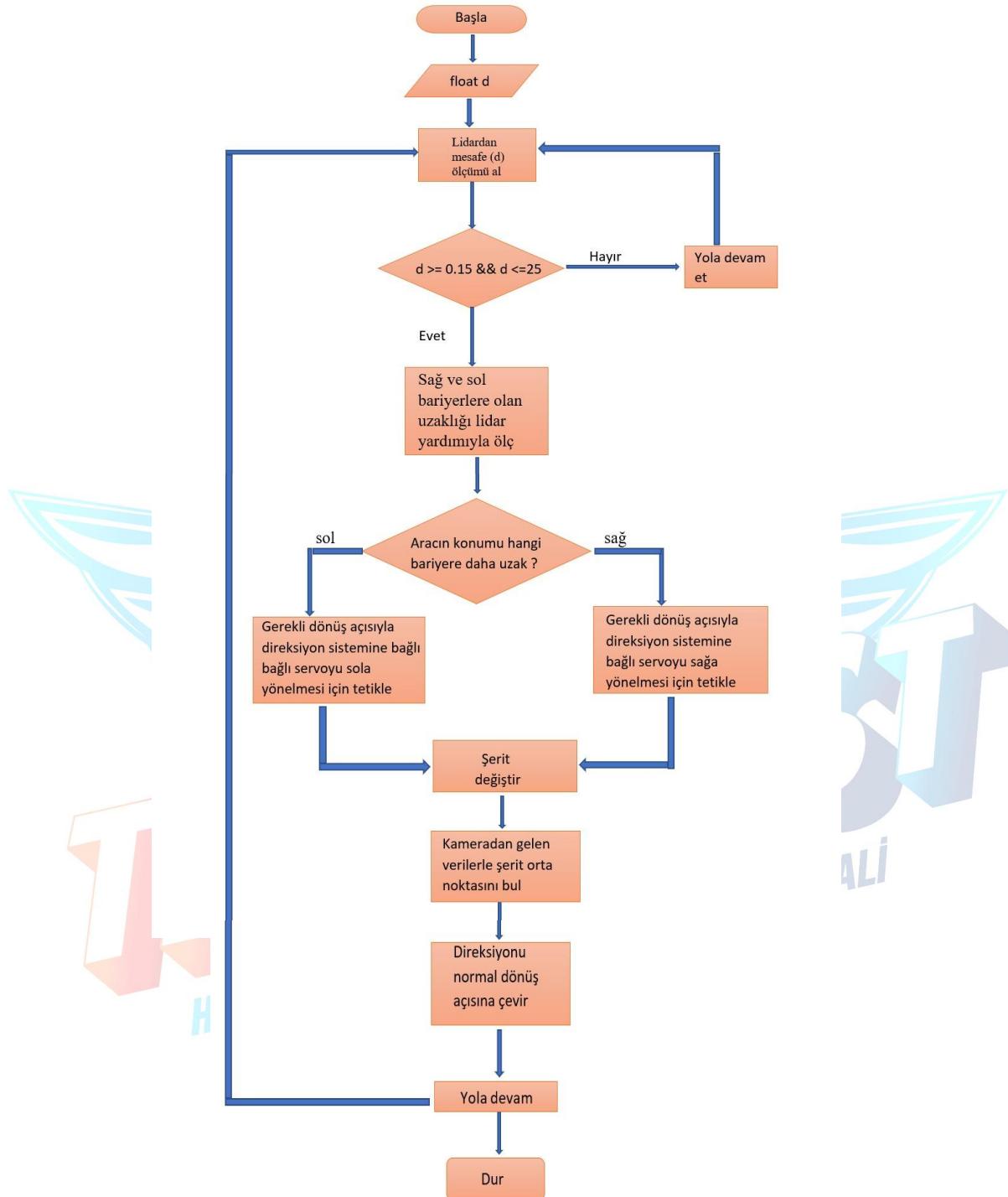
Şekil 80 Şerit Değiştirme Manevrası

## ➤ Şerit Değiştirme Manevra Algoritması

Kullanacağımız sensörler aracılığıyla aracımızın engeli tespit ettiğten sonra, engelin bulunmadığı şeride doğru geçiş ve şeride geçtikten sonra orta noktayı bularak yoluna devam edebilmesi bilgileri doğrultusunda çizdiğimiz akış diyagramımızla da şerit değiştirme manevrası algoritmamızı şekillendirmiştir olduk.



Çizdiğimiz şerit değiştirme manevra algoritması akış diyagramımız:  
 $d$  = Aracımız ve engel arasındaki mesafe

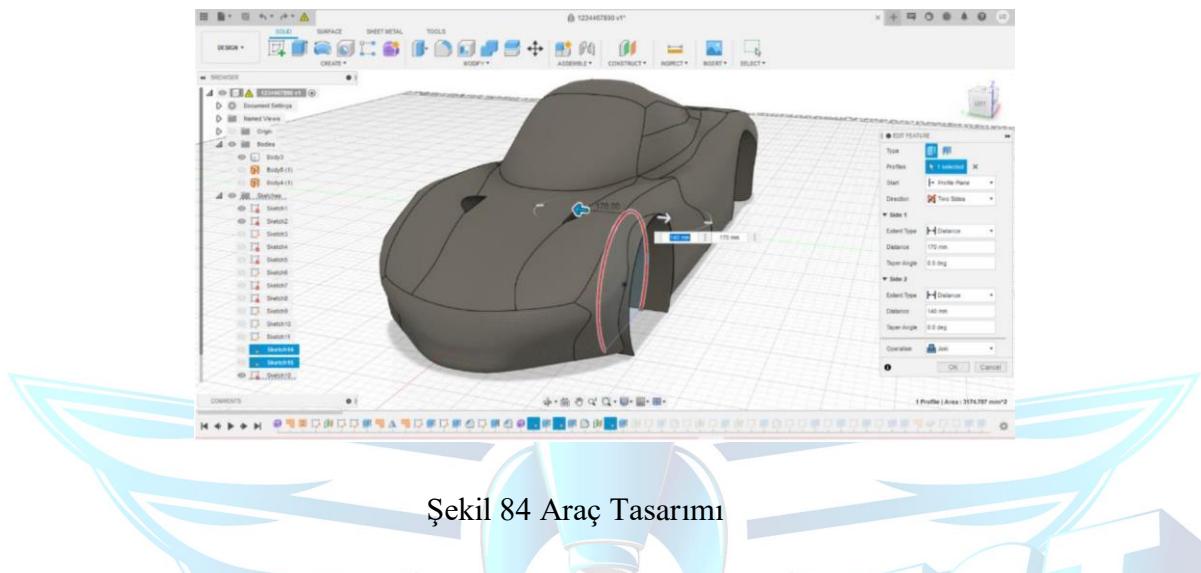


Şekil 83 Şerit Değiştirme Algoritması Akış Diyagramı

## 6. Özgün Bileşenler

Yarışmamız için hazırlamış olduğumuz aracın şase, kaporta, mekanik, elektronik ve yazılımsal tüm sistemleri ekibimize aittir. Araç tasarım süreci, imalatı ekibimiz tarafından yapılmış olup, mekanik aksamlar ve motorlar araca entegre edilmiştir.

- Araç kabuğu ve şasesinin tasarımı ve üretimi ekibimiz tarafından gerçekleştirilmiştir.

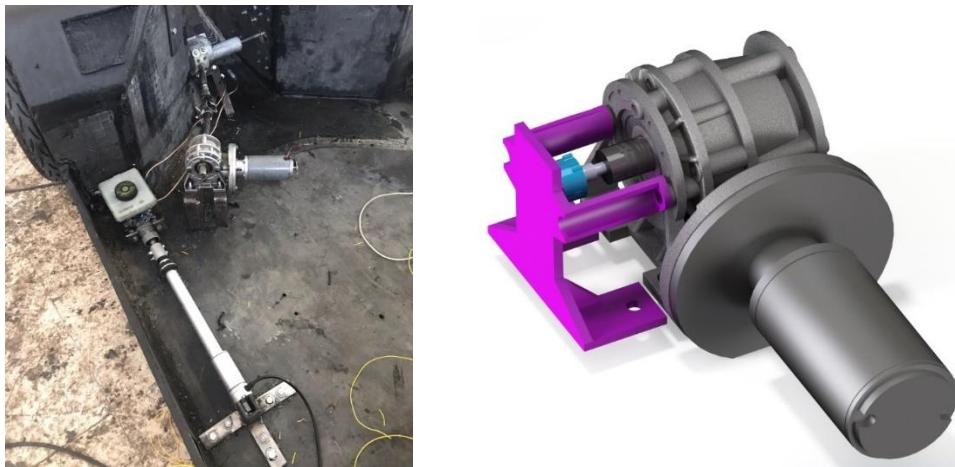


Şekil 84 Araç Tasarımı



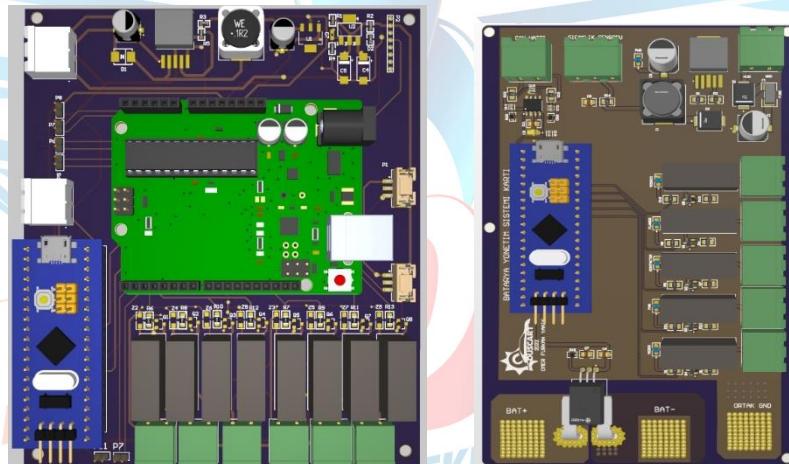
Şekil 85 Araç Üretimi

- Aracımızda kullanacağımız direksiyon ve fren sistemleri aracımız üzerinde uyumlu bir şekilde çalıştırılmak üzere ekibimiz tarafından tasarlandı.



Şekil 86 Fren ve Direksiyon Sistemi Tasımı

- Alt sistemleri kontrol edeceğimiz araç kontrol kartı ve batarya yönetim sistemi kartı ekip üyelerimiz tarafından tasarlanmaktadır. Tasarlanan Araç Kontrol Kartının ve Batarya Yönetim Sisteminin detayları “Donanım Mimarisi” başlığı altında detaylı anlatılmıştır.



Şekil 87 AKS ve BYS PCB Tasarımları

- Araca konumlandırdığımız kameralardan aldığımız veriler üzerinde işlem yaparak aracımızın şeritler arasında kalmasını sağlayan şerit takip algoritmamızı özgün olarak geliştirdik. Aynı zamanda kullanılan filtrelerdeki parametreler de farklı koşullarda test edilerek şerit tespiti için en uygun şekilde seçilmiştir. Orta nokta tespitinde ise tamamen özgün bir algoritma kullandık ve revize etmeye devam etmekteyiz.
- Trafik levhalarını ve ışıklarını tespit etmek için uzun süren çalışmalar sonucu çeşitli ortamlardan toplayarak kendi veri setimizi elde ettik fakat daha sonra eğitimimiz esnasında ortaya çıkan hata paylarını en aza indirmek adına veri setimizi genişlettik. Genişletilmiş veri setimiz ile kendi özgün modelimizi tekrar eğiterek daha yüksek doğruluğa sahip sonuçlar elde ettik.

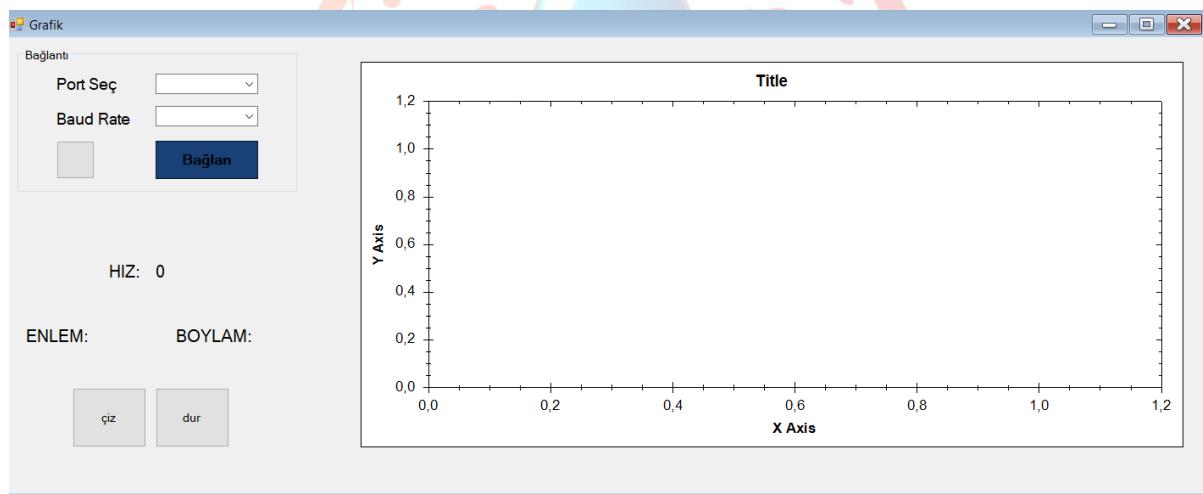
## 7. Test

Aracımızın stabilizasyonunu ölçmek ve farklı durumlardaki tepkilerini görebilmek amacıyla ivmelenme, frenleme, yönlendirme olmak üzere 3 farklı teste tabii tuttuk. Test sonuçlarını analiz edip bu analiz sonuçlarını grafiğe döktük (Bu verilere aşağıda yer verilmiştir). Sonuçları daha net bir şekilde yorumlayarak yazılımsal ve donanımsal değişiklere gittik.

Aracımızın test sürüşleri, araç kontrol kartımızda bulunun bluetooth sensörü ile uzaktan gerçekleştirilmişdir. Aracımızın hızını Hall effect sensörü ile ölçtük. GPS ile de konum bilgilerimizi gerçek zamanlı olarak aldık.

Aracımızın haberleşmesinde kullandığımız bluetooth sensör dışında kontrol kartımıza DORJI RF modül de takarak verileri gerçek zamanlı olarak bilgisayarımıza aktardık.

Aracımızın istenilen girdilere göre test sürüşü yapılrken aynı zamanda C# programlama dilini kullanarak tasarladığımız arayüz uygulaması ile gelen verilerin anlık olarak grafiğini oluşturduk.



Şekil 88 Arayüz Tasarımı

Test sonuçlarını gerçek zamanlı olarak grafiğe döktüğümüz için simülasyondan aldığımız veriler ile aracımızın sürüsünde aldığımız verileri tek bir grafik üzerinde göstermedik. Simülasyonda bulunan aracımızın istenilen girdilere göre yapılan testleri sonucunda aldığımız verileri de Python programlama dilinde Matplotlib kütüphanesini kullanarak grafiğeleştirdik.

Test sonuçlarını elde etmek için gerçekleştirdiğimiz test sürüşlerimizden bir kısmının videosuna aşağıda bulunan linkten ulaşabilirsiniz.

Test Sürüşi Video Linki: [https://youtu.be/MGM\\_Ln9rTmo](https://youtu.be/MGM_Ln9rTmo)

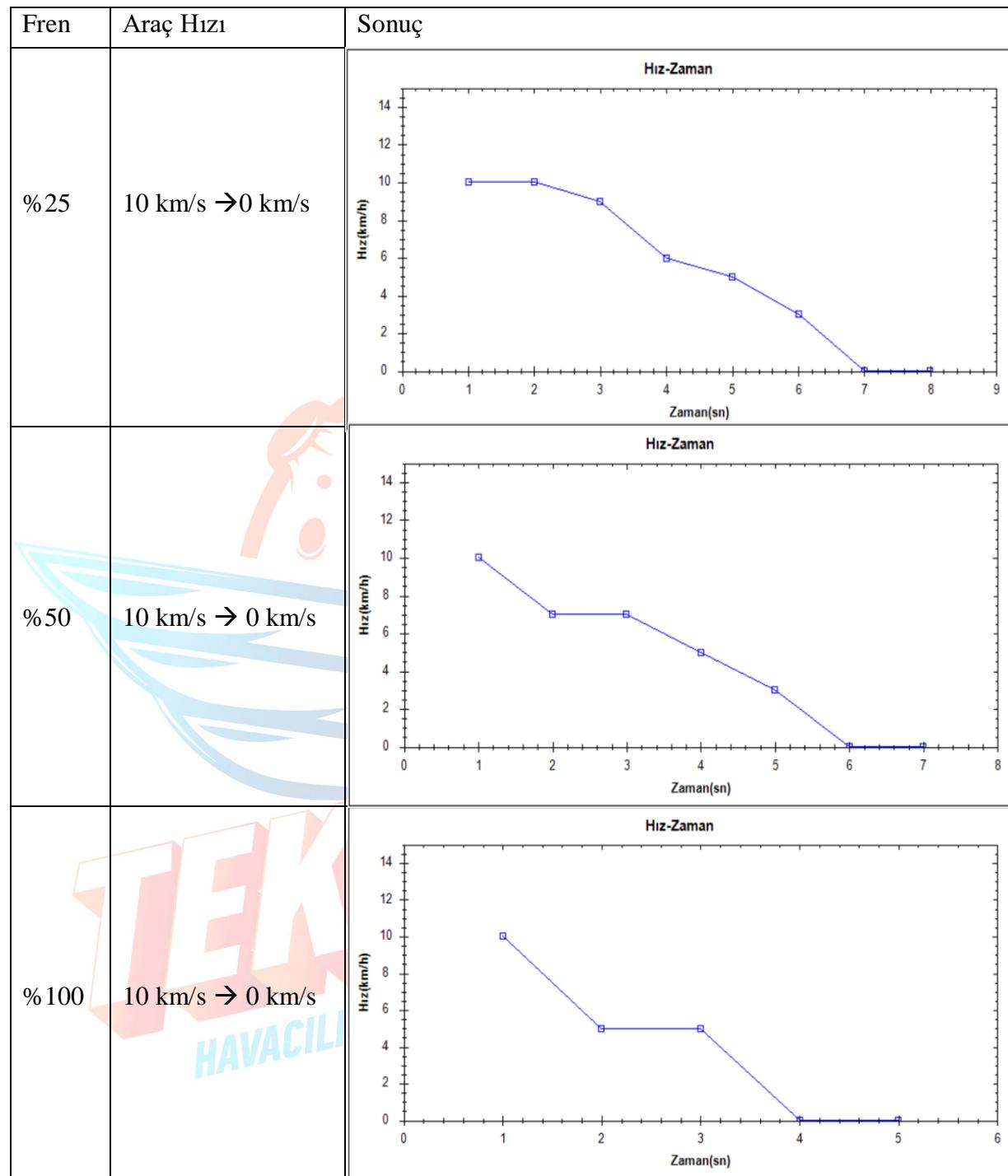
## İvmelenme

Gaz	Araç Hızı	Sonuç														
%25	0 km/s → 10 km/s	<p>Hız-Zaman</p> <p>Yatay eksen: Zaman(sn) (0-12)</p> <p>Dikey eksen: Hız(km/h) (0-14)</p> <table border="1"> <thead> <tr> <th>Zaman(sn)</th> <th>Hız(km/h)</th> </tr> </thead> <tbody> <tr><td>0-2</td><td>0</td></tr> <tr><td>2-3</td><td>5</td></tr> <tr><td>3-4</td><td>5</td></tr> <tr><td>4-6</td><td>8</td></tr> <tr><td>6-10</td><td>10</td></tr> <tr><td>10-12</td><td>10</td></tr> </tbody> </table>	Zaman(sn)	Hız(km/h)	0-2	0	2-3	5	3-4	5	4-6	8	6-10	10	10-12	10
Zaman(sn)	Hız(km/h)															
0-2	0															
2-3	5															
3-4	5															
4-6	8															
6-10	10															
10-12	10															
%50	0 km/s → 10 km/s	<p>Hız-Zaman</p> <p>Yatay eksen: Zaman(sn) (0-9)</p> <p>Dikey eksen: Hız(km/h) (0-14)</p> <table border="1"> <thead> <tr> <th>Zaman(sn)</th> <th>Hız(km/h)</th> </tr> </thead> <tbody> <tr><td>0-3</td><td>0</td></tr> <tr><td>3-4</td><td>5</td></tr> <tr><td>4-5</td><td>5</td></tr> <tr><td>5-6</td><td>10</td></tr> <tr><td>6-9</td><td>10</td></tr> </tbody> </table>	Zaman(sn)	Hız(km/h)	0-3	0	3-4	5	4-5	5	5-6	10	6-9	10		
Zaman(sn)	Hız(km/h)															
0-3	0															
3-4	5															
4-5	5															
5-6	10															
6-9	10															
%100	0 km/s → 10 km/s	<p>Hız-Zaman</p> <p>Yatay eksen: Zaman(sn) (0-7)</p> <p>Dikey eksen: Hız(km/h) (0-14)</p> <table border="1"> <thead> <tr> <th>Zaman(sn)</th> <th>Hız(km/h)</th> </tr> </thead> <tbody> <tr><td>0-1</td><td>0</td></tr> <tr><td>1-2</td><td>5</td></tr> <tr><td>2-3</td><td>5</td></tr> <tr><td>3-4</td><td>13</td></tr> <tr><td>4-5</td><td>13</td></tr> <tr><td>5-7</td><td>14</td></tr> </tbody> </table>	Zaman(sn)	Hız(km/h)	0-1	0	1-2	5	2-3	5	3-4	13	4-5	13	5-7	14
Zaman(sn)	Hız(km/h)															
0-1	0															
1-2	5															
2-3	5															
3-4	13															
4-5	13															
5-7	14															

Aracımızın simülasyon ortamından farklı zamanlarda 10 km/s hızı çıktığını fark ettim. Yazılımsal olarak herhangi bir eksiklik gözlemlenmedi ancak aracın mekaniksel olarak simülasyon ortamındaki özellikleri bire bir uyumadığı için simülasyon ortamında aracımızın hızlanması daha kolay olduğunu gözlemledik.

Araca verdigimiz gaz değeri şuanlığı şasemizden kaynaklı sıkıntılarından dolayı tam olarak motorun çıkabileceği maksimum gaz değerini vermek yerine güvenlik açısından sıkıntı çıkarmayan bir gaz değeri ile sınırladık.

## Frenleme



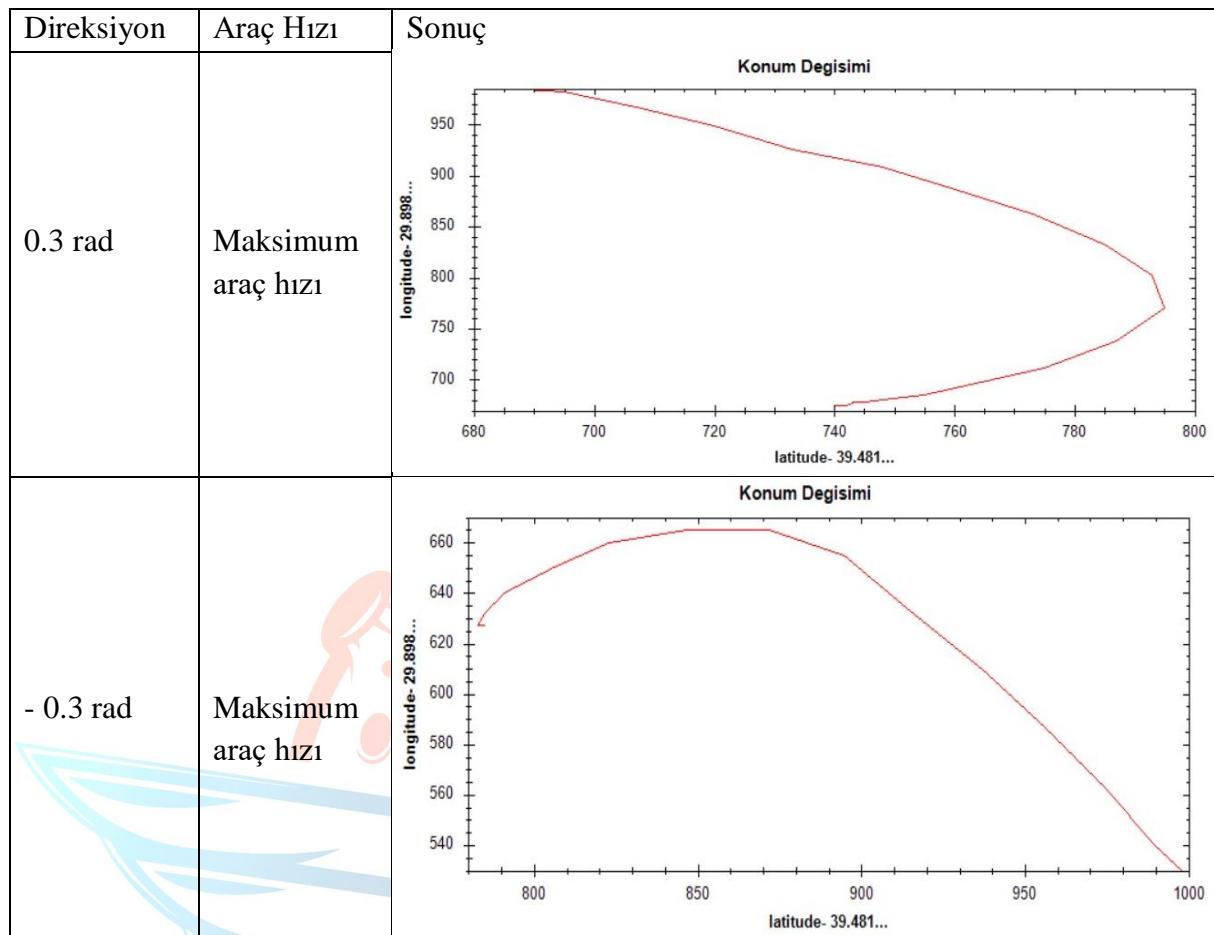
Fren pedalı ve fren ana merkezi arasında montaj sırasında pay bırakılmadığı için fren merkezi hidrolik sıvayı tek tarafa daha iyi iletti. Bu yüzden araç çift teker yerine tek tekerden frenlendi. Bir de fren motorunun 12 V ile beslemişti ancak 24 V ile beslediğimiz de daha hızlı hareket ettiğini tespit ettik.

Ulaştığımız sonuçlar doğrultusunda araç üzerinde mekaniksel ve elektriksel olarak düzenlemelere gidilecektir.

## Yönlendirme

Direksiyon	Araç Hızı	Sonuç
0.3 rad	5 km/s	<p>Konum Degisimi</p> <p>longitude-29.898... latitude- 39.481...</p> <p>This plot shows the position change over time for a steering angle of 0.3 rad. The x-axis represents latitude from 640 to 760, and the y-axis represents longitude from 420 to 540. The path starts at approximately (635, 425) and curves upwards and to the right, ending near (765, 535).</p>
- 0.3 rad	5 km/s	<p>Konum Degisimi</p> <p>longitude-29.898... latitude- 39.481...</p> <p>This plot shows the position change over time for a steering angle of -0.3 rad. The x-axis represents latitude from 660 to 760, and the y-axis represents longitude from 420 to 560. The path starts at approximately (665, 435) and curves upwards and to the left, ending near (765, 560).</p>

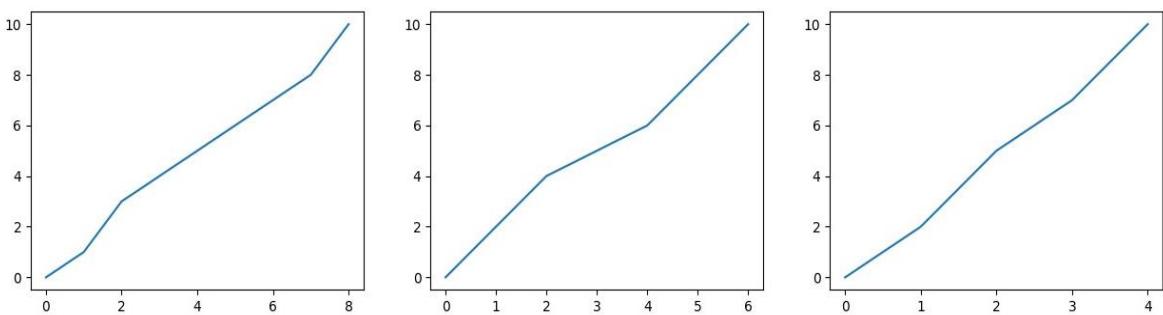
Direksiyon	Araç Hızı	Sonuç
Direksiyon Maksimum Açı Değeri	5 km/s	<p>Konum Degisimi</p> <p>longitude-29.898... latitude- 39.481...</p> <p>This plot shows the position change over time for the maximum steering angle. The x-axis represents latitude from 730 to 820, and the y-axis represents longitude from 500 to 560. The path starts at approximately (735, 505) and curves upwards and to the right, reaching a peak longitude of about 560 at latitude 795 before slightly decreasing.</p>
Direksiyon Maksimum Açı Değeri (Ters Yön)	5 km/s	<p>Konum Degisimi</p> <p>longitude-29.898... latitude- 39.481...</p> <p>This plot shows the position change over time for the maximum steering angle in the reverse direction. The x-axis represents latitude from 650 to 750, and the y-axis represents longitude from 440 to 560. The path starts at approximately (655, 445) and curves upwards and to the left, reaching a sharp turn at latitude 740 where the longitude increases from 520 to 560.</p>



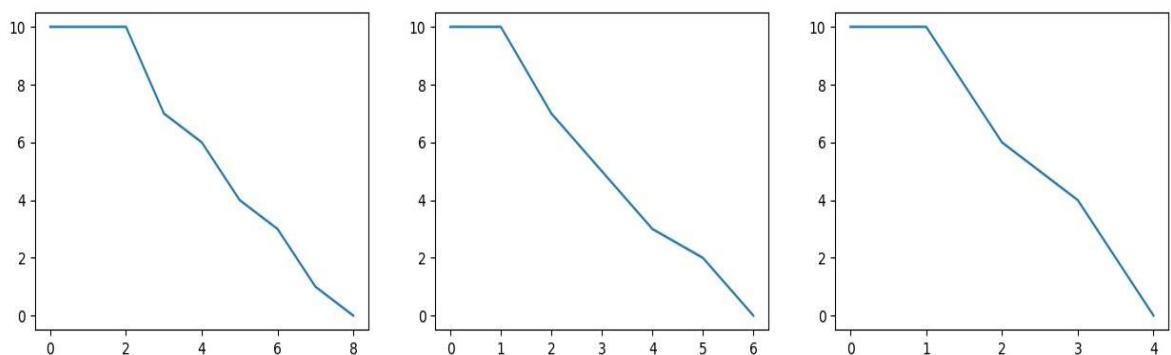
Salıncak sistemimizdeki wishbone ile deve boynu arasındaki rotinin yeterince sıkılmamasından kaynaklı teker açılarında istenilen değere ulaşılmadı. Mekaniksel olarak sorunun giderilmesi için çalışmalara başlanıldı.

Test sürüslerinde aldığımız konum bilgilerini anlık grafiğe dökerken de haberleşme sistemimizde oluşan mesaj kayıplarından ötürü konum grafiğinde bazı tutarsızlıklar oluştu.

Aynı zamandan istenilen girdilerde maksimum teker açıları şartını bahsettiğimiz mekaniksel aksaklıktan ötürü ön tekerlerde açı farkı olmuşmuştur.



Şekil 89 Simülasyon Ortamında Aaracın İstenilen Gaz Değerlerindeki İvmelenmesi



Şekil 90 Simülasyon Ortamında Aaracın İstenilen Fren Değerlerinde Duruşu

Simülasyon ortamındaki aracımızdan gelen hız verilerini grafiği dökebilirken yönlendirme için verilen test şartlarını gerçekleştirdik ancak konum verilerini alma konusunda yaşadığımız aksaklılıklardan dolayı yer değiştirme grafiğini oluşturamadık.

Simülasyon ortamında daha stabil verilere ulaştık. Bu farklılıkların sebebini aracımız üzerinde hem elektriksel hem de mekaniksel olarak kaynaklanan sebepleri test ederek iyileştirme çalışmalarına başladık.

## 8. Referanslar

<https://dronebotworkshop.com/getting-started-with-lidar/>

<https://www.youtube.com/watch?v=VhbFbxyOI1k&list=LL&index=2>

<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7893614>

[https://www.direnc.net/rplidar-a3m1-360-derece-lazer-tarayici-gelistirme-seti?gclid=Cj0KCQjwm6KUBhC3ARIsACIwxBgQGk8MBj8pdslj7sj1ISef1nHg7FTOd-kEg38t8CzB500PSjQurDUaAtiQEALw\\_wcB](https://www.direnc.net/rplidar-a3m1-360-derece-lazer-tarayici-gelistirme-seti?gclid=Cj0KCQjwm6KUBhC3ARIsACIwxBgQGk8MBj8pdslj7sj1ISef1nHg7FTOd-kEg38t8CzB500PSjQurDUaAtiQEALw_wcB)

<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=397896>

<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=61200>

<https://onlinelibrary.wiley.com/doi/pdf/10.1002/atr.5670430104>

<https://www.kgm.gov.tr/Sayfalar/KGM/SiteTr/Trafik/DurmaIntikal.aspx>

<https://otomobilteknoloji.blogspot.com/2016/09/frenleme-kuvveti-hesabi-ve-surtunme-katsayisi.html#:~:text=F%3D%20m.%20a%20form%C3%BCl%C3%BCne%20g%C3%BCre,frenleme%20kuvveti%20de%20fazla%20olacakt%C4%B1r.>

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8806152>

<http://nek.istanbul.edu.tr:4444/ekos/TEZ/29819.pdf>

[http://www.megep.meb.gov.tr/mte\\_program\\_modul/moduller\\_pdf/Fren%20Sistemleri.pdf](http://www.megep.meb.gov.tr/mte_program_modul/moduller_pdf/Fren%20Sistemleri.pdf)

<https://web.itu.edu.tr/temizv/Sunular/Frenler.pdf>

<https://depo.btu.edu.tr/dosyalar/makine/Dosyalar/Ta%c5%9fc4%b1t%20Fren%20Sistemleri%20Testi%20Deneyi.pdf>

<https://www.geeksforgeeks.org/yolo-you-only-look-once-real-time-object-detection/>

<https://neptune.ai/blog/data-augmentation-in-python>

<https://openaccess.firat.edu.tr/xmlui/bitstream/handle/11508/18394/524615.pdf?sequence=1>

[https://www.udemy.com/share/104m4y3@7b\\_EFuwVYLfyYTQYR61sjDbaypifcL1AwUdmquiQUDcBXo7XjqQyr0sMbZeIk9gh/](https://www.udemy.com/share/104m4y3@7b_EFuwVYLfyYTQYR61sjDbaypifcL1AwUdmquiQUDcBXo7XjqQyr0sMbZeIk9gh/)

<https://youtube.com/playlist?list=PLvpJkeZbByrMDjMsl3Irvus012pM3dMzB>

<https://www.geeksforgeeks.org/perspective-transformation-python-opencv/>