

# Teknofest Robotaksi Yarışması

-En basit olarak 2 kategoriden oluşan robotaksi yarışması bir özgün araç bir de hazır araç kısmında yarışılabilir.Özgün araç en az 5 kişi hazır araç en az 3 kişilik takımlardan oluşuyor.test videosu ve simülasyon raporu ve sunumu videosu sonuçlarına göre finale katılmaya hak olup olmadığı anlaşılır.

-3 farklı aşamadan oluşuyor yarışma TYK(teknik yeterlilik formu) KTR(Kritik Tasarım Raporu ve Araç test videosu & simülasyon raporu ve sunumu.

-referans noktası= parkur içinde önceden belirtilmiş bir gps koordinatıdır.

-araç başlangıç noktası = aracın görev başlangıcında bulunduğu gps koordinatıdır ve kuzey referansına göre kafa açısı bilgilerinden oluşur.

## 1. Otonom Sürüş Algoritmaları (Kontrol ve Takip)

### Pure Pursuit

-Şerit takibi, düşük hızda rota takibi. Basit ve hızlı uygulaması sayesinde sıkça tercih edilir.

#### **Avantaj:**

-Hesaplama maliyeti düşüktür.

-Hızlıdır.

#### **Dezavantaj:**

-Eğrilerde ve yüksek hızda kararsızlık gösterebilir.

-Takip noktası seçimi kritiktir.

### Stanley Controller

-Yüksek hızlı şerit takibi ve rota takibi önemli özelliklerindendir. Ön tekerlek direksiyon açısını, ön aks üzerindeki yol hatasına göre hesaplar.

#### **Avantaj:**

-Yüksek hızda daha kararlıdır

-Hatayı minimuma indirmede daha başarılıdır.

**Dezavantaj:**

-Başlangıç koşullarına duyarlı ve kalibrasyonu biraz daha zor oluyor genelde.

**PID (Oransal-İntegral-Türev)**

-Düşük seviye kontrol, duraklama, park gibi konularda hassas konumlandırmaya sahiptir.

**Avantaj:**

Basit ve güvenilirdir.

-Tepki süresi hızlıdır.

**Dezavantaj:**

-Tüm sürüş senaryolarına genellenemez.

-Her parametre için ayrı ayar gerektirir.

**MPC (Model Predictive Control)**

-Gelişmiş bir rota ve kontrole sahiptir. Özellikle karmaşık senaryolarda aracın gelecekteki durumunu tahmin ederek en iyi manevrayı belirler.

**Avantaj:**

Performansı yüksektir ve kısıtlamaları hesaba katarak en uygun yolu çizer.

**Dezavantaj:**

Yüksek hesaplama maliyeti, gerçek zamanlı gereksinimler için donanım optimizasyonu şarttır.

**ROS (Robot İşletim Sistemi)**

-Robot yazılımları geliştirmek için kullanılan, birbirleri ile iletişimde olan node'lar yaratarak robotik projeleri modüler hale getiren bir yazılım çatısıdır

**Avantaj:**

-Kod parçaları bağımsız olduğundan hata ayıklama daha kolaydır.

-Kütüphane havuzu geniştir.

-Birden fazla bilgisayarda çalışabilir.

## Dezavantaj:

-Yeni başlayanlar için karmaşıktır , geniş yer kaplar.

## 2. Algılama (Nesne Tespiti) Algoritmaları

Ekibinin görevi trafik konileri, yol işaretleri, trafik lambaları ve yayalar gibi çevredeki kritik nesneleri hızlı ve doğru bir şekilde tespit etmektir.

### a)YOLO Serisi (v5, v7, v8)

SxS'lik ızgaralara bölmektedir Her ızgara kendi içinde nesne olup olmadığını, varsa orta noktasının içinde olup olmadığını,orta noktası da içindeyse uzunluğunu, yüksekliğini ve hangi sınıftan olduğunu bulmaktadır.Tek bir ağ üzerinden olduğu için hızlıdır ayrıca gömülü sistemler için uygundur.Özellikle trafik lambası tespiti ve sınıflandırması için kritiktir.

### b)OpenCV

Renk alanı dönüşümü gürültü azaltma temel şerit çizgisi tespitinde kullanılır

### c)TensorRT

Donanım üzerinde maksimum hızda çalışması için önemlidir düşük gecikme süresi elde etmek için idealdir.

Kullanılan Yazılım Dilleri

### a)Python

-Bu dilin kullanılmasının sebebi geçtiğimiz yıllarda veri bilimi ve yapay zekâ gibi sektörlerde yoğunca kullanılması ve bu alanlarda çok miktarda doküman ve kaynak bulunmasıdır.Genel olarak daha basit bir dil ve kullanım alanı çok fazla hemen hemen her türlü platformda çalışır.

## 3. Yapay Zekâ Yaklaşımları

Otonom sürüş sistemini **Makine Öğrenmesi (Machine Learning)** temelli alt sistemlerle kurar ve genel karar mekanizmasını bir **Durum Makinesi (State Machine)** ile yönetir.

### a)Makine Öğrenmesi(Machine Learning)

Nesne sınıflandırma şerit yol segmentasyonu ve rota tahmininde kullanılır.

Zor olan işleri çözebilir insan hatasını azaltabilir değişen koşullara uyum sağlayabilir fakat büyük hacimli olması yüksek performanslı donanım ve sürekli kontrole ihtiyacı vardır.

#### b)Pekiştirmeli Öğrenme(Reinforcement Learning)

Park etme, yoğun trafikte şerit değiştirme gibi karmaşık işlerde idealdir.Genellikle simülasyon üzerinde eğitilerek politikaların öğretilmesinde kullanılır.

Uzun vadeli işlerde mükemmeldir değişen ortama uyum sağlama yeteneği yüksektir.Genel itibari ile maliyetli ve risklidir.

### 4. Sensörler

#### a)LiDAR

Lazer darbeleri ile nesne uzaklığını anlama nesne şekil tespiti ve haritalamaya yarayan teknolojidir. dikeyde 20 derece.

Lidar sensöründen alınan veriler filtrelenerek yapılan işlemler hızlandırılır.

#### **Avantaj:**

-Çok hassas veri üretmektedir.

#### **Dezavantaj:**

-Olumsuz hava koşullarında alınan verilerde gürültü artmaktadır.

-Pahalı bir sensördür.

#### b)Kamera

Görüntü işleme algoritmaları ile işlenir. LiDAR verisi ile füzyon yapılarak algılama doğruluğu artırılır.

#### **Avantaj:**

-Yapay zeka algoritmaları ile beraber çalışabilir bu sayede de levha okuyabilme,

trafik ışıklarını saptayabilme gibi özellikleri bulunmaktadır.

-Ucuz bir sensördür.

**Dezavantaj:**

-Menzili oldukça düşüktür.

-Hava koşullarından etkilenmektedir.

c)IMU

Konumunu yönünü ve hareketini belirlemek için kullanılan temel bir sensör setidir.

**Avantaj:**

-Çok daha hassas ve hızlı bir konumlama tahmini sağlar

d)Encoder

Aracın hızı, direksiyon dönüş açısı ve fren baskı derecesini hesaplamak için kullanılır.

**Avantaj:**

-çok yüksek hassasiyetle ve hızlı bir şekilde ölçebilir.

-daha ucuz ve araca entegrasyonu kolaydır.

**Dezavantaj:**

- Kaygan yüzeylerde veya engebeli arazide hatalı veri üretir.

- Aracın haritadaki mutlak konumunu bilemez.

5. Simülasyon Ortamı

a)Unity

Yüksek doğrulukta görsel simülasyon, Derin Öğrenme eğitimi için sentetik veri üretimi için kullanılan bir simülasyon ortamıdır.

**Avantaj:**

-Daha fotogerçekçi çıktılar üretir.

-Büyük miktarda sentetik veri üretimine imkan tanır.

b)OpenGL

-komutlarının doğrudan kullanılabileceği JAVA eko-sisteminde çalışan Processing kütüphanesi kullanılmaktadır.

**Avantaj:**

-Neredeyse tüm işletim sistemlerinde çalışır.

-Telifsiz ve açık bir standarttır.

-Çok uzun süredir var olduğu için geniş bir topluluk desteği, dokümantasyon ve örnekler mevcuttur.

**Dezavantaj:**

-Donanım detaylarını manuel olarak yönetmeyi gerektirir bu yüzden de başlangıç seviyesi için zorlu olabilir.