

TEKNOFEST

FESTIVAL FÜR LUFT- UND RAUMFAHRT UND

WETTBEWERB FÜR FAHRGASTFAHRZEUGE

VORENTWURF UND SIMULATIONSBERICHT

(URSPRÜNGLICHE FAHRZEUGKATEGORIE)

TEAM- UND FAHRZEUGNAME: AKROTEK / ŞUBAR

NAME DES BERATERS Selim SOYLU

TEAMKAPITÄN Şerafettin GÜVEN

TEAM ID: #53614

INHALT

1.	Zusammenfassung	3
2.	Team Organisation	3
	2.1 Team Fotos	4
	2.2 Team-Organigramm	6
	2.3 Zeitablaufplan	7
3.	Analyse der Wettbewerbsregeln und der Ziele der Designstudie	7
4.	Fahrzeug Spezifikationen	8
5.	Original Komponenten	9
6.	Sensoren	9
7.	Fahrzeug-Steuergerät	11
8.	Algorithmen für autonomes Fahren	12
	8.1. Modul zur Erkennung von Fahrspuren und Barrieren	12
	8.2. Modul zur Erkennung von Schildern/Verkehrsampeln	12
	8.3. Modul zur Erstellung von Routen	13
	8.4. Fahr-Algorithmus	13
	8.5. Parken Algorithmus	14
9.	Sicherheitsvorkehrungen	14
	9.1 Physikalisches Notfall-Reaktionssystem	14
	9.2 Ferngesteuertes Notrufsystem	14
	9.3 Überhitzung	15
	9.4 Besuchen Sie LIDAR Definition	15
	9.5 Sicherheit der Stromversorgung	15
	9.6 Schutz des Ladeeingangs	15
	9.7 Kabel-Layout	15
10.	Simulation	16
11.	Referenzen	17

TEKNOFEST
HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

1. Zusammenfassung

Als AKROTEK-Team haben wir uns für den Wettbewerb Robotaxi - Autonomous Passenger Car 2021 zusammengefunden. Obwohl wir zum ersten Mal an diesem Wettbewerb teilnehmen, achten wir darauf, dass wir uns in kurzer Zeit durch harte Arbeit eine ausreichende Ausrüstung aneignen und trotzdem weiterarbeiten. Wir geben unser Bestes für den Wettbewerb Robotaxi - Autonomous Passenger Car, indem wir diese Ausrüstung, die wir erworben haben, mit unserem alten Wissen kombinieren.

2021 Robotaksi - Von dem für den Wettbewerb Autonomous Vehicle konzipierten Fahrzeug wird erwartet, dass es die von der Generaldirektion für Autobahnen aufgestellten Schilder und Verkehrsregeln gemäß den Vorgaben des Wettbewerbs beachtet, Fahrgäste an den vorgesehenen Haltestellen auf der Strecke korrekt aufnimmt und absetzt und die Strecke in kurzer Zeit und korrekt zurücklegt, indem es den entsprechenden Schildern folgt. Seit der Gründung des Teams haben wir uns zum Ziel gesetzt, das gewünschte Problem zu lösen, indem wir die Fahrzeugtechnologien und Open-Source-Codes der in diesem Bereich tätigen Unternehmen untersuchen und die hier gewonnenen Kenntnisse und Erfahrungen für den Wettbewerb nutzen. Die Fähigkeiten, die wir zu diesem Zweck erworben haben, können wie folgt beschrieben werden:

a-Erstellung von echten und Vollzeit-Algorithmen.

b-Optimierung erstellten Algorithmen entsprechend dem zu verwendenden Prozessor.

c-Erstellung der notwendigen 3D-Konstruktion, die für die Erstellung der notwendigen Simulationsumgebung des autonomen Fahrzeugs erforderlich ist.

d- Nutzung der Bildverarbeitungsfunktionen, die für die Lesbarkeit der Beschilderung erforderlich sind.

e- Entwicklung von Prozessoren für industrielle Zwecke und Verwendung dieser entwickelten Prozessoren.

f- Verwendung von Kommunikationsprotokollen, die für industrielle Zwecke verwendet werden.

g- Projekt- und Ressourcenmanagement.

Die Simulationsumgebung für das Fahrzeug und die Algorithmen für das autonome Fahren wurden von unseren Freunden erstellt, die in unserem Team für Simulation und Software zuständig sind. Unsere Simulation wurde in Übereinstimmung mit den in der Spezifikation von Robotaksi - Autonomous Passenger Car geforderten Spezifikationen entworfen. Die übrigen Teile des Fahrzeugs sind in zwei Gruppen unterteilt, die mechanische und die elektrische, und wurden entsprechend der Spezifikation entworfen.

2. Team Organisation

Unser Team besteht aus vielen Schülern und Studenten und multidisziplinäre Struktur. Das Team ist entsprechend den Aufgaben Arbeitsplan in 4 Untergruppen aufgeteilt. Dies erleichterte die Weiterverfolgung und Aufteilung der Arbeitspakete.

Das Software-Team, das die Simulationsumgebung erstellt, das Fahrzeug simuliert und zeichnet, die erstellten Algorithmen in der Simulationsumgebung einrichtet, die für das Fahren erforderliche befahrbare Strecke, den Parkalgorithmus, eine Fernverbindung mit dem Fahrzeug herstellt

Es ist die Aufgabe dieses Teams, alle Daten, die während der Fahrt vom Fahrzeug kommen, zu speichern und Notfallsituationen an das Fahrzeug zu übermitteln, Algorithmen zur Erkennung von Hindernissen, Fußgängern, Ampeln und Schildern zu entwickeln, auf die das Fahrzeug stoßen könnte, und einen Straßenplan in Übereinstimmung mit den vom Fahrzeug erkannten Objekten zu erstellen.

Das Testteam ist das Team, das die Stellen testet, an denen das Fahrzeug in bestimmten Phasen kontrolliert werden muss, die Ergebnisse aufzeichnet und dann diese Aufzeichnungen vergleicht, um das Fahrzeug stabiler zu machen.

Das Elektronikteam ist für die Bereitstellung von Informationen darüber verantwortlich, wo und wie die Sensoren, Lidargeräte, Kabel usw., die wir im Fahrzeug verwenden werden, eingesetzt werden sollen, und stellt sicher, dass wir sie im Fahrzeug korrekt verwenden.

Das Mechanik-Team ist das Team, das die Konstruktion des Fahrgestells unseres Fahrzeugs gemäß den Spezifikationen durchführt, die mechanischen Teile montiert, die für die Bewegung des Fahrzeugs sorgen, und Pläne darüber erstellt, wo die Sensoren am Fahrzeug angebracht werden sollen.

2.1 Team Fotos



Selim SOYLU: Akademischer Berater.



Şerafettin GÜVEN Mannschaftskapitän

Er studiert an der Aksaray Universität Elektrotechnik und Elektronik in 4. In den vergangenen Jahren er an Modellflugzeugen, unbemannten autonomen Fahrzeugen und autonomen Systemen gearbeitet und sich mit Software-Sprachen wie Matlab, Javascript, HTML und C# beschäftigt. Derzeit arbeitet er weiter an Elektrofahrzeugen und autonomen Systemen und ist als Teamkapitän, Elektrik-Elektronik-Teil und Fahrzeugtester tätig.



Süleyman KAYA: Teammitglied

Heute ist er Cybersicherheitsforscher, Softwareentwickler, Python, C#, VB.Net, Bash und andere Sprachen. Er hat Erfahrung Bildverarbeitung, maschinellem Lernen und künstlicher Intelligenz. Er verfügt über mittlere Kenntnisse in der Linux/UNIX-Systemadministration und in Penetrationstests. Er hat eine Dankesnachricht von der ITU IT-Abteilung für die von ihm gemeldete Sicherheitslücke erhalten. Er interessiert sich auch für , Netzwerke und Sicherheit, Bug Bounty Hunting und die Sicherheit von Webanwendungen. Aksaray Hazım Kulak

der Anatolian High School. Er arbeitet als Bildverarbeitungsspezialist und Softwareentwickler in unserem Team.



Sinan KURTMAN Teammitglied

Zunächst arbeitete er an analogen elektronischen Systemen und entwickelte die Techniken zur Herstellung von Verstärkern, Radios und Speisekarten. Später stieg er in die digitale Elektronik ein und realisierte Logikschaltungen mit Transistoren und verwendete dann integrierte Logikschaltungen. Im Jahr 1991 baute er seine erste digitale Schaltung, die digitale Waage. Zurzeit ist er Student im 3. Jahr an der Aksaray Universität, Fakultät für Ingenieurwesen, Abteilung für Elektrotechnik und Elektronik. Er arbeitet als Kommunikationsbildverarbeiter und Softwareentwickler im Team.



Erdem CEYLAN Teammitglied

Mitglied, der an der Universität Düzce im Fachbereich Computertechnik studiert. arbeitet mit den Software-Sprachen C, C ++, C# und Python. Er studiert derzeit Advanced Arduino. interessiert sich für alle Arten von elektrischen und autonomen Geräten. Derzeit arbeitet er an IHA und autonomen Fahrzeugen. Er erfüllt seine Aufgabe als Softwareentwickler im Team.



Yunus Emre ERDEM: Teammitglied

Er begann bereits in der Sekundarschule, Projekte mit Arduino zu entwickeln. Heute geht er in die 11. Klasse der Ürgüp Tesan Science High School und kennt Fusion 360, Blender sowie die Sprachen Python und C#. Jetzt interessiert er sich für Cybersicherheit, autonome Fahrzeugtechnologien und Umweltenergietechnologien. Er arbeitet als Entwickler von Bildverarbeitungs- und Simulationsumgebungen in unserem Team.



Abdullah Mert KARAYAZILI: Teammitglied

Das Mitglied, das während seiner Sekundarschulzeit an Arduino-Projekten und 3D-Zeichenprogrammen wie Solidworks und SketchUp arbeitete, erhielt eine Ausbildung in Software-Sprachen wie Python und C, nachdem er auf die Abdulhamid Han Science High School wechselte, wo er jetzt ist. Heute arbeitet er weiter an der Sprache Python. Mit seiner Erfahrung aus verschiedenen Wettbewerben und Schulungen, an denen er teilgenommen hat, hat er die Rolle eines Mitglieds in diesem Team übernommen. Innerhalb des Teams ist er in der Abteilung für Mechanik und Fahrzeugtests tätig.

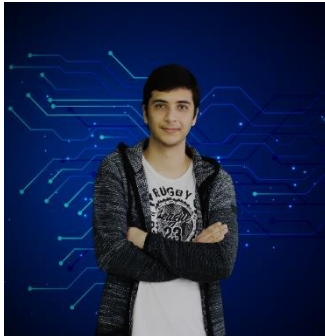


Salih ÜLGER Teammitglied

Er hat ein Zertifikat für Flutter und Go für Anfänger.

Er hat Zertifikate in den Bereichen Unreal Engine, C++ und Cybersicherheit.

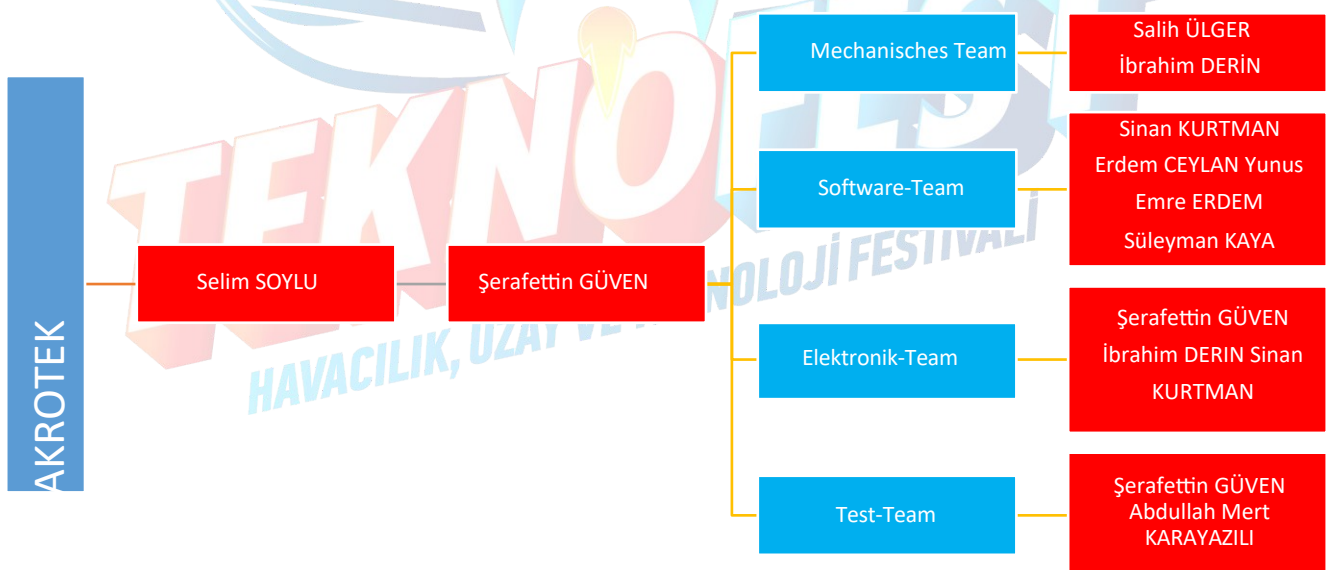
Er hat ein fortgeschrittenes Cisco Webex Python-Zertifikat mit internationaler Gültigkeit und ein fortgeschrittenes C#-Studium. Er besucht die 11. Klasse des Abdülhamid Han Wissenschaftsgymnasiums. Er ist im Team für das Design des Fahrzeugchassis zuständig.



İbrahim DERİN Teammitglied

Seit 2018 studiert er an der Aksaray Universität und arbeitet an C-Programmierung, Arduino, Autocad 2D und 3D. Er spricht auch Englisch auf Niveau B. Zurzeit er Student im 3. Studienjahr in der Abteilung für elektrische Elektronik der Universität, wo er derzeit Python und Android Studio. Heute beschäftigt er sich mit autonomen Fahrzeugen und Umweltenergietechnologien. Innerhalb des Teams übernimmt er die Aufgabe der Montage.

2.2 Teamorganisation Diagramm



2.3 Zeitfluss Zeitplan

ARBEITSZEITPLAN													
ARBEITSPAKETE Und AKTIVITÄTEN		Startdatum	Enddatum	Dauer (Woche)	November	Dezember	Januar	Februar	Mart	April	Mai	Juni	Juli
1.	Konzept und Entwurf Entwurf	14.11.20	16.04.21	15									
1.1	Literaturübersicht			6									
1.2	Konzeptioneller Entwurf			4									
1.3	Entwurf eines Designs			4									
1.4	Simulation			6									
2.	Detaillierter Entwurf und Materialauswahl	16.04.21	17.05.21	6									
2.1	Detaillierter Entwurf			1									
2.2	Materialauswahl			1									
2.3	Erstellen des Entwurfs			2									
3.	Prototypenherstellung und Materialmontage	17.05.21	17.06.21	4									
3.1	Herstellung und Montage von Prototypen			2									
3.2	Elektrische und elektronische Montage			1									
3.3	Installation von Kamera, Sensor und Lidar			1									
4.	Prototyp und Systemtests	17.06.21	31.07.21	8									
4.1	Fahrzeug-Fahrwerkstest			1									
4.2	Sensor Test			1									
4.3	Autonomes Fahren Test			2									
4.4	Autonomes Parken Test			1									
4.5	Kommunikationstest			1									

3. Analyse der Wettbewerbsregeln und Design Studienziele

Der Algorithmus für autonomes Fahren, der während des Wettbewerbs verwendet und entwickelt wird

- Beförderung von Passagieren gemäß den Regeln des Gewinnspiels
- Das Herunterladen von Passagieren gemäß den im Gewinnspiel festgelegten Regeln
- Richtiges Erreichen des Parkplatzes auf der Strecke
- Parken gemäß den Wettbewerbsregeln
- Genaueres Befolgen der auf dem Track erstellten Route

Es wird erwartet, dass es jede seiner Aufgaben ordnungsgemäß erfüllt. Da das Fahrzeug autonom fahren wird, ist eine Kommunikation zwischen Sensoren und ROS erforderlich.

Das Fahrzeug muss so viel wiegen wie nötig, so viel Energie aufwenden wie nötig und so viel Last auf die Räder bringen wie nötig. Die Räder müssen sich in ausreichendem Maße drehen und mit der erforderlichen Winkelgeschwindigkeit ausrichten lassen. Wenn das Fahrzeug diese Eigenschaften aufweist, verfügt es über die für den Wettbewerb erforderliche Mobilität und ist in der Lage, die im Wettbewerb gestellten Aufgaben rechtzeitig zu erfüllen.

Die für das Fahrzeug spezifizierten Spezi­alsensoren müssen für die zu erfüllenden Aufgaben nützlich sein. Mit den verwendeten Sensoren sollten die Fahrspuren auf der Wettkampfstrecke sehr genau erkannt werden und die Sensordaten im erforderlichen Frequenzbereich an das ROS übertragen werden. Der entworfene Algorithmus sollte die Standorte der Schilder und Ampeln auf der Rennstrecke entsprechend der Position des Fahrzeugs finden. Die Sensoren sollten die Schilder, Ampeln und Fahrspuren entsprechend der Position des Fahrzeugs in einem Zentrum platzieren. Damit die Sensorkombination ordnungsgemäß abgeschlossen werden kann, muss die Lage der verwendeten Sensoren zueinander sehr genau bestimmt und platziert werden. Damit das äußere Erscheinungsbild des Fahrzeugs einwandfrei ist und die Verkehrsregeln auf das Fahrzeug angewendet werden können, muss die Platzierung der verwendeten Sensoren akribisch und richtig erfolgen und die Verkabelung sorgfältig durchgeführt werden.

4. Fahrzeug Spezifikationen

Die Stromversorgung des Fahrzeugs besteht aus 5 12V 20AH Batteriegruppen, die eine Gesamtspannung von 60V liefern. Zwei dreiphasige BLDC HUB-Motoren mit 1000W Leistung werden für den Antrieb des Fahrzeugs verwendet. Die Nabenmotoren werden von zwei dreiphasigen 1000W-Drehzahlreglern angetrieben. Die Geschwindigkeitssteuerung erfolgt über die STM32duino-Entwicklungskarte, die wir im Fahrzeug verwenden werden. Die STM32duino-Hilfskarte sorgt für die serielle Kommunikation mit der NVIDIA Jetson Nano-Karte, die wir als Hauptcomputer bevorzugen. Sie überträgt die Sensordaten, die sie sammelt, an den Hauptcomputer und erfüllt die Steuerungsanforderungen des Hauptcomputers.

Als Lenksystem im Fahrzeug wird ein fertiges elektrisches Lenksystem (Steer by Wire) verwendet. Das Lenksystem wird um einen Encoder erweitert, um Rücklaufinformationen zu erhalten. Die Informationen des Encoders werden an die STM32duino-Hilfskarte übertragen und der Drehvorgang wird innerhalb bestimmter Grenzen ausgeführt.

Das Batteriemanagementsystem (BMS) wird eingesetzt ein ausgewogenes Laden und Entladen der Batterien zu gewährleisten. Der Gesundheitszustand der Batterien wird auf dem Armaturenbrett und per Fernbedienung überwacht. Es wird ein Stromverteilungssystem (PDS) verwendet, da das Fahrzeug verschiedene Spannungen benötigt, z. B. 60 V für die Antriebsmotoren, 24 V für die elektrischen Kolben der Hecktüren, 12 V für das Beleuchtungssystem und 12 V für die Servolenkung.

5. Original Komponenten

Das Design des Fahrzeugs, das in dem Projekt verwendet werden soll, wird in einer virtuellen Umgebung in 3D entworfen, völlig unabhängig von anderen Markenfahrzeugen auf dem Markt. Während der Herstellungsphase wird das Scheinwerfersystem des Fahrzeugs mit einem 3D-Drucker gedruckt. Die vorderen Layout-Akzente des Fahrzeugs werden überarbeitet, damit das Fahrzeug die notwendigen Manöver im Wettbewerbsbereich durchführen kann. Die äußere Karosserie des Fahrzeugs soll mit Fiberglas auf Decota verkleidet werden.

Der Schaltkreis des Batteriemanagementsystems (BMS) für das gesunde Entladen und Laden der Fahrzeugbatterie, das Stromverteilungsmodul, das die Energie aus der Batterie weiterleitet, indem es sie auf die von den erforderlichen Einheiten benötigten Spannungen reduziert, das Lenkungssteuergerät, das Rolle bei der Lenkung des Fahrzeugs spielt, usw. Die Schaltkreise werden in der virtuellen Umgebung entworfen und hergestellt und dann in das Fahrzeug eingebaut.

Für die Erkennung von Verkehrsschildern werden Techniken des maschinellen Lernens mit Hilfe von neuronalen Netzwerken über die Bibliotheken Tensorflow und/oder pytorch angewendet. Das maschinelle Lernen, das wir entwickeln werden, ist überwachtes Lernen und der Datensatz wird in Übereinstimmung mit den Standards der Generaldirektion für Autobahnen und Verkehrsschilder vorbereitet.

6. Sensoren

Die USB-Kameras werden für die Bildverarbeitung des Fahrzeugs verwendet. Das Ergebnis unserer Vorstudien ist, dass zwei oder drei Kameras verwendet werden sollen. Das Sichtfeld der Kameras die Fahrspuren auf der Straße und die Verkehrsschilder am Straßenrand umfassen. Da der NVIDIA Jetson Host-Computer FPGA-basierte Verarbeitung durchführt, kann er viel schnellere Ergebnisse liefern als Raspberry Pi-Prozessoren. Auf diesem Computer wird das Betriebssystem Ubuntu installiert und für die Programmierung die Sprache Python bevorzugt. Die Objekterkennungsalgorithmen werden mit Open CV oder YOLO v4 installiert, um das Fahrzeug zu steuern und zu kontrollieren. Die Kommunikation zwischen den Sensoren und Aktivatoren erfolgt über das ROS-System des Fahrzeugs. Die Kameras, die wir im Fahrzeug verwenden werden, sind in Abbildung 1 dargestellt.



Bild-1: 8 MP Kamera mit USB-Anschluss

Das DF Robot SEN0253 Sensor-Kit wird verwendet, um die Stabilität und Kontrolle im Steuerungssystem des Fahrzeugs zu erhöhen. Die Kompassinformationen, die dieser Sensor liefert, sind besonders nützlich für die Steuerung des Fahrzeugs.



Abbildung-2: DF Robot SEN0253 Sensor-Kit

Ein 2D-LIDAR-System wird verwendet, um Objekte oder Hindernisse in der Umgebung zu erkennen. Die Daten dieses Sensors werden insbesondere für die Erkennung der Straßenrichtung verwendet.



Abbildung 3: RPLIDAR A2M6 - 360-Grad-Laserscanner



Abbildung 4: HB100 Radar mit 20 m Reichweite

Außerdem wird ein Radargerät eingesetzt, um Hindernisse oder Personen zu erkennen, die plötzlich in einem Umkreis von 20 Metern vor dem Fahrzeug auftauchen könnten.

7. Fahrzeug-Steuergerät



Abbildung 5: NVIDIA Jetson Nano Mini-Computer

In autonomen Fahrzeugen muss die Bildverarbeitung sehr schnell und genau erfolgen und die Ergebnisse müssen korrekt interpretiert werden, um schnelle und vollständige Reaktionen zu gewährleisten. Zu diesem Zweck wurde der NVIDIA Jetson Nano als Hauptrechner in unserem Projekt bevorzugt.



Abbildung-6: STM32duino Mikrocontroller-Platine

Das STM32duino-Entwicklungsboard wurde bevorzugt, weil es schneller arbeitet als andere Arduino-Boards und FreeRTOS- und CMSIS-RTOS-Unterstützung bietet. Außerdem können durch die Installation des Maple Mini Bootloaders auf diesem Board Programme direkt über den USB-Anschluss geladen werden, ohne dass ein separater Programmlader erforderlich ist. Dieses Board, das über eine umfangreiche Bibliotheksunterstützung verfügt und mit 72 MIPS und 32 Bit eine sehr hohe Verarbeitungsleistung bei 72 MHz aufweist, enthält auch viele periphere Kommunikationseinheiten. Die ADC-Auflösung und die PWM-Auflösung sind ebenfalls hoch.

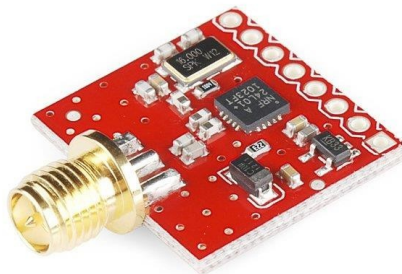


Abbildung-6: nRF24L01 RF-Modul

Obwohl das Fahrzeug eine autonome Steuerung bietet, kann es bei Bedarf auch aus der Ferne gesteuert werden. Zu diesem Zweck wird eine nRF24L01 Kommunikationskarte mit einer Reichweite von 1 km als RF-Karte verwendet.

8. Algorithmen für autonomes Fahren

Die folgenden Algorithmen zum autonomen Fahren wurden vom AKROTEK-Team im Rahmen des Wettbewerbs entwickelt.

Modul zur Erkennung von
Fahrspuren/Barrieren Modul zur
Erkennung von Schildern/Ampeln
Modul zur Routengenerierung
Fahr-Algorithmus Park-
Algorithmus

8.1. Fahrspur- und Schrankenerkennung Modul

Der Algorithmus für die Spurverfolgung, eine der wichtigsten Komponenten, spielt eine große Rolle bei der erfolgreichen Umsetzung des Spurverfolgungsprozesses und bei der Erstellung der richtigen Route. Während es einfach ist, den Spurfolgeprozess in der Simulation zu realisieren, ist dieser Prozess im wirklichen Leben aufgrund externer Faktoren recht schwierig. Ein Beispiel für das Verfolgen der Fahrspur durch Erkennung von Hindernissen und Grenzen im Simulator finden Sie in Abbildung 7

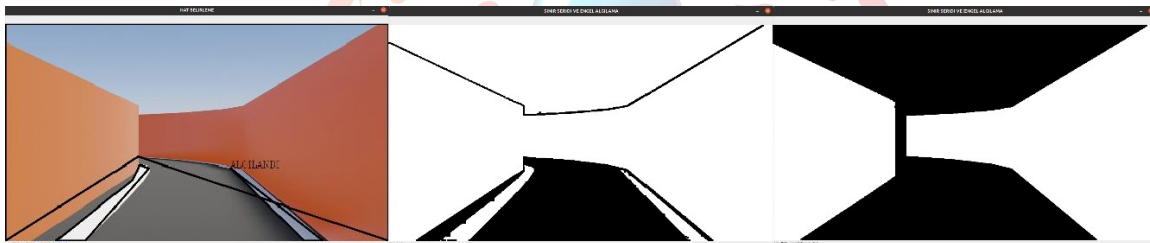


Bild-7

Unser Team will sicherstellen, dass die Algorithmen und Methoden, die in dem zu entwickelnden autonomen Fahrzeug verwendet werden sollen, unter realen Bedingungen erfolgreich funktionieren.

8.2. Schilder-/Verkehrssampel-Erkennung Modul

In diesem Modul wurde die Computer Vision effektiv eingesetzt. Während die Methode der Merkmalsextraktion zur Ermittlung des Inhalts des 3D-Objekts (Merkmalsextraktionsmethode) der auf der Kamera erkannten Schilder und Ampeln erfolgreich funktioniert, erhalten wir manchmal falsche Ergebnisse. Aus diesem Grund wird maschinelles Lernen für die Erkennung von 3D-Objekten mit Hilfe der Tensorflow Object Detection API und der YOLO-Algorithmen eingesetzt, so dass die auf der Kamera erkannten Objekte mit hoher Genauigkeit erkannt werden. Beispiele finden Sie in den Abbildungen 1, 2 und 3.



Bild-8

8.3. Erstellung von Routen Modul

Unser Fahrzeug erstellt seine Route, indem es Statusprüfungen durchführt.

Ereignisse auf der Route:

- Nicht rechts abbiegen
- Nicht links abbiegen
- Kein Weg nach vorn

Ereignisse im Zusammenhang mit der Fahrzeuggeschwindigkeit:

- Geschwindigkeitslimit einstellen
- Temporäres Tempolimit festlegen

Teller Ereignisse:

- 1 - Route ohne Zugang: Aktiviert die entsprechenden Einheiten aus den Routenereignissen.
- 2 - Vorwärts und links erzwungene Richtung: Aktiviert die entsprechenden Einheiten aus den Routenereignissen.
- 3 - Vorwärts und rechts erzwungene Richtung: Aktiviert die entsprechenden Einheiten aus den Routenereignissen.
- 4 - Erzwungene Richtung von vorwärts nach links: Aktiviert die entsprechenden Einheiten aus den Routenereignissen.
- 4 - Erzwungene Richtung von vorwärts nach rechts: Aktiviert die entsprechenden Einheiten aus den Routenereignissen.
- 5 - Geschwindigkeitsbeschränkung Ende 20km/h: Führt den Befehl Geschwindigkeitslimit auf 20 setzen aus Fahrzeuggeschwindigkeitsereignissen aus
- 6 - Die Höchstgeschwindigkeit beträgt 20km/h: Führt den Befehl Geschwindigkeitslimit auf 20 setzen aus Fahrzeuggeschwindigkeitsereignissen aus
- 7 - Die Höchstgeschwindigkeit beträgt 30km/h: Führt die Funktion Geschwindigkeitslimit auf 30 setzen aus den Ereignissen zur Fahrzeuggeschwindigkeit aus
- 8 - Straße für den Verkehr gesperrt: Aktiviert die entsprechenden Einheiten aus den Routenereignissen.
- 9 - Nicht links abbiegen: Aktiviert die Bedingung Kein Linksabbiegen für Routenereignisse.
- 10 - Nicht rechts abbiegen: Aktiviert die Bedingung Kein Rechtsabbiegen aus Routenereignissen.
- 11 - Parken verboten: Aktiviert die Parkverbotsbedingung von Routenereignissen.
- 12 - Parkbereich: Aktiviert die Bedingung Kein Objekt in der Nähe oder Parkbar aus den Routenereignissen.
- 13 - Anhalten: Führt den Befehl Temporäres Geschwindigkeitslimit auf 0 setzen aus Fahrzeuggeschwindigkeitsereignissen aus.
- 14 - Anhalten: Wenn der Abstand des Schildes zu den Geschwindigkeitsereignissen kleiner als x ist -> Temporäres Tempolimit auf 0 setzen und 35 Sekunden warten -> Maximales Tempolimit auf 20 setzen.
- 15 - Rote : Wenn der Abstand des Schildes zu den Geschwindigkeitsereignissen kleiner als x ist -> Temporäre Geschwindigkeitsbegrenzung auf 0 setzen und Grün gesehen -> Maximale Geschwindigkeitsbegrenzung auf 20 setzen.
- 16 - Gelbe Ampel: Ineffektiv
- 17 - Grüne : Führt den Befehl Maximalgeschwindigkeit auf 20 setzen von Fahrzeuggeschwindigkeitsereignissen aus.

8.4. Fahren Algorithmus

Die Daten, die durch die Integration der Informationen des Moduls zur Erkennung von Fahrspuren/Barrieren, des Moduls zur Erkennung von Schildern/Ampeln und des Moduls zur Routengenerierung gesammelt wurden, werden als Eingabe in den Fahralgorithmus verwendet und das Fahrzeug wird in die Lage versetzt, eine Aktion unter Berücksichtigung der Situationen/Bedingungen durchzuführen. Die Aufgabe dieses Algorithmus ist es, aus den Daten der anderen Module eine Reihe von

Bedingungen zu schaffen und das Fahrzeug entsprechend dieser Bedingungen erfolgreich zu bewegen.



Bild-9

8.5. Park Algorithmus

Er prüft die notwendigen Bedingungen (Parkschild erkannt, kein Parkschild erkannt, ob der Bereich, in dem sich das Parkschild befindet, verfügbar ist) mit Hilfe des Fahralgorithmus und soll die Parkaktion durchführen, sobald er die Information erhält, dass er zum Parken verfügbar ist.

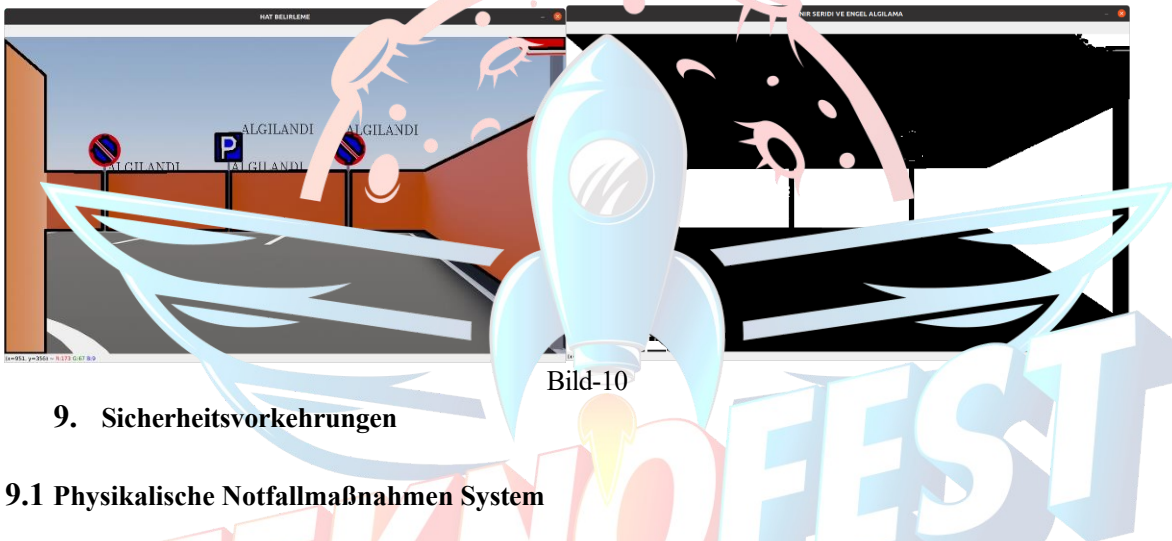


Bild-10

9. Sicherheitsvorkehrungen

9.1 Physikalische Notfallmaßnahmen System



Bild-11

Es gibt 1 "Notrufknopf" am Fahrzeug, wie in Abbildung 11 , so dass von außen eingegriffen werden kann. Dieser Knopf ist in Reihe mit der Hauptstromleitung unseres Fahrzeugs verbunden. Im Notfall wird der gesamte Strom des Fahrzeugs durch Drücken des Knopfes abgeschaltet.

9.2 Ferngesteuerte Notfallhilfe System

Durch Drücken der "Emergency Intervention Key" auf der Fernbedienung, mit der das Fahrzeug auch ferngesteuert werden kann, wird der Strom des Fahrzeugs abgeschaltet und das Fahrzeug in Sicherheit gebracht.

9.3 Überhitzung

Wenn der Motor bei laufendem Fahrzeug stärker als normal überhitzt, bremst der Motor ab, hält das Fahrzeug an und schaltet sich aus, wenn das Fahrzeug in Bewegung ist. Wenn das Fahrzeug nicht in Bewegung ist, schaltet sich der Motor ab. Der Motor kann erst dann wieder gestartet werden, wenn er abgekühlt ist.

9.4 Stopp durch LIDAR Definition

Wenn ein externer Faktor plötzlich vor dem Fahrzeug auftaucht, während es sich bewegt, wird dieser Faktor von der LIDAR-Kamera erkannt. Dann wird die Stromzufuhr zu den Motoren unterbrochen und das Fahrzeug hält durch plötzliches Abbremsen an. Wenn die Straße wieder sicher und frei ist, setzt das Fahrzeug seine Fahrt wieder fort.

9.5 Stromversorgung Sicherheit

Die Stromversorgung ist mit einer nicht brennbaren und stoßfesten Hülle gegen alle Arten von Stößen, Flüssigkeiten und Feuer geschützt, und falls sie erreicht werden muss, kann die Hülle geöffnet und eingegriffen werden. Es wird an einer zugänglichen Stelle im Fahrzeug mit einem Rahmen befestigt und gegen Verschieben gesichert.

9.6 Ladeeingang Schutz

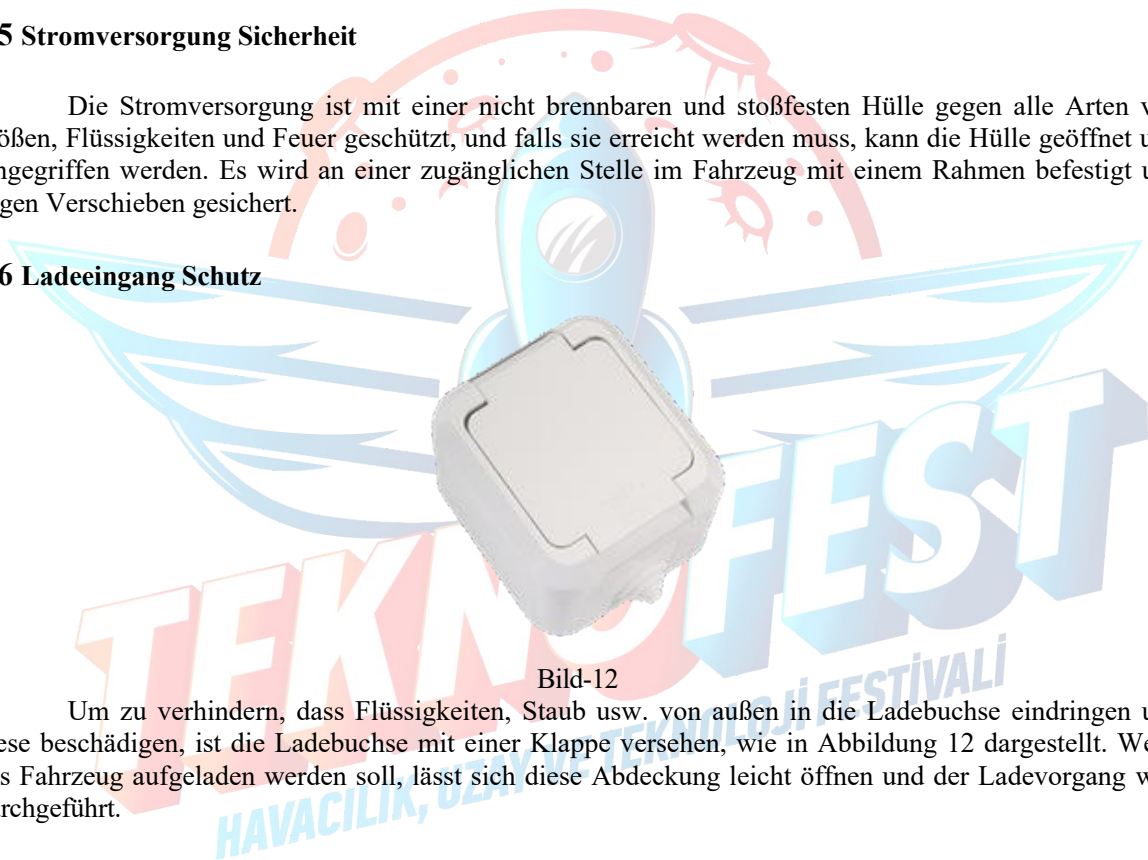


Bild-12

Um zu verhindern, dass Flüssigkeiten, Staub usw. von außen in die Ladebuchse eindringen und diese beschädigen, ist die Ladebuchse mit einer Klappe versehen, wie in Abbildung 12 dargestellt. Wenn das Fahrzeug aufgeladen werden soll, lässt sich diese Abdeckung leicht öffnen und der Ladevorgang wird durchgeführt.

9.7 Kabel Layout



Bild-13



Bild-14

Um sowohl im Fahrzeug als auch im Fahrzeug selbst keine Verwirrung zu stiften, werden die Kabel mit den in Abbildung-13 gezeigten Kunststoffklemmen miteinander verbunden und an der Ausbreitung gehindert. Anschließend wird der Kontakt mit den Kabeln verhindert, indem das in Abbildung 14 gezeigte Spiralrohr einen leichten Eingriff von außen verhindert.

Es werden Maßnahmen für mögliche Gefahrensituationen während der Testphase und während des Wettbewerbs festgelegt und Informationen über die dafür vorgesehenen Systeme bereitgestellt.

10. Simulation

Die Simulation wurde durch Zeichnen in Blender, einem 3D-Modellierungsprogramm, erstellt. In diesem Programm wurde der Wettbewerbsbereich als Bild-16 modelliert, indem die Strecke in den Spezifikationen als Referenz zu Bild-15 genommen wurde.

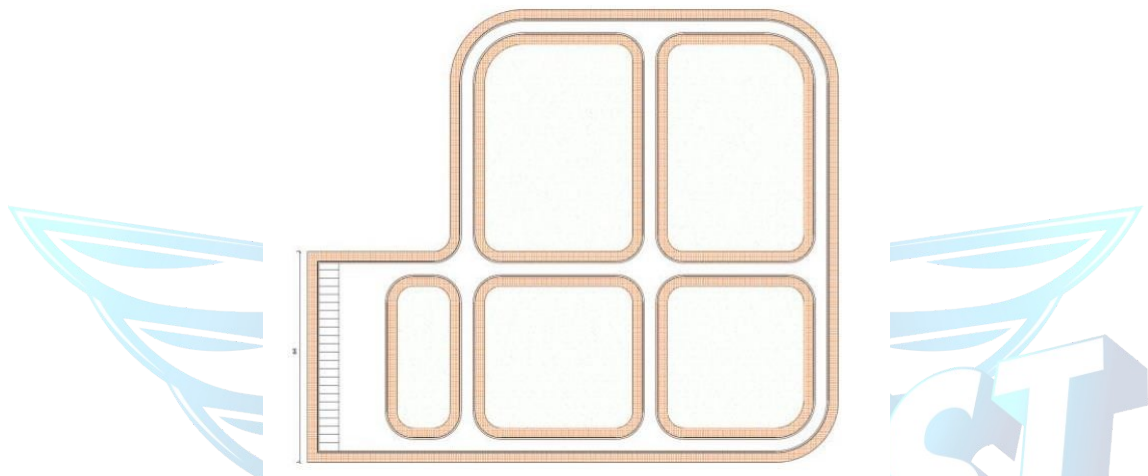


Bild-15 Bild in der Wettbewerbsbeschreibung

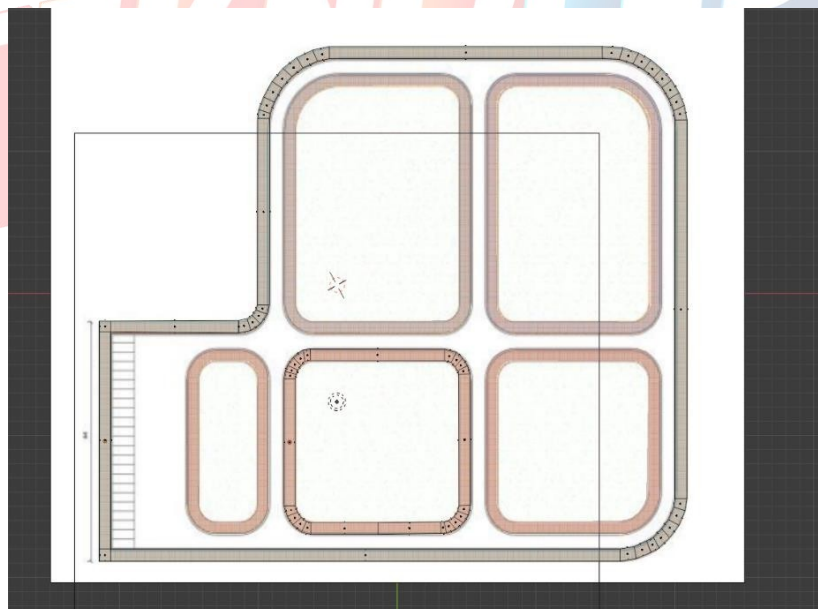


Abbildung 16 Zeichnen des Tracks im Programm Blender

Nachdem die Strecke modelliert war, wurden die von der Generaldirektion für Autobahnen vorbereiteten Schilder modelliert und gezeichnet, wie in Abbildung 17 dargestellt.

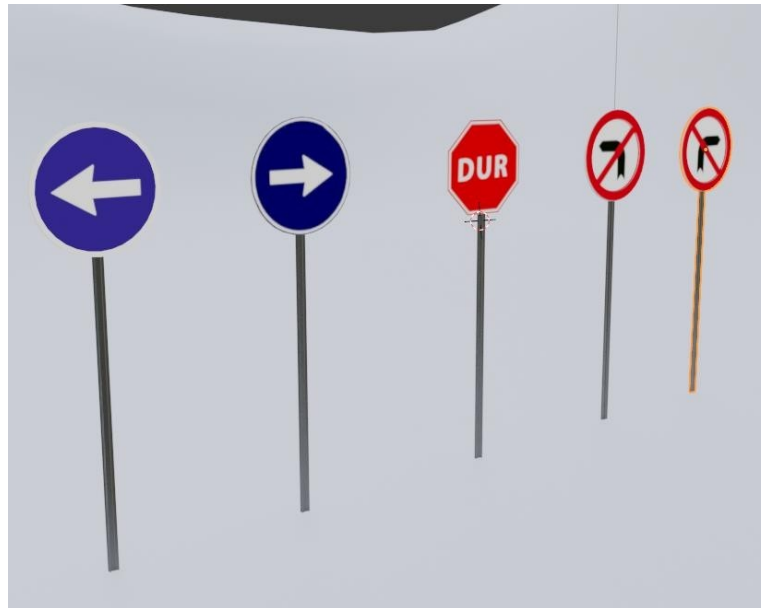


Bild-17 Beschilderungsbilder

Die gezeichneten Schilder werden entsprechend ihrer Position auf der Strecke platziert. Sie können unser Simulationsvideo über den Link <https://www.youtube.com/watch?v=YVZOEQhnmM> aufrufen. Unser Fahrzeug auf der Strecke wird als simulierte Animation gezeigt. Unsere Arbeit für autonomes Fahren geht weiter.

11. Referenzen

- <https://www.blender.org/>
- <https://www.ros.org/>
- https://en.wikipedia.org/wiki/CAN_bus
- <https://arxiv.org/pdf/1804.02767.pdf>
- <https://www.kgm.gov.tr/Sayfalar/KGM/SiteTr/Trafik/TehlikeUyarilsaretleri.aspx>
- J. Redmon, A. Farhadi, "Yolov3: An Incremental Improvement", arXiv:1804.02767, 2018.
- <https://dergipark.org.tr/en/download/article-file/126158> (G. Küçükyıldız, H. Ocak, "Development and Optimization of A Dsp-Based Real-Time Lane Detection Algorithm on A Mobile Platform", Turk. J. Elec. Eng. & Comp Sci., vol. 22, no. 6, pp. 1484-1500, 2014.)
- V. Gaikwad und S. Lokhande, "Fahrspurerkennung für fortschrittliche Fahrerassistenz", IEEE Trans. Intell. Transp. Syst., vol. 16, Nr. 2, S. 910-918, 2015.