

DRIVERLESS

Inhalt

DRIVERLESS 1

 KISS 2

 Software im Fahrzeug 2

 Hardware im Fahrzeug 3

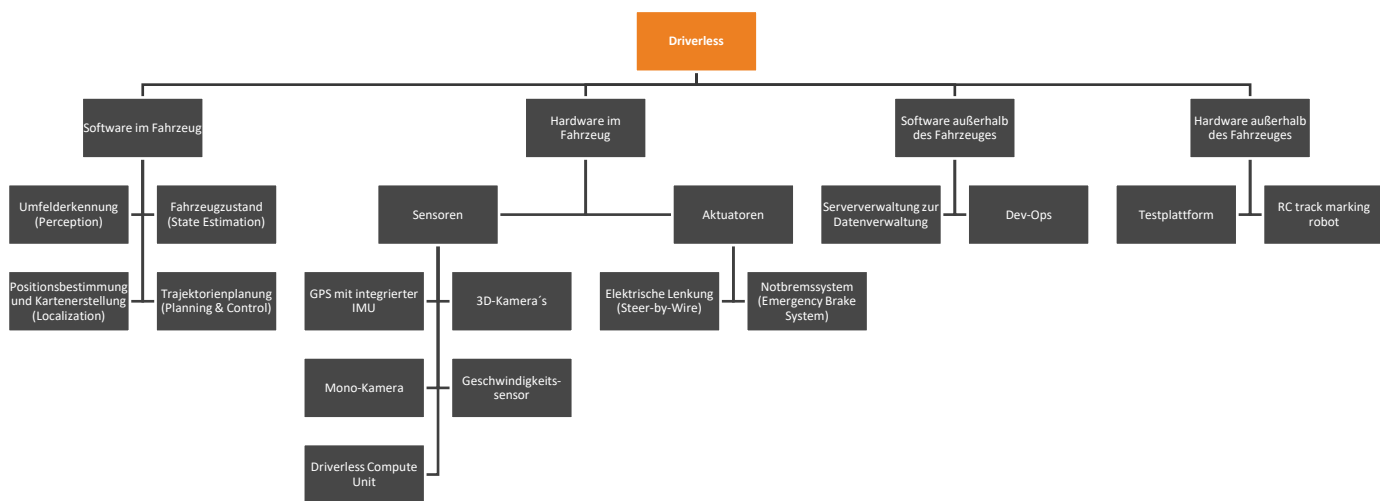
 Software außerhalb des Fahrzeuges 3

 Hardware außerhalb des Fahrzeuges 3

 Langfristiger Zeitplan 4

 Was du hier lernen kannst 4

 Team & Struktur 4



KISS

Die Entwicklung eines autonomen Fahrzeuges ist komplex und vielschichtig. Mit KISS für „Keep it simple, stupid“ haben wir ein Leitmotiv etabliert, mit dem wir erfolgreich das erste autonome Fahrzeug entwickeln möchten.



Unser Ziel ist eine...

- ❖ ... möglichst einfache Lösung.
- ❖ ... möglichst einfache Lösung um möglichst viele Teammitglieder unabhängig von ihrem Studiengang zu integrieren.
- ❖ ... möglichst einfache Lösung mit einem Minimum an Komplexität. Wir erhoffen uns hiervon eine Gewährleistung des Wissenstransfers, die einfache Integration neuer Mitglieder als auch kurze Entwicklungszyklen.
- ❖ ... möglichst einfache Lösung für den Start die nur für die Disziplinen Acceleration und SkidPad ausgerichtet ist und auf unnötige features verzichtet. Es soll jedoch so konzipiert sein, dass es für die Zukunft ein skalierbares Fundament darstellt.
- ❖ ... angemessen sichere und zuverlässige Lösung, die robust mit Fehlern in einzelnen Modulen sich verhält.

Mit KISS haben wir auch an der europaweiten ZF Driverless Challenge teilgenommen und die Endrunde der top-5 erreicht. Die Auswahl der Plätze erfolgte im Juni 2021. Wir freuen uns darauf!

SOFTWARE IM FAHRZEUG

Die Software im Rennwagen basiert auf dem Robotic Operating System (ROS) in Linux und besteht aus vier großen Teilbereichen:

Der Bereich **Perception** umfasst alle Themenbereiche der Umfelderkennung, d.h. dem Erkennen von verschiedenfarbigen Leitkegeln in unterschiedlicher Größe. Mehrere (3D)-Kameras und ein LiDAR detektieren die Kegel durch ein neuronales Netz das mit über 100.000 Kegelbildern trainiert wird. Über die Position von Keypoints im Bild, der Kegelgröße im Bild, der Street-Camera-Pose und Tiefendaten der Sensoren wird die Position der Kegel inklusive einer Ungenauigkeit geschätzt. Die Position der Kegel kann aus LiDAR-Daten durch klassische Algorithmen (RANSAC, Clustering) bestimmt werden.

Das Ziel des Bereichs **State Estimation** ist den Zustand des Autos zu schätzen. Sensoren wie ein GPS mit integrierter IMU, ein optischer Geschwindigkeitssensor, Radgeschwindigkeitssensoren und Lenkwinkelsensoren erfassen hierzu verschiedene Messwerte, die zusammengeführt Größen der (lokalen) Position und Richtung, Geschwindigkeit, Gierrate und das Schlupfverhältnis der Reifen schätzen. Zu den Werkzeugen im Bereich State Estimation gehören Kalman-Filter und ein einfaches dynamisches Modell für das Auto.

Die Softwarekomponente für **Localization** verarbeitet die erkannten Verkehrskegel (Perception) und den Zustand des Autos (State Estimation). Damit wird das Auto auf einer Karte der Strecke lokalisiert. Ist die Strecke unbekannt, so wird simultan eine Karte mit den Kegeln erstellt. Hierbei stehen verschiedene SLAM-Algorithmen (simultaneous localization and mapping) zur Verfügung.

Im Bereich **Planning & Control** gilt es eine Trajektorie des Autos zu planen und durch Regelung das Auto auf dieser Trajektorie zu halten. Das Team orientiert sich dabei an einfachen Trajektorien, wie der Mittellinie auf der Strecke. In Zukunft soll eine optimale Bahnplanung und Regelung mittels Model-Predictive-Control (MPC) durchgeführt werden.

HARDWARE IM FAHRZEUG

Die **Sensoren** mit Anwendung in der Raumfahrt und der Formel 1 und einem Gesamtwert von ca. 80.000€ zeichnen sich durch hochgenaue Informationen über die Umgebung und das Fahrzeug selbst aus. Die gesammelten Daten werden in der zentralen Driverless Compute Unit verarbeitet, die mit Standard-Hardware ausgestattet ist.

Alle **Aktuatoren** werden in enger Zusammenarbeit mit den Subteams Electrics und Vehicle Dynamics erarbeitet. Von zentraler Bedeutung sind die elektrische Lenkung (Steer-by-Wire) sowie das pneumatisch-hydraulische Notbremssystem (Emergency Brake System – EBS) mit Fernabschaltung für den Notfall. Das Bremsen im Nicht-Notfall erfolgt über die Elektromotoren über Rekuperation.

SOFTWARE AUßERHALB DES FAHRZEUGES

Die verschiedenen Sensoren sammeln während dem Entwickeln und Fahren viele verschiedene Datensätze, die in Summe mehrere Terrabyte groß sind. Mit eigenem NAS und Backup im IT-Servicezentrum der Universität Bayreuth (ITS) werden diese mit *S3* und *DVC* ("git für Daten") verwaltet. In Zukunft soll für das **Data Management** eine webbasierte GUI entwickelt werden.

Für das **Annotieren** der Bilddaten nutzt das Team Supervise.ly neben CVAT.

Für das **Training der neuronalen Netzwerke** hat das Team zwei GPU-Workstations mit mehreren Quadro-Grafikkarten. Zusätzlich nutzen wir Amazon Web Services (AWS) für große Rechenoperationen.

Skalierbar ist der Bereich durch Optimierungen der **Data-Pipeline** mit Datenvisualisierungen, der Extraktion von Daten oder der Automatisierung verschiedener Datenverarbeitungsschritte. Wir legen Wert auf einen professionellen, modernen und automatisierten Softwareentwicklungs- und verteilungsprozess vom Laptop über den Server bis in das Auto. DevOps-Interessierte optimieren die Prozesse mit Tools wie Gitlab und Docker.

HARDWARE AUßERHALB DES FAHRZEUGES

Die Testplattform aus Figure 1 hat zwei zentrale Funktionen:

1. Data Collection

- Die Qualität des Driverless Projektes hängt wesentlich von der Anzahl gesammelter Daten ab. Je mehr variantenreiche Daten z.B. Kamerabilder existieren, desto zuverlässiger werden die Kegel erkannt und in ihrer Position geschätzt. Im Idealfall erfolgt das Daten sammeln mit allen im Konzept vorgesehenen Sensoren. Durch die Testplattform wird die Anzahl benötigter Personen reduziert und gewährleistet, dass eine einzelne Person mit allen Sensoren entsprechende Daten sammeln kann.

2. Testen

- Die Entwicklung eines autonomen Fahrzeuges besteht aus vielen Iterationszyklen zwischen Entwickeln, Testen am Laptop und Testen am Fahrzeug selbst bis ein funktionsfähiges und zuverlässiges System existiert. Die Testplattform ermöglicht neue Aufbauten unabhängig vom Fahrzeug zu testen, gerade auch dann wenn das Fahrzeug selbst noch nicht fertig ist.



Figure 1

LANGFRISTIGER ZEITPLAN

- **FÜR 2020: PASSIVES TESTEN**
 - Konzeptionierung
 - Errichtung nötiger Infrastrukturen für Softwareentwicklung (Workstations), Data Management (Server & CICD) und Testen (Testplattform)
 - Komponentenbeschaffung
 - Modulentwicklung Perception & State Estimation
 - Beginn der Hardwareentwicklung
- **FÜR 2021: ACCELERATION UND SKIDPAD**
 - Modulentwicklung Localization & Planning/ Control
 - Formula Student Driverless Events mit Acceleration und Skidpad
- **AB 2022: VOLLAUTONOM AUF UNBEKANNTEN STRECKEN.**
 - Formula Student Driverless Events mit allen Driverless Disziplinen

WAS DU HIER LERNEN KANNST

Der Fokus des Driverless Teams liegt auf den Themenbereichen der **Software**, sodass umfangreiche praxisnahe und intensive **Programmiererfahrungen** in unterschiedlichen Sprachen gesammelt werden können. Wir nutzen je nach Themenfeld C++, Python oder Julia. Die Syntax von Julia ist ähnlich zu MATLAB und erlaubt es jedem Mitglied sein Wissen einzubringen und das System interaktiv zu entdecken. Damit können sowohl reine Programmierer als auch Domänenexperten aus den Ingenieurwissenschaften eingebunden werden.

Besonders interessant ist der Bereich **Sensorik**, als dass die Sensoren teilweise die creme de la creme auf dem Markt darstellen und du besondere Erfahrungen bereits zu Studienzeiten sammeln kannst.

Neben dem Themenfeldern Informatik und der Sensorik gibt es starke Überschneidungen zur **Elektrotechnik** und **Maschinenbau**, nachdem elektrische Schaltungen oder Konstruktionen im CAD durchgeführt werden.

Die Probleme des fahrerlosen Fahrens sind aus der Robotik bekannt und werden seit Jahrzehnten erforscht. Deswegen machen Standardliteratur, Tutorials und Bibliotheken den Einstieg einfach. Zusätzlich haben wir mit dem weltgrößten Automobilzulieferer ZF Friedrichshafen einen zweiwöchentlichen Austausch im Rahmen eines **Mentoren-Programms** sodass du auch hier intensiv mit Experten sprechen kannst. Im Rahmen des intensiven Austausches mit Sponsoren wie Valeo, Daimler, Hexagon Novatel, Kistler oder ZF bieten sich interessante Möglichkeiten für z.B. Praktika oder Werksbesichtigungen an.

TEAM & STRUKTUR

Das Subteam wird aktiv durch zwei leitende Funktionen gesteuert:

- Die **technische Leitung** ist für die Weiterentwicklung des gesamten Systems inkl. Softwarearchitektur verantwortlich.
- Die **administrative Leitung** (Product Manager) kümmert sich um die interne und externe Organisation des Projektes.