Karlsruher Institut für Technologie

Institut für Fahrzeugsystemtechnik
Institutsteil Fahrzeugtechnik
Komm. Leiter: Prof. Dr.-Ing. Marcus Geimer

Kaiserstr. 12
76131 Karlsruhe

Flexibles Mobilitäts- und Cargo-System für den Werksverkehr

Schlussbericht

zum Verbundvorhaben:

Flexibles Mobilitäts- und Cargo-System für den Werksverkehr (FLOOW)

Teilvorhaben:

Fahrzeugplattformen und Connectivity

FKZ: 19A20025C

Projektlaufzeit: 01.03.2021 bis 29.02.2024

Datum
August 2024

Bearbeiter
Michael Frey, Philip Rautenberg, Kevin Simon, Garvin Ellwanger

Gefördert durch:

Bundesministerium für Wirtschaft und Klimaschutz

aufgrund eines Beschlusses des Deutschen Bundestages

Inhaltsverzeichnis

1 Kurzdarstellung 1
1.1 Aufgabenstellung 1
1.2 Voraussetzungen des Vorhabens 2
1.3 Planung und Ablauf des Vorhabens 3
1.4 Wissenschaftlicher und technischer Stand 5
1.5 Zusammenarbeit mit anderen Stellen 7
2 Eingehende Darstellung 8
2.1 Verwendung der Zuwendung und erzieltes Ergebnis 8
2.2 Wichtigste Positionen des zahlenmäßigen Nachweises 45
2.3 Notwendigkeit und Angemessenheit der geleisteten Arbeit 45
2.4 Voraussichtlicher Nutzen 46
2.5 Fortschritte bei anderen Stellen 46
2.6 Veröffentlichung der Ergebnisse 47
3 Verzeichnisse 49
3.1 Literaturverzeichnis 49
3.2 Abbildungsverzeichnis 50
3.3 Tabellenverzeichnis 52

1 Kurzdarstellung

Das Forschungskonsortium im Projekt FLOOW, bestehend aus der ANavS GmbH, der Schaeffler Technologies AG & Co. KG, dem FZI Forschungszentrum Informatik und dem Karlsruher Institut für Technologie, Institut für Fahrzeugsystemtechnik (KIT-FAST), nutzte die zentralen Vorteile der Künstlichen Intelligenz (KI) zur Schaffung neuer Lösungen für die Mobilität von Menschen und Gütern. Der Fokus lag dabei auf einer robusten und hochgenauen Lokalisierung von Fahrzeugen im Mobilitätssystem, welche für Güter- oder Personentransport im Werksverkehr eingesetzt werden, sowie einer generalisierten Umfelderkennung als auch einer risikobewussten Manöverplanung auf dedizierter Hardware zur energieeffizienten Lösung der komplexen Teilprobleme auf den Fahrzeugen.

1.1 Aufgabenstellung

Das KIT-FAST beteiligte sich im Rahmen seines Teilvorhabens an den Arbeitspaketen (AP) 1, 2, 3, 6, 7 und 8. An der Erarbeitung der Inhalte der AP 4 und 5 war das KIT-FAST nicht beteiligt.

In AP1 arbeitete das KIT-FAST an der Analyse von Anwendungsfällen zur Evaluation der Multi-Sensor-Lokalisierung und der KI-basierten Umfelderkennung mit, unterstützte bei der Sammlung von Anforderungen, bei der Erarbeitung einer Methodik zur Bewertung sowie bei der methodischen Analyse von Systemabhängigkeiten und der Definition von Schnittstellen.

Zur Entwicklung einer hochgenauen und zuverlässigen Outdoor-Lokalisierung brachte das KIT-FAST seine Erfahrung bei der Erstellung der Odometrie-Modelle in AP2 ein.

Als Grundlage für die Entwicklung und Erprobung der KI-basierten Algorithmen erstellte das KIT-FAST in AP3 eine dedizierte Simulationsumgebung für die im Projekt verwendeten Demonstratorfahrzeuge sowie digitale Karten des Campusgeländes und der Laborhalle. Dabei wurden auch die notwendigen Schnittstellen für die Kommunikation und Interaktion der Fahrzeugmodelle mit den Algorithmen der Konsortialpartner bereitgestellt. Bei der Konzeptionierung von Chassis-Systemen für kombinierten Indoor- und Outdoorbetrieb brachte das KIT-FAST u.a. die Erfahrungen aus den Verbundprojekten e²-Lenk und OmniSteer ein, leitete die funktionalen Anforderungen an derartige Chassis-Systeme ab und unterstütze bei der Auslegung der zugehörigen Regelungs- und Automatisierungsansätze in AP6. Diese wurden dann in die Simulationsumgebung aus AP3 integriert.

Das KIT-FAST leitete das AP7 „Mobilitätssystem und Connectivity (KI)“ und baute die entsprechende Rechner- und Kommunikationstechnik auf. Zusätzlich wurde ein User-Interface für eine intuitive Bedienung zur Eingabe der Aufgabe bzw. des Fahrziels auch durch ungeübte Nutzer erstellt. Um die Daten für die beim Flottenmanagement eingesetzten KI-Verfahren zu generieren, erstellte das KIT-FAST ein kombiniertes Belegungs- bzw. Nachfragemodell für die in der vereinfachten Erprobungsumgebung vorhandenen Streckenabschnitte. Dieses bildete die Basis für die Bereitstellung bzw. Verwaltung der notwendigen Daten, um daraus mit Hilfe einer KI-basierten Auswertung bzw. Optimierung eine Verteilungsstrategie der Flottenfahrzeuge hinsichtlich der Verfügbarkeit, der Bereitstellungszeit und der Gesamtauslastung zu erarbeiten.

Innerhalb des ebenfalls vom KIT-FAST geleiteten AP8 „Integration und Evaluation“ fand eine kontinuierliche und regelmäßige Integration von Teilsystemen zu immer größer werdenden Funktionseinheiten statt. Dazu wurden die Teilsysteme integriert und anschließend die so entstandene Funktionseinheit getestet. Ziel der Koordination durch das KIT-FAST war dabei, die notwendigen Umfänge der Abstimmung zwischen verschiedenen technischen Partnern sinnvoll zu begrenzen. Mit fortlaufender Projektdauer wurden die einzelnen Teilsysteme größer und umfassender, bis letztlich am Ende die Integration aller Teilsysteme in das Gesamtsystem erfolgte. Schließlich erfolgte die Evaluation des Gesamtsystems anhand eines realitätsnahen Anwendungstests in der vereinfachten Einsatzumgebung (Simulation eines Werksgeländes auf dem Campus Ost des KIT).

1.2 Voraussetzungen des Vorhabens

Das Verbundprojekt FLOOW wurde unter Leitung der ANavS GmbH in folgendem Verbund durchgeführt:

- ANavS GmbH
- Schaeffler Technologies AG & Co. KG, Schaeffler Hub for Advanced Research am KIT (SHARE am KIT)
- FZI Forschungszentrum Informatik
- Karlsruher Institut für Technologie, Institut für Fahrzeugsystemtechnik (KIT-FAST)

Es waren neben dem Konsortium keine weiteren assoziierten Partner waren am Projekt beteiligt. Allerdings diente ein Sounding Board der Sicherstellung, dass die entwickelten Lösungen praxisgerecht sind. Hierzu wurden die Projektergebnisse und relevante Zwischenstände bei mehreren Treffen mit dem Sounding Board diskutiert. Die Mitglieder des Sounding

Boards waren u.a. potenzielle Anwender und Kunden sowie Vertreter der e-mobil BW GmbH (Landesagentur für neue Mobilitätslösungen und Automotive Baden-Württemberg). Damit sollte auch eine weiterführende Zusammenarbeit in Anschlussprojekten und weitere Verwertungsmöglichkeiten vorbereitet werden.

Das Projekt wurde im Zeitraum 01.03.2021 bis 29.02.2024 bearbeitet, wobei die Abschlussveranstaltung am 22.02.2024 in den Örtlichkeiten des KIT-FAST in Karlsruhe stattfand. Das Vorhaben konnte in der geplanten Zeit erfolgreich abgeschlossen werden.

Aufgrund der während des Projektzeitraums aufgetretenen Corona-Pandemie und den damit verbundenen Einschränkungen fand das erste Projekttreffen erst nach einem Jahr statt.

1.3 Planung und Ablauf des Vorhabens

Um die Inhalte des Vorhabens strukturiert zu erarbeiten, wurde das Verbundvorhaben in insgesamt 9 Arbeitspakete unterteilt. Die Inhalte der jeweiligen Arbeitspakete werden nachfolgend kurz erläutert.

AP0 – Projektkoordination und -management

Im Rahmen des AP0 wurden über die gesamte Projektlaufzeit alle administrativen Aufgaben umgesetzt. Diese umfassten die Projektorganisation sowie -koordination durch den Konsortialführer. Des Weiteren fielen in das AP0 die Organisation von Konsortialtreffen, die Kommunikation mit dem Projektträger sowie die Öffentlichkeitsarbeit, wie beispielsweise durch die Erstellung einer gemeinsamen Projektwebsite (https://floow-project.de).

AP1 – Use-Case Analyse und Anforderungsdefinition

In AP1 sollten zuerst die Anwendungsszenarien untersucht und Anforderungen für diese Szenarien definiert werden. Dazu wurden Einsatzgebiete, Szenarien für die Entwicklung und Funktionalität aus Nutzer- und Betreibersicht gesammelt und bewertet. Daraus wurden Anforderungen an das Fahrzeug abgeleitet und konkretisiert. Schnittstellen zwischen den Teilsystemen wurden erarbeitet und für die nachfolgenden Arbeitspakete definiert und detailliert.

AP2 – Entwicklung eines effizienten Multi-Sensor-Lokalisierungssystems

Das Ziel von AP2 war die Entwicklung eines hochgenauen und robusten Lokalisierungssystems mit kostengünstigen Sensoren für den Innen- und Außenbereich. Ein wichtiger Aspekt war dabei auch eine kontinuierliche und sprungfreie Positionslösung bei dem Übergang vom Innen- in den Außenbereich und umgekehrt.

AP3 – Entwicklung einer KI-basierten Umfelderkennung

In AP3 sollte eine hoch-performante KI-basierte Umfelderkennung mithilfe von Deep-Learning-Algorithmen entwickelt werden. Zur Erreichung einer hohen Integrität sollten dabei sowohl Kamera- als auch LiDAR-Daten verwendet werden. Mithilfe einer simulativen Repräsentation des Geländes des Campus Ost sollten die trainierten Neuronalen Netze optimiert und validiert werden.

AP4 – Entwicklung einer KI-basierten Manöver- und Trajektorienplanung

Im Rahmen von AP4 sollte eine Bewegungsplanung entwickelt werden, welche Situationen risikobewusst einschätzt und neben einer Verhaltensintention eine Fahrtrajektorie vorgibt. Dabei sollten Methoden und Verfahren erforscht und entwickelt werden, die mit Hilfe von KI unstrukturierte Situationen lösen und das Fahrzeugverhalten durch Lernen stetig verbessern. Darüber hinaus wurden Verfahren der künstlichen Intelligenz zur Parametrierung verwendet, um eine Adaption der generischen Verfahren an verschiedene Fahrzeugplattformen zu ermöglichen.

AP5 – Entwicklung einer KI-Low-Power Architektur für die HAD-Plattform

Das Ziel von AP5 war die Entwicklung einer Hardware-Plattform, welche die definierten Anforderungen aller verwendeten Systeme berücksichtigt. Der Fokus lag dabei auf einer ausreichend hohen Berechnungsgeschwindigkeit sowie einem minimalen Energieverbrauch, um einerseits die aufwendigen Berechnungen für den autonomen Fahrbetrieb in einer angemessenen Berechnungszeit bereitstellen zu können und um andererseits den Ressourcenbedarf gering zu halten und somit beispielsweise nicht die Reichweite der Demonstratoren zu beeinträchtigen.

AP6 – Fahrzeugplattformen

In AP6 wurden die Fahrzeugplattformen für den Funktionsnachweis des KI-basierten Mobilitätssystems für Indoor- und Outdooreinsätze bereitgestellt. Dabei wurden die Demonstratoren im Hinblick auf Manövrierbarkeit erweitert sowie neue Regelungsansätze für die X-by-wire-Systeme für Indoor- und Outdoor-Einsatz für die Fahrzeugplattformen der verschiedenen Klassen entworfen. Zu guter Letzt erfolgte im Rahmen dieses AP die Inbetriebnahme und Einbindung der adaptierten Fahrzeuge in das Mobilitätssystem.

AP7 – Mobilitätssystem und Connectivity (KI)

Im Rahmen von AP7 wurden eine Eingabemöglichkeit für grundlegende Fahraufgaben der Demonstratoren in Form eines GUI entwickelt, eine Infrastruktur sowie die dazugehörigen KI-

Services zur Verlagerung leistungsintensiver Rechenaufgaben aus den Fahrzeugen ins Backend bereitgestellt und ein Konzept für ein KI-basiertes Flottenmanagement entwickelt, welches insbesondere eine optimierte Verteilungsstrategie der Flottenfahrzeuge aufweist.

AP8 – Integration und Evaluation

Das Ziel von AP8 war die Evaluation des Gesamtsystems, bestehend aus den Fahrzeugplattformen, den KI-basierten Lokalisierungs-, Umgebungserkennungs- und Planungsalgorithmen sowie der vernetzten Mobilitätsinfrastruktur. Hierzu wurde ein realitätsnaher Anwendungstest in der vereinfachten Einsatzumgebung (Simulation eines Werksgeländes auf dem Campus Ost des KIT) durchgeführt. Als Vorbereitung hierzu fand eine kontinuierliche und regelmäßige Integration von Teilsystemen zu immer größer werdenden Funktionseinheiten statt. Mit fortlaufender Projektdauer wurden die einzelnen Teilsysteme größer und umfassender, bis letztlich am Ende die Integration aller Teilsysteme in das Gesamtsystem erfolgte.

1.4 Wissenschaftlicher und technischer Stand

KI-basiertes Flottenmanagement: Flottenmanagement ist das Zuteilen und Verfolgen von Aufgaben an Fahrzeuge einer Flotte. Ziel im Kontext der Mobilität ist eine maximale Verfügbarkeit und kleine Bereitstellungszeit für den Kunden sowie eine maximale Auslastung aller Flottenfahrzeuge. Entscheidend dafür sind Faktoren wie der Aufenthaltsort und die Beförderungswünsche von Kundengruppen zur jeweiligen Zeit, die momentane Position der Fahrzeuge und ihre verbleibende Betriebsdauer (Ladezustand). Die Schätzung von zukünftigen Aufenthaltsorten der Fahrzeuge bzw. von Beförderungswünschen von Kunden, welche ein Fahrzeug der Flotte in Anspruch nehmen wollen, und die dafür beste Positionierung der Fahrzeuge sind weitere anspruchsvolle Aufgaben.

Für die Anwendung in Logistikhallen bzw. Industrieumgebungen werden inzwischen auch Reinforcement Learning basierte Ansätze verfolgt. Beispielsweise wird in [1] Reinforcement Learning zur Verteilung von Transportaufgaben auf autonome Flurförderfahrzeuge in einer Fließbandfertigung eingesetzt. Hierbei basiert der Algorithmus auf Q-learning. Für eine Verteilung von Waren in einem simulierten Lagerhaus mithilfe von Gabelstaplern verwenden die Autoren in [2] sowohl ein Deep Q-Network (DQN), ein Asynchronous Actor Critic (A2C), als auch ein Proximal Policy Optimization (PPO) Algorithmus. Weitere Reinforcement Learning Ansätze für ein Fahrzeug-Flottenmanagement sind bspw. für Mitfahrplattformen aktuell in der Erforschung. Hierbei werden u.a. ein QRewriter–Dueling Deep Q-Network [3] oder ein

Contextual Deep Q-Learning bzw. Contextual Multi-Agent Actor-Critic Algorithmus [4] verwendet.

Verwandte Forschungsprojekte

Zu laufenden bzw. in Vorbereitung befindlichen Aktivitäten der nationalen und europäischen Forschungs- und Innovationspolitik gab es konkrete Anknüpfungspunkte, welche während der Projektumsetzung berücksichtigt wurden. Durch die Beteiligung einzelner bzw. mehrerer Partner aus diesem Konsortium an den nachfolgend benannten Forschungsvorhaben waren die Voraussetzungen für einen zeitnahen Erkenntnis- bzw. Ergebnistransfer geschaffen. Im Folgenden werden diese Projekte beschrieben und die Unterschiede zu FLOOW dargestellt.

3F - Fahrerlose und Fehlertolerante Fahrzeugsysteme für den Niedergeschwindigkeitsbereich: Das Projekt konzentriert sich insbesondere auf die Ausfallsicherheit des Mobilitätssystems und des Bordnetzes, wobei die Entwicklung auf marktübliche Pkw abzielt. FLOOW zeigt dagegen, wie für flottenbasierte Mobilitätssysteme in dynamischer Umgebung mit urbanem Charakter eine Fahrzeugführung umgesetzt werden kann. Interessant für FLOOW könnten die verfolgten Ansätze zur Sensorfusion für die Lokalisierung und Umfelderkennung sein. Der für FLOOW wichtige Indoor-Bereich als auch der Übergangsbereich Indoor/Outdoor wurden aber in 3F nicht betrachtet.

@CITY - Automatisierte Fahrzeuge und intelligenter Verkehr in der Stadt: Die Fokussierung liegt dabei auf Pkw, wobei der urbane Einsatz adressiert wird. In FLOOW wurden exemplarische Fahrzeugplattformen in Form von fahrerlosen Transportsystemen (FTS), Lastenfahrrädern und Rolling-Chassis verfolgt, die eine neuartige Lokalisierung und Umfelderfassung nutzen. Interessant für FLOOW waren die High-Definition (HD) Karten von @CITY.

@CITY-AF: Automatisierte Fahrfunktionen für die Stadt: Hier werden komplexe straßengebundene Szenarien betrachtet, wohingegen FLOOW eine höhere Unstrukturiertheit in den Szenarien abdeckt und eine Interaktion zwischen den Agenten ermöglicht.

AirPortMover - Entwicklung und Erprobung autonomer digitaler Nutzfahrzeuge für Flughafenaußenflächen und Hallen eines Flughafens: Der Fokus liegt auf neuartigen Konzepten zur Lokalisierung und Umfelderkennung im dedizierten Bereich von Flughäfen. Dieses Vorhaben konzentriert sich auf explizite Lösungen mittels KI, wobei ressourcenschonende Umsetzungen entwickelt werden und ebenfalls das Umfeld durch Mensch-Maschine-Interaktion einbindet, wie sie im urbanen Raum häufig anzutreffen ist.

GAMA - Gemischter Verkehr von automatisierten und manövergeführten Fahrzeugen im abgeschlossenen Hafenbereich: Es wird sich auf einen abgeschlossenen Bereich in Form von Hafenmobilität beschränkt und ein modularer Aufbau des Konzepts verfolgt. Im urbanen Bereich hingegen herrscht eine größere Unstrukturiertheit, welche mit den Ergebnissen aus FLOOW durch KI-basierte Methoden gelöst wurden.

AutoTruck - Vollautomatische Verteil-LKW für Automatisierungszonen: Hier wurde auf stark strukturierte Automatisierungszonen fokussiert und Entwicklungen für schwere Fahrzeuge verfolgt. Im vorliegenden Vorhaben werden wendigere Kleinlastentransportfahrzeuge als Fahrzeugplattformen in der Flotte betrachtet.

eMERGE: Das Projekt konzentriert sich auf die Bedarfsanalyse und optimale Ladeinfrastruktur für Elektrofahrzeugflotten, wobei FLOOW auf solche Infrastrukturen aufbauen kann und dedizierte Flotten-Managementsysteme erforscht, welche für autonome Fahrzeuge konzipiert sind.

1.5 Zusammenarbeit mit anderen Stellen

Während der Projektlaufzeit fanden im Konsortium verschiedene Termine zur Wahrung des Projektfortschrittes statt. Neben den regelmäßig online stattfindenden AP-spezifischen Abstimmungsterminen fanden darüber hinaus auch mehrere Konsortialtreffen in Karlsruhe sowie in München statt. Zum Ende des Vorhabens fanden auch vermehrt Termine in der Laborhalle des KIT-FAST statt, um die Teilsysteme im Gesamtverbund zu erproben.

Vor allem in Bezug auf die Datenkommunikation zwischen den Fahrzeugplattformen und der graphischen Benutzeroberfläche zur Auswahl der Fahraufgabe sowie zur Datenaufzeichnung von Sensordaten fanden regelmäßige Abstimmungstermine abseits der AP-Termine mit den Konsortialpartnern ANavS sowie Schaeffler statt.

2 Eingehende Darstellung

2.1 Verwendung der Zuwendung und erzieltes Ergebnis

In den nachfolgenden Kapiteln werden die vom KIT-FAST erarbeiteten Projektergebnisse im Kontext der entsprechenden Arbeitspakete erläutert.

2.1.1 AP0 – Projektkoordination und -management

Im Rahmen des AP0 fand die Koordination aller APs sowie der jeweiligen Zuarbeiten in Abstimmung mit den Konsortialpartnern statt. Die daraus resultierende Zeitplanung wurde im Laufe des Projektes stetig detailliert. Darüber hinaus fand die Organisation der Konsortialtreffen sowie der Treffen des Soundingboards statt, welche nachfolgend aufgelistet sind.

- Konsortialtreffen
  - 05.05.2022 in Karlsruhe am KIT-FAST
  - 20.10.2022 in München bei der ANavS GmbH
  - 22.-23.05.2023 in Karlsruhe am FZI
  - 18.-19.09.2023 in Karlsruhe am SHARE sowie am KIT-FAST
- Treffen Soundingboard
  - 21.10.2022 in München bei der ANavS GmbH
  - 25.10.2023 in Karlsruhe am SHARE

Für die Außendarstellung wurden zu Beginn des Projektes am Campus Ost des KIT Fotos mit den Demonstratorfahrzeugen in einer Werkhalle angefertigt. Die Fotos wurden auf der öffentlichen Projektwebsite (https://floow-project.de veröffentlicht. Eine Auswahl dieser Fotos ist Abbildung 1 zu entnehmen.

Abbildung 1: Die Demonstratorfahrzeuge in einer Werkhalle. (Bilder: Alexander Seiffer, SHARE)

2.1.2 AP1 – Use-Case Analyse und Anforderungsdefinition

Das AP1 behandelte die Use-Case Analyse und Anforderungsdefinition für die verschiedenen Teilsysteme. Die Leitung hatte hierbei ANavS. Das AP teilte sich in nachfolgende Teilarbeitspakete (TAP) auf, wobei das KIT-FAST an allen beteiligt war:

- TAP 1.1: Sammlung und Clustering von Anwendungsfällen für die Entwicklung und die Evaluation
- TAP 1.2: Sammlung und Bewertung von Anforderungen für die Entwicklung und Umsetzung
- TAP 1.3: Analyse von Abhängigkeiten und Definitionen von Schnittstellen

In AP1 wurden gemeinsam mit den Konsortialpartnern die Anwendungsfälle für die drei Demonstratorfahrzeuge (ELF++, Lastenfahrrad, FTS) grundlegend definiert. Dafür wurden eine Kategorisierung sowie eine Bewertung der für die Fahrzeugplattformen notwendigen Software und Hardware durchgeführt. Danach wurden die Anwendungsfälle ausführlich beschrieben, kategorisiert sowie eine Ablaufbeschreibung zur Umsetzung im Projekt erstellt. Die Anwendungsfälle sollten dabei zunächst exemplarisch für das Demonstratorfahrzeug ELF++ ausgearbeitet und dargestellt werden. Nach erfolgreicher Anwendung sollte diese dann auch auf das Lastenfahrrad und das FTS ausgeweitet werden.

Zur Definition der Schnittstellen zwischen den Hard- und Softwarelösungen der Konsortialpartner fanden diverse Workshops statt, um beispielsweise den Datenaustausch zwischen dem in AP7 zu entwickelnden Flottenmanagement und den Lösungen der Projektpartner (Lokalisierung, Trajektorienplanung etc.) zu definieren.

In einem weiteren Workshop wurde gemeinsam ein Value Proposition Canvas für Fahrzeuge der Mikromobilität ausgearbeitet. Hierbei wurden gemeinsam mit den Projektpartnern verschiedene technische Konzepte zur Umsetzung entwickelt.

Für detailliertere Inhalte und Ergebnisse zu AP1 sowie für die Vorstellung der Demonstratorfahrzeuge sei auf den gemeinsamen Schlussbericht des Konsortiums zum Verbundvorhaben FLOOW verwiesen.

Die Inhalte seitens des KIT-FAST für die Meilensteine 1.1 „Fertigstellung der Zusammenstellung von Anwendungsfällen“, 1.2 „Fertigstellung der Definition von Anforderungen“ sowie 1.3 „Fertigstellung der Definition von Abhängigkeiten und Schnittstellen“ wurden (jeweils planmäßig im Projektmonat 6) vollständig erbracht.

2.1.3 AP2 – Entwicklung eines effizienten Multi-Sensor-Lokalisierungssystems

Das AP2 behandelte die Entwicklung eines effizienten Multi-Sensor-Lokalisierungssystems für alle Fahrzeugplattformen. Die Leitung oblag hierbei dem Konsortialpartner ANavS. Das AP teilte sich in folgende TAP‘s auf:

- TAP 2.1: Entwicklung einer hochgenauen und zuverlässigen Outdoor-Lokalisierung
- TAP 2.2: Entwicklung einer hochgenauen und zuverlässigen Indoor-Lokalisierung
- TAP 2.3: Entwicklung einer hochgenauen und zuverlässigen Lokalisierung für einen kontinuierlichen und unterbrechungsfreien Übergang zwischen Indoor und Outdoor

In AP2 war das KIT nur im Rahmen von TAP2.1 beteiligt, nahm zum Zwecke des Informationsaustauschs dennoch regelmäßig an den AP-Treffen teil. Die beigesteuerten Inhalte für die hochgenaue Lokalisierung betrafen Odometriemodelle, welche im Rahmen einer wissenschaftlichen Veröffentlichung durch Mitarbeiter des KIT-FAST bereits vor Beginn des Projektes veröffentlicht [5], im Rahmen des Projektes auf die vorhandenen Fahrzeugmodelle angewandt bzw. adaptiert und im Anschluss dem Projektpartner ANavS bereitgestellt wurden.

Die Inhalte seitens des KIT-FAST für den Meilenstein 2.1 „Fertigstellung der Outdoor-Lokalisierung“ (Projektmonat 18) wurden in TAP2.1 somit vollständig erbracht.

2.1.4 AP3 – Entwicklung einer KI-basierten Umfelderkennung

Das AP3 behandelte die Entwicklung einer KI-basierten Umfelderkennung. Die Leitung oblag hierbei dem Konsortialpartner ANavS. Das AP teilte sich in folgende TAP‘s auf:

- TAP 3.1: Erstellung einer dedizierten Simulationsumgebung für Demonstrator-Fahrzeuge
- TAP 3.2: Entwicklung einer KI-basierten Umfelderkennung mit bestehenden, vortrainierten Neuronalen Netzen
- TAP 3.3: Optimierung der vortrainierten Neuronalen Netzwerke zur Umfelderkennung
- TAP 3.4: Umsetzung der Schnittstelle zur Manöverplanung

In AP3 war das KIT nur im Rahmen von TAP3.1 beteiligt, nahm zum Zwecke des Informationsaustauschs dennoch regelmäßig an den AP-Treffen teil.

Zu Beginn wurden gemeinsam mit den Konsortialpartnern die Schnittstellen sowie die Anforderungen der Simulationsumgebung definiert. Als Software zur Implementierung der

Simulationsumgebung wurden die Softwarelösungen CARLA (https://carla.org/), Gazebo (https://gazebosim.org/home) sowie CarMaker (https://www.ipg-automotive.com/de/produkte-loesungen/software/carmaker/) diskutiert. Die Software CarMaker wurde dabei seitens des KIT-FAST vorgeschlagen, da im Umgang mit dieser Software bereits umfangreiche Erfahrungen vorhanden waren und Modelle zur Abbildung der Fahrphysik einer vorigen Ausbaustufe des Demonstratorfahrzeugs ELF++ bereits vorhanden waren. Da es sich bei der Software allerdings um eine lizenzpflichtige Anwendung handelt und nur das KIT-FAST über entsprechende Lizenzen verfügte, wurde von dieser Lösung Abstand genommen. Sowohl CARLA, als auch Gazebo sind hingegen Open-Source-Lösungen und somit für alle Projektpartner nutzbar. In beiden Fällen müssen physikalische Fahrzeug- sowie Sensormodelle separat entwickelt und integriert werden, wobei teilweise auf öffentlich verfügbare Ansätze zurückgegriffen werden kann. Im Hinblick auf den Detailierungsgrad der Umgebungsvisualisierung zeigte CARLA Vorteile gegenüber Gazebo, wohingegen Gazebo wiederrum besser für die gleichzeitige Ansteuerung mehrerer Fahrzeuge geeignet war. Schlussendlich wurde aus den genannten Gründen die Software Gazebo in Kombination mit der Software Robot Operating System (ROS, https://www.ros.org/) ausgewählt. Der Nachteil in der Umgebungsdarstellung gegenüber CARLA sollte durch die Verwendung von zusätzlichen Computerspielegrafikengines verbessert werden, wie beispielsweise Unreal Engine (https://www.unrealengine.com/de) bzw. Blender (https://www.blender.org/). Für den Aufbau der physikalischen Fahrzeugmodelle wurde die Software MATLAB/Simulink (https://de.mathworks.com/products/simulink.html) ausgewählt.

Der grundlegende Aufbau der Simulationsumgebung ist in Abbildung 2 dargestellt und mit der realen Umgebung verglichen, um die Notwendigkeit der Simulationsumgebung hervorzuheben. Dabei wurden die realen Fahrzeuge und Sensoren durch virtuelle Modelle ersetzt, um deren physikalisches Verhalten abbilden zu können. Des Weiteren wurde die reale Testumgebung des Campus Ost in Gazebo nachgebildet, um ebenfalls die Fahrumgebung realistisch simulieren zu können. Ziel war die Schaffung einer Umgebung zum Testen der Algorithmen der Konsortialpartner. Entsprechend sollten die Schnittstellen in der Simulationsumgebung sowie bei den Fahrzeugmodellen so gestaltet werden, dass diese mit den Schnittstellen in der realen Umgebung übereinstimmen und die Konsortialpartner keinen zusätzlichen Implementierungsaufwand bei der Integration ihrer Algorithmen in der Simulationsumgebung haben.

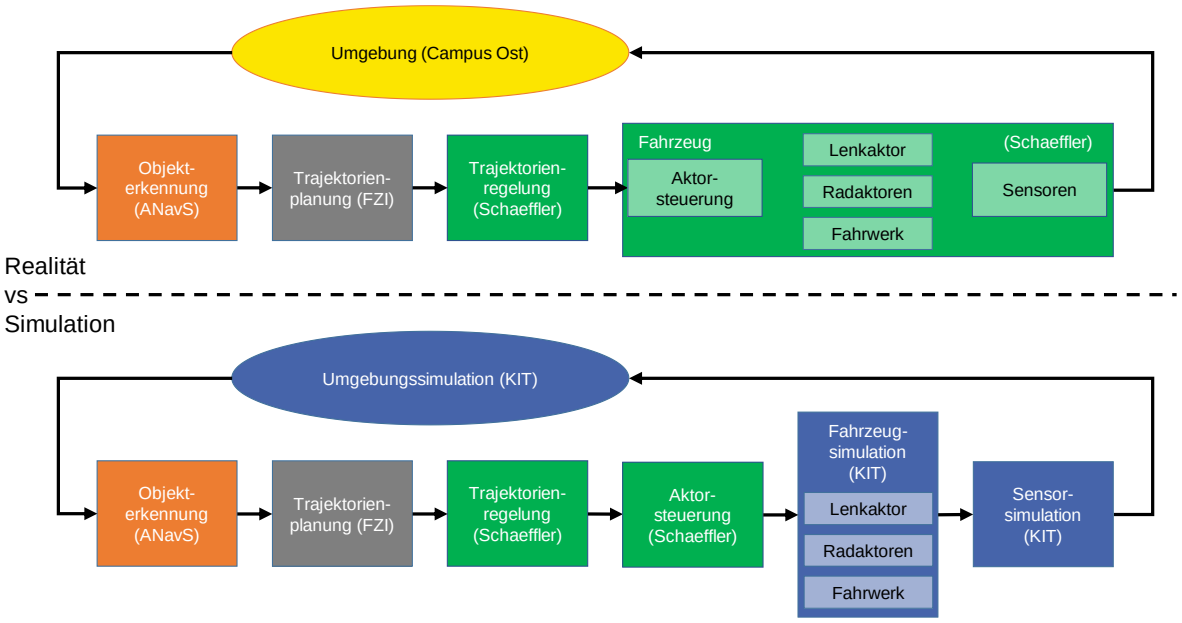

Abbildung 2: Vergleich der realen Prozesskette (oben) mit der virtuellen Prozesskette (unten).

Der Abbildung sind auch die Zuständigkeiten beim Aufbau der Modellumgebung zu entnehmen. Mit dem Aufbau der Fahrzeug-, Sensor- und Umgebungsmodelle oblag dem KIT-FAST nur ein Teil des Gesamtmodells. Die Teilsysteme von Konsortialpartner Schaeffler wurden durch das KIT-FAST lediglich integriert. Die Integration der Teilsysteme der Konsortialpartner ANavS sowie FZI erfolgte bei den jeweiligen Partnern.

Nachfolgend wird der grundlegende Aufbau der vom KIT erstellten Modelle erläutert und die Kommunikationsstruktur der Simulationsumgebung ausführlicher dargestellt.

Aufbau der Gazebo Umgebungssimulation

Der Aufbau der Umgebungssimulation erfolgte in Gazebo. Hierfür wurde der KIT Campus Ost in Karlsruhe mithilfe von Satellitenbildern grundlegend nachgebildet. Zur Darstellung der verschiedenen Gebäude konnte auf eine Bibliothek mit Standard-3D-Modellen zurückgegriffen werden. Mit einer hohen Detailgenauigkeit wurde nur das Gebäude 70.21 sowie der Flächenbereich vor diesem Gebäude nachgebaut, da sich die Fahrzeuge später in diesem Bereich bewegen sollten. Die Anforderungen an die Umgebungssimulation lagen in der möglichst genauen Nachbildung dieser Flächen des KIT Campus Ost als Beispiel für ein Werksgelände. Auch sollte auf eine genaue Texturierung der Gegenstände und Barrieren Wert gelegt werden, um eine Orientierung mit verschiedenen Sensoren wie Kamera oder LiDAR zu ermöglichen. Die simulierte Außenansicht des Werkgebäudes ist in Abbildung 3 und ein Bild des realen Gebäudes in Abbildung 4 dargestellt. In der Außenansicht der virtuellen Werkhalle

fehlen im Vergleich zur realen Werkhalle einige Eingangstore. Diese wurden nicht modelliert, da es sich bei den hinter diesen Toren befindlichen Hallenabschnitten um Bereiche handelt, die für den Verkehr der Demonstratorfahrzeuge nicht vorgesehen waren.

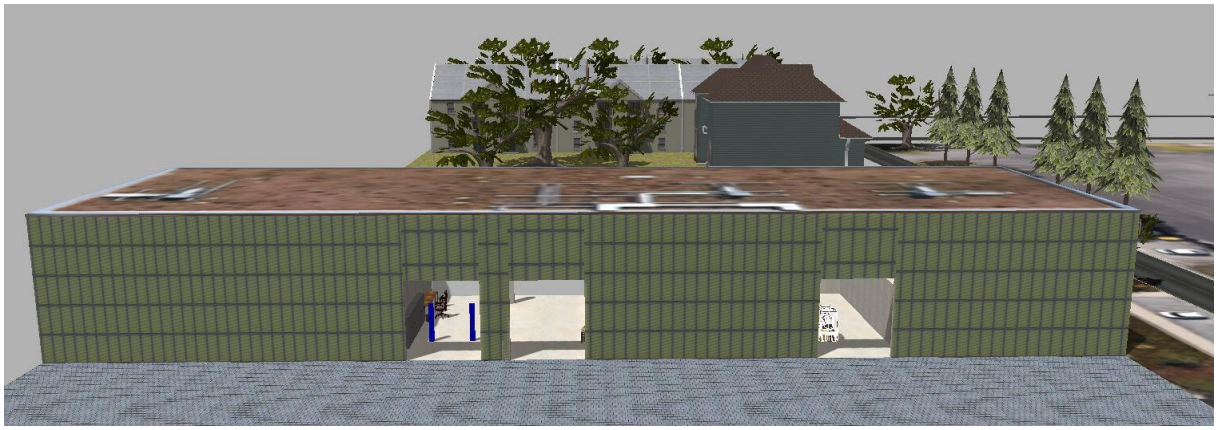

Abbildung 3: Außenansicht der virtuellen Werkhalle.

Abbildung 4: Außenansicht der realen Werkhalle am Campus Ost des KIT in Karlsruhe.

Zusätzlich wurde die Simulationsumgebung in diesem Bereich auch durch andere Objekte ergänzt, wie beispielsweise Tische, Stühle oder Schränke, um eine realitätsnähere Simulation zu erreichen. Auch bei diesen Objekten konnte auf eine Standardbibliothek zurückgegriffen werden, wodurch eine schnelle Integration verschiedener Objekte ermöglicht wurde. Ebenso konnten neben diesen statischen Objekten auch dynamische Objekte eingefügt werden. So wurden zum einen Personen eingefügt, deren Bewegung in einem vorprogrammierten Kurs im Gebäude erfolgte, zum anderen Gabelstapler, welche vor dem Werkgebäude fahren können. In Abbildung 5 sind verschiedene statische als auch in der unteren Bildmitte eine Person als dynamisches Objekt in der Werkhalle zu erkennen. Gazebo ermöglicht in der „.world“-Datei, in der alle Objekte definiert sind, die Definition sogenannter „Goals“, mit denen der Pfad der Bewegung beschrieben werden kann.

Abbildung 5: Innenansicht der Werkhalle mit statischen und dynamischen Objekten.

Auch das Einfügen dreidimensionaler Darstellungen der Demonstratorfahrzeuge war mithilfe von Konsortialpartner Schaeffler bereitgestellter CAD-Modelle möglich, siehe Abbildung 6. Die so integrierten Fahrzeuge können dann wiederum über die im weiteren Verlauf beschriebenen Fahrdynamikmodelle in der Simulationsumgebung gesteuert werden. Gazebo ermöglicht zur Erprobung der Steuerung der Fahrzeuge auch die Montage virtueller Sensoren auf den Fahrzeugen, welche Daten für die Umgebungserkennung sowie Trajektorienplanung bereitstellen. In Abbildung 7 sind neben einem FTS sowie zwei ELF++ auch zwei Gabelstapler als dynamische Objekte zu erkennen, die sich alle im Außenbereich vor der Werkhalle bewegen.

Abbildung 6: Darstellung der drei Demonstratorfahrzeuge ELF++, FTS und Lastenrad

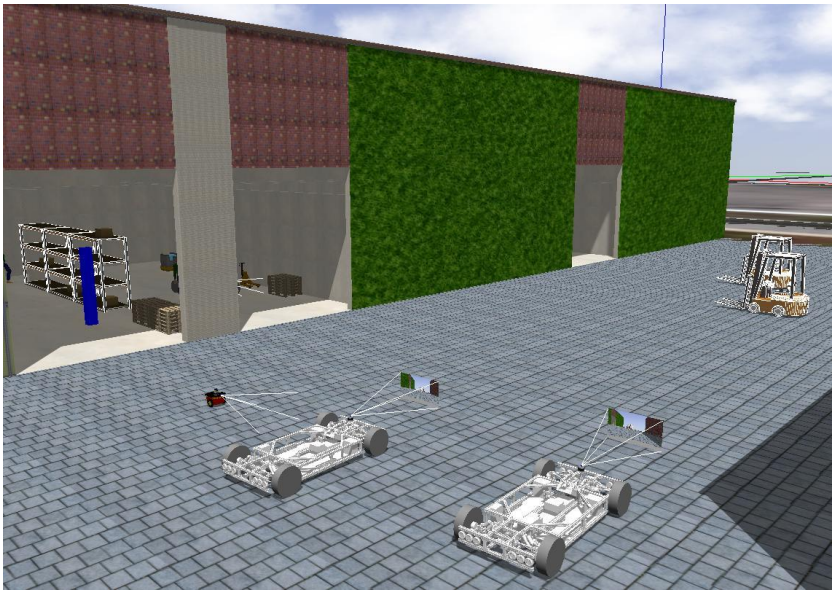

Abbildung 7: Außenansicht der Werkhalle mit einem FTS, zwei ELF++ sowie zwei Gabelstaplern im Außenbereich vor der Werkhalle.

Zur besseren Steuerung der Fahrzeuge wurde mithilfe von Lanelet2 eine Karte in die Umgebung von Gazebo integriert, welche die Straßen in und vor der Werkhalle definiert. Diese wurde auf Basis von Karten von OpenStreetMap erstellt. Die erstellten Straßennetze in und um die Werkhalle können Abbildung 8 entnommen werden.

Abbildung 8: Erstellte Straßennetze im Bereich der Werkhalle mithilfe von Lanelet2.

Zum Austausch der Daten nutzt Gazebo eine ROS-Kommunikation. Hierbei werden definierte Daten auf zugeordneten topics gepublisht, also veröffentlicht. Auf diese topics können dann alle anderen Kommunikationspartner, die sogenannten ROS-Nodes, zugreifen und die enthaltenen Daten weiterverarbeiten. In der entwickelten Simulationsumgebung treten als ROS-Nodes jeweils die verschiedenen Fahrzeuge, die Fahrdynamikmodelle, die Trajektorienplaner der einzelnen Fahrzeuge sowie die GUI zum Flottenmanagement auf. Die Fahrzeuge veröffentlichen hierbei ihren aktuellen Standort auf der topic „*base_pose_ground_truth*“ in Form einer „*nav_msgs::Odometry*“ Nachricht, in der die Position in x,y,z-Koordinaten sowie die Orientierung des Fahrzeugs in kartesischen Koordinaten enthalten sind. Über diese Nachricht ist die Position des Fahrzeugs somit eindeutig bestimmt. Die Kommunikationsstruktur ist in Abbildung 14 dargestellt und wird im weiteren Verlauf des Berichts noch mal detaillierter erläutert. Ebenso ermöglicht diese Kommunikation auch die Aufzeichnung der veröffentlichten Daten über sogenannte rosbags. In diesen rosbags können ausgewählte Daten verschiedener Sensoren, wie etwa eines LiDARs, gespeichert und anschließend zur Validierung der Trajektorienplanung genutzt werden. Ein Beispiel für die Punktwolken eines der verwendeten LiDAR-Sensormodelle ist in Abbildung 9 dargestellt. Die verwendeten Sensormodelle wurden nicht selbst entwickelt, sondern konnten durch öffentlich verfügbare Modelle implementiert werden.

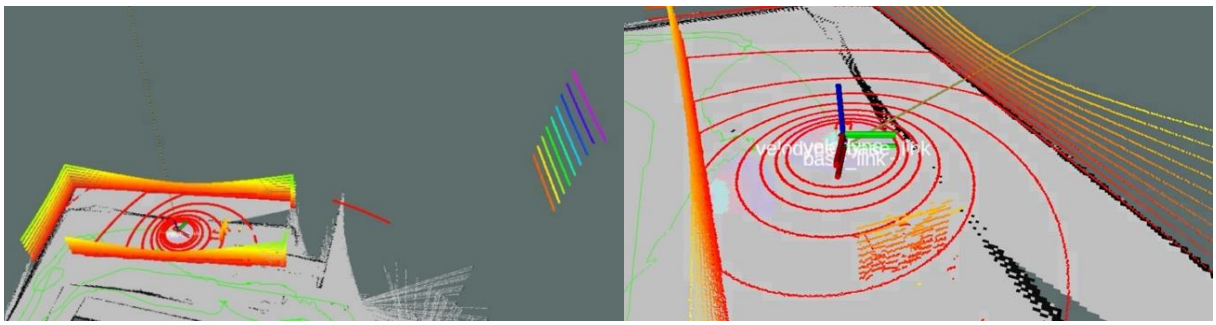

Abbildung 9: Aufgezeichnete Punktwolke eines LiDAR-Sensors in Gazebo.

Physikalische Fahrzeugmodelle

Ein zweiter Aufgabenbereich des KIT-FAST war die Erstellung physikalischer Modelle der Fahrzeuge für die Simulationsumgebung. Die in Simulink aufgebauten Modelle haben dabei die Aufgabe, das Fahrverhalten der verschiedenen Fahrzeuge in der Simulationsumgebung möglichst genau abzubilden. Aufgrund fehlender realer Messdaten der Fahrzeuge, wurden die Modelle anhand der vorhandenen Datenblätter ausgelegt. Nach Rücksprache mit den Projektpartnern wurden die Modelle des Lastenrads und des FTS weniger detailliert aufgebaut, da unter anderem die geringen Geschwindigkeiten dieser Fahrzeuge eine sehr detaillierte

Nachbildung nicht rechtfertigten. Das Modell für das Demonstratorfahrzeug ELF++ wurde umfangreicher in Simulink aufgebaut, wobei auch die im Projektverlauf durch den Partner Schaeffler neu integrierte Hinterachslenkung berücksichtigt wurde. Im Folgenden werden Aufbau und Funktionsweise der verschiedenen Fahrzeugmodelle erläutert.

ELF++

Das Modell des ELF++ lässt sich in mehrere Bereiche unterteilen: den Modelleingang, die Trajektorienfolgeregelung, die Berechnung der fahrphysikalischen Größen im eigentlichen Fahrzeugmodell und die Erstellung der Ausgangssignale. Eine schematische Übersicht ist in Abbildung 10 dargestellt. Nachfolgend wird ausführlicher auf die einzelnen Teilbereiche eingegangen.

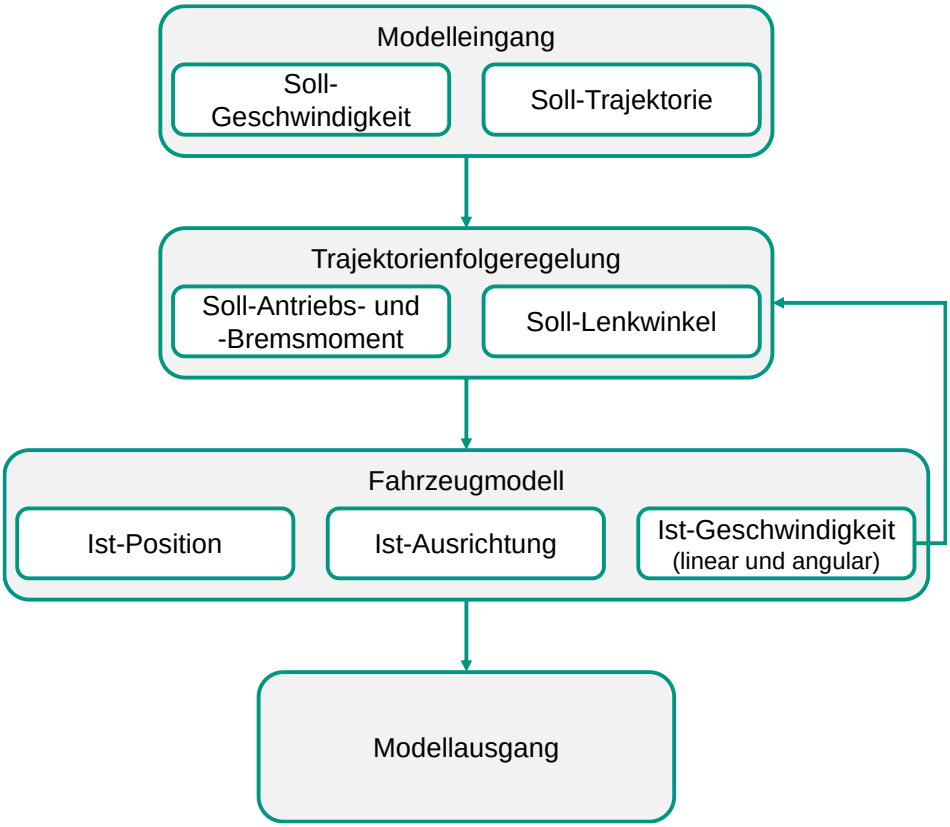

Abbildung 10: Schematische Darstellung der generellen Modellstruktur des ELF++.

Als Eingangsgröße wird dem Modell zum einen eine Trajektorie bereitgestellt, welche durch das Fahrzeug abgefahren werden soll. Des Weiteren werden dem Modell die Position, an der sich das Fahrzeug initial – also zu Simulationsbeginn – befindet, sowie generelle Limits, welche von der Regelung nicht überschritten werden dürfen, zugeführt. Diese Eingänge können dabei je nach Konfiguration der Simulationsumgebung unterschiedlich beschrieben

werden. Zum einen können die Eingänge mit in der Simulation hinterlegten (Mess-)Daten beschrieben werden, was ein einfaches Testen des Modells ohne Kopplung an die Simulationsumgebung in Gazebo ermöglicht. Zum anderen ist das Beschreiben der Eingänge über die noch vorzustellende Kommunikationsschnittstelle möglich.

Zur lateralen Regelung des Fahrzeugs wird im Modell ein erweiterter Stanley-Controller eingesetzt. Hierfür berechnet sich das Modell zuerst benötigte Referenzwerte und gibt diese an den eigentlichen Stanley-Controller weiter. Dieser kann sich dann aus der Geschwindigkeit und der Gierrate des Fahrzeugs, dem Kurvenverhalten der Trajektorie sowie der lateralen Abweichung und der Abweichung der Orientierung von der gewünschten Trajektorie den benötigten Lenkwinkel berechnen. Der Lenkwinkel wird anschließend durch die physikalischen Grenzen des Fahrzeugs beschränkt und mit dem Lenkübersetzungsverhältnis multipliziert. Eine diesem Signal nachgeschaltete Ackermann-Funktion berechnet mithilfe der geometrischen Daten des Fahrzeugs hieraus die Lenkwinkel der einzelnen Räder.

Zur longitudinalen Regelung wird ein PI-Regler verwendet, welcher auf Basis der Differenz der Ist-Geschwindigkeit zur Soll-Geschwindigkeit das Antriebsmoment des Fahrzeugs einregelt. Hierbei werden die durch verschiedene Verlustmechanismen auftretenden Momente addiert, um ein gesamtes Antriebsmoment zu berechnen. Dieses Gesamtantriebsmoment wird anschließend gleichmäßig auf alle vier Räder aufgeteilt und je nach Vorzeichen in ein Beschleunigungsmoment und ein Bremsmoment unterschieden.

Beide Regelungsalgorithmen wurden vom Konsortialpartner Schaeffler bereitgestellt.

Dem eigentlichen (physikalischen) Fahrzeugmodell werden die zuvor berechneten Werte für die Lenkwinkel und das Beschleunigungs- und Bremsmoment sowie die initiale Position des Fahrzeugs weitergeleitet. Aus diesen wird unter Berücksichtigung der Stellvermögen der Aktuatoren die real gestellten Werte für die Lenkwinkel und die Momente berechnet. Diese können dann verwendet werden, um das dynamische Verhalten der Radaufstandspunkte, des Fahrzeugaufbaus, des Fahrwerks und der Reifen zu simulieren. Mithilfe dieser berechneten Größen und der initialen Position des Fahrzeugs können dann die neue Position des Fahrzeugs sowie die Lenkwinkel und Orientierung berechnet und ausgegeben werden. Es handelt sich bei dem Modellansatz um ein ebenes Zweispurmodell mit drei Freiheitsgraden. Eine Übersicht über das Fahrzeugmodell ist in Abbildung 11 dargestellt.

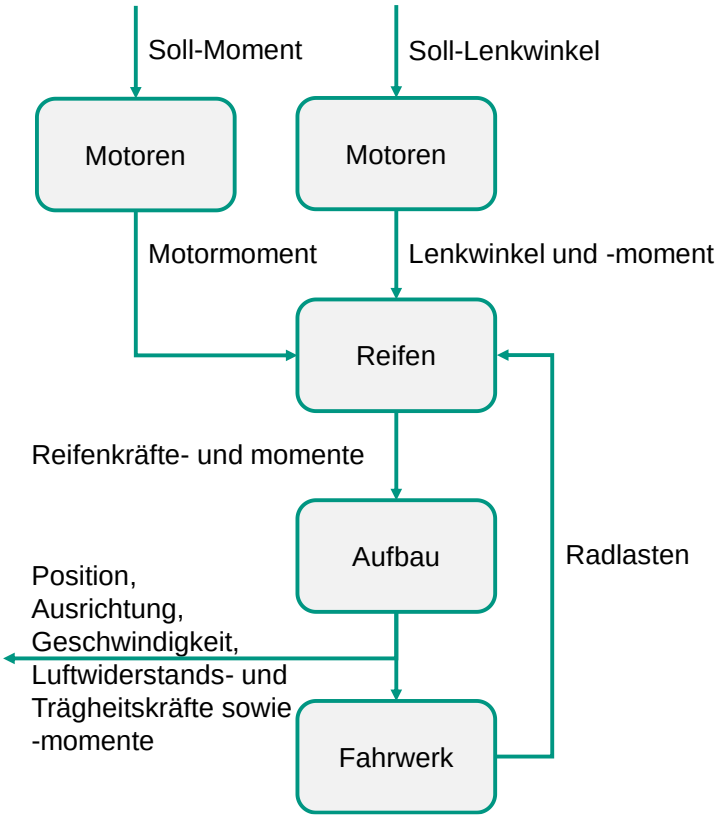

Abbildung 11: Schematische Darstellung des ELF++-Fahrzeugmodells.

Im Ausgangsblock werden alle zuvor berechneten Größen gesammelt und je nach Einstellung an die restliche Simulationsumgebung ausgegeben.

FTS

Der grundsätzliche Aufbau des FTS-Modells ist vergleichbar mit dem des ELF++-Modells. So gibt es auch in diesem Modell einen Eingangsblock, in dem die Eingangssignale bereitgestellt werden, einen Ausgangsblock, in dem die berechneten Größen gesammelt ausgegeben werden sowie Blöcke für die Trajektorienfolgeregelung und für das eigentliche (physikalische) Fahrzeugmodell, siehe Abbildung 12. Im Vergleich zum ELF++ unterscheiden sich wegen des unterschiedlichen Aufbaus des Fahrzeugs bzw. dessen Antriebssystems allerdings die Art der Eingangsgrößen sowie die Art der Trajektorienfolgeregelung. Als Eingangsgrößen wird neben der Soll-Geschwindigkeit auch die Soll-Gierrate anstelle der Soll-Trajektorie bereitgestellt. Mithilfe dieser Eingangsgrößen berechnet der Trajektorienfolgeregler – basierend auf einem PID-Regler – dann aufgrund des fehlenden Lenksystems im FTS lediglich ein Antriebs- oder Bremsmoment, welches dem Fahrzeugmodell bereitgestellt wird. Die Ausrichtung des Fahrzeugs wird indirekt über die Differenz der Momente an den beiden Antriebsrädern eingeregelt.

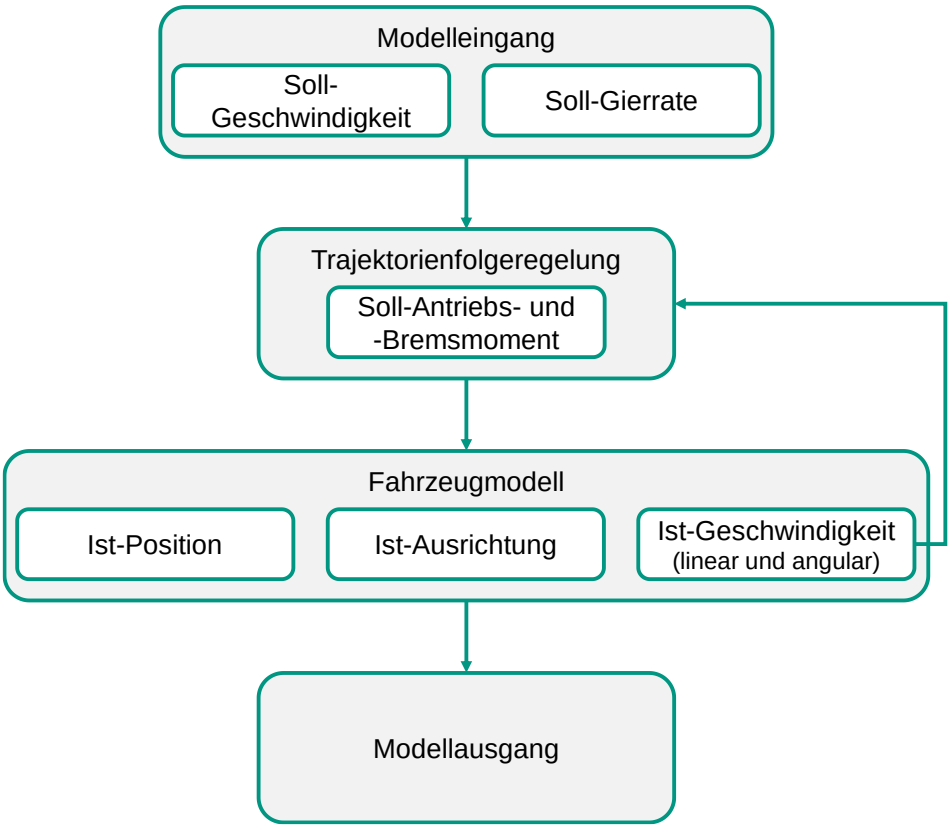

Abbildung 12: Schematische Darstellung der generellen Modellstruktur des FTS sowie Lastenrads.

Das eigentliche Fahrzeugmodell des FTS ist weniger detailliert aufgebaut als beim ELF++. In diesem werden nur die Dynamik des Aufbaus und der Rollwiderstand der Reifen simuliert, das Fahrwerk sowie der Strömungswiderstand der Reifen werden aufgrund der niedrigen Geschwindigkeiten vernachlässigt. Eine schematische Darstellung des Fahrzeugmodell-aufbaus kann Abbildung 13 entnommen werden.

Lastenrad

Das Modell des Lastenrads gleicht vom Aufbau und der Funktionsweise dem des FTS. Auch hier wird zur Regelung der Beschleunigungs- und Bremsmomente ein PID-Regler verwendet, der die Radmomente anhand der Soll-Ist-Geschwindigkeitsabweichung einregelt. Diese Radmomente werden ebenso an das eigentliche (physikalische) Fahrzeugmodell weiter-geleitet, in dem anhand der Motorcharakteristik und der Simulation des Aufbaus und der Reifen die Position des Fahrzeugs simuliert wird. Der einzige Unterschied zum FTS liegt hierbei in der Simulation der Lenkung des Lastenrads und der Berechnung des Lenkwinkels, siehe gestrichelte Linien in Abbildung 13. In diesem Abschnitt wird der Lenkwinkel des dritten, nicht angetriebenen und nur hinterhergezogenen Rades im Verhältnis zur angetriebenen vorderen Achse berechnet.

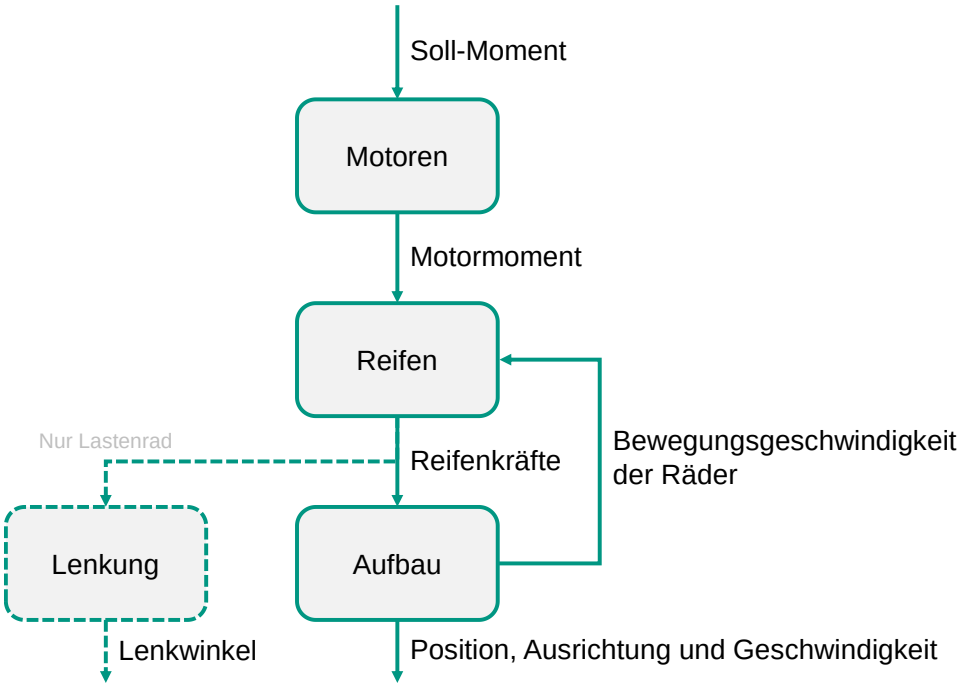

Abbildung 13: Schematische Darstellung des Fahrzeugmodells für FTS und Lastenrad.

Integration der Fahrdynamikmodell in die ROS-Kommunikation

Die in MATLAB/Simulink erstellten Fahrzeugmodelle wurden anschließend mithilfe der Simulink C-Code-Generierung in C-Code umgewandelt, um sie ohne MATLAB-Installation in der Gazebo-Umgebung verwenden zu können. Hierfür wurden für jedes Fahrzeugmodell roscpp-Nodes erstellt, in welche der generierte C-Code eingebunden wurde. Die ROS-Nodes verwenden dabei Subscriber, um die von den Fahrzeugmodellen benötigten Signale von anderen ROS-Nodes zu empfangen und zwischenzuspeichern. Diese Signale werden anschließend innerhalb der ROS-Node an das kompilierte Simulink-Fahrzeugmodell weitergeleitet und ein Simulationsschritt wird ausgeführt. Nach diesem Simulationsschritt stehen wiederum die Ausgangssignale innerhalb der ROS-Node zur Verfügung und können über Publisher an die übrigen ROS-Nodes weitergeleitet werden. Die Fahrdynamik-ROS-Nodes erhalten dabei die Werte der Soll-Trajektorie oder der Sollgeschwindigkeit und Sollgierrate von den Trajektorienplanern. Zusätzlich erhält die ROS-Node des ELF++ Fahrzeugs noch Werte für die Limits und die aktuelle Position des Fahrzeugs. Zur Steuerung der Fahrzeuge in der Gazebo-Umgebung publishen die ROS-Nodes die neue Position der Fahrzeuge in dem Topic „*SetModelState*“. Eine Darstellung der Signalverknüpfung innerhalb der gesamtheitlichen Simulationsumgebung ist in Abbildung 14 dargestellt.

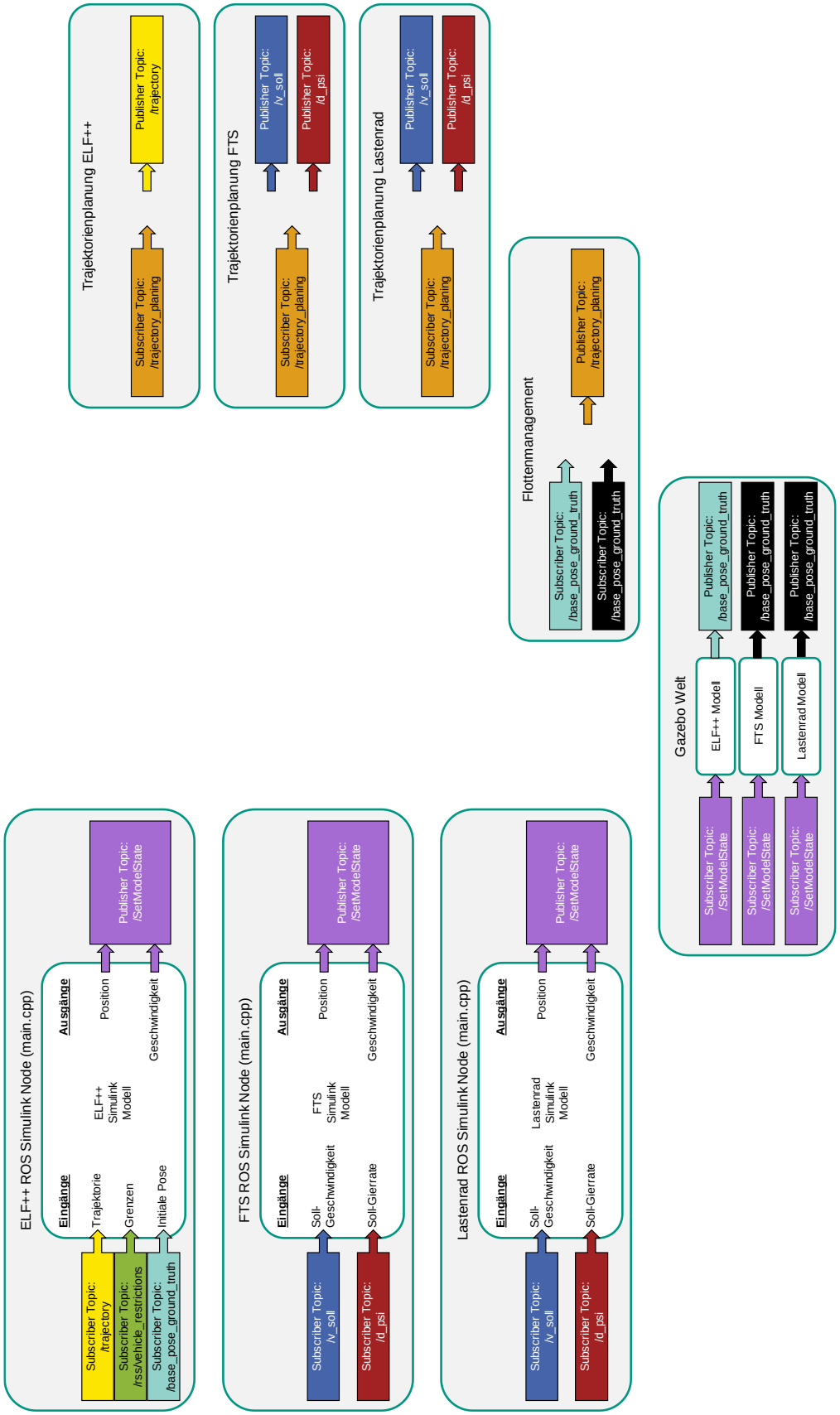

Abbildung 14: Signalverknüpfungen in der gesamtheitlichen Simulationsumgebung.

LiDAR-Datenaufzeichnung

Da 3D LiDAR-Sensoren eine unmittelbare präzise 3D-Erfassung der Umgebung erlauben, lag der Fokus auf der 3D Objekterkennung mittels LiDAR. Im Gegensatz zu Kameras bieten LiDAR-Sensoren den Vorteil, dass die Tiefenerfassung unmittelbar durch Distanzmessungen geschieht und die Tiefen nicht aus Bildern geschätzt werden muss. Außerdem können LiDAR-Sensoren auch bei Nacht eingesetzt werden und sind damit nicht abhängig von den Beleuchtungsverhältnissen wie bspw. Kameras.

Für das Training der KI-Algorithmen zur Umfelderfassung der Demonstratorfahrzeuge wurde deshalb ein gelabelter LiDAR-Datensatz der Testumgebung (Werkhalle des KIT) mit ausgewählten Objektklassen benötigt. Open-source LiDAR-Datensätze stellten keine Lösung dar, da während der Laufzeit des Projektes keine öffentlichen LiDAR-Datensätze für Industrieumgebungen vorhanden waren (zumeist nur Datensätze aus dem Straßenverkehr etc.). Daher betreute das KIT gemeinsam mit dem Projektpartner ANavS eine Masterarbeit, in deren Rahmen ein gelabelter LiDAR-Datensatz für die Werkhalle erstellt wurde. Dabei kam die Integrierte Sensorplattform (ISP) von ANavS zur Aufzeichnung von Rohdaten zum Einsatz, welche in Abbildung 15 erläutert ist.

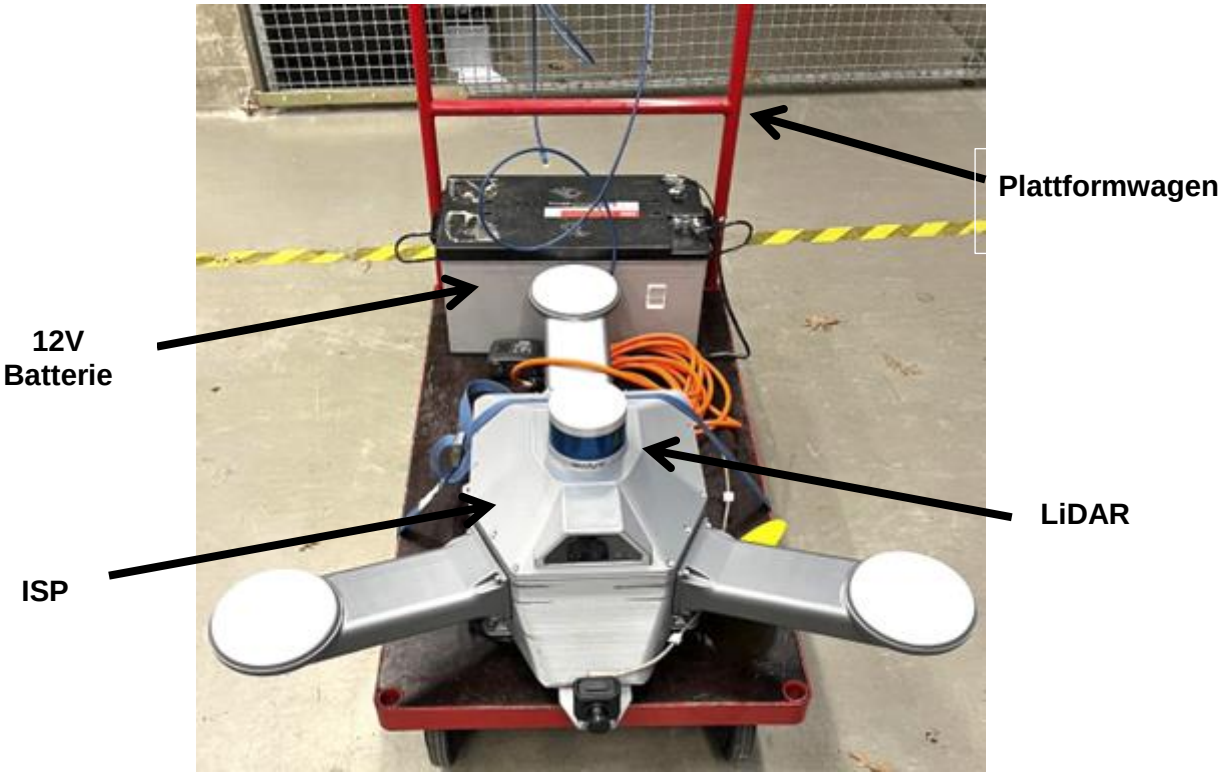

Abbildung 15: Setup zur Aufzeichnung der LiDAR-Daten in der Werkhalle des KIT

Das Setup besteht aus einem Plattformwagen zum Manövrieren der Sensorplattform durch die Werkhalle, der eigentlichen ISP Sensorplattform, einer 12V-Batterie zur Stromversorgung sowie einem Velodyne VLP-16 LiDAR. Weiterhin sind auch ein GNSS-Receiver sowie Kameras verbaut. An die Sensorplattform wurde mit Hilfe eines LAN-Kabels ein Laptop angeschlossen, welcher zur Steuerung der Datenaufzeichnung verwendet wurde. Die Datenaufzeichnung enthielt unter anderem Gitterboxen, Paletten, Tische und auch eine Reinigungsmaschine. Da die meisten open-source Modelle zur Objekterkennung hochauflösende LiDAR-Punktwolken benötigen, bspw. wurde im KITTI-Datensatz ein 64-Channel LiDAR verwendet, die Sensorplattform allerdings nur einen 16-Channel LiDAR besitzt, wurde ein Upsampling (Erzeugung dichterer Punktwolken) der LiDAR-Punktwolken durchgeführt. Das Ergebnis ist beispielhaft für eine Datenaufzeichnung in Abbildung 16 dargestellt.

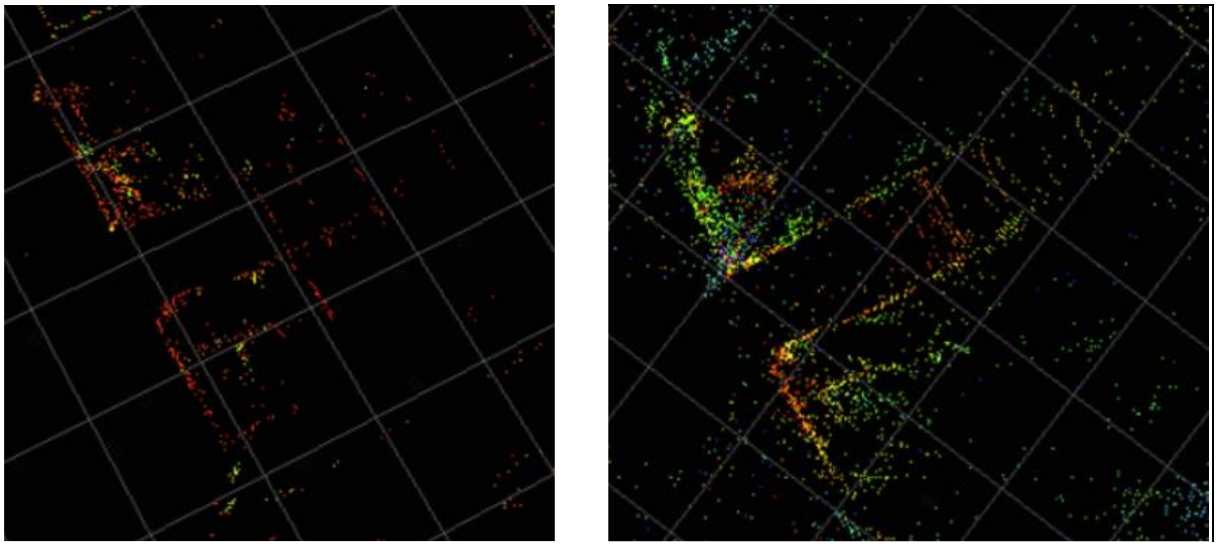

Abbildung 16: Aufgezeichnete Punktwolke mit 16-Channel LiDAR (links) und mit dem LiDAR Super Resolution Algorithmus erzeugte upsampled Punkwolke (rechts)

Im Anschluss erfolgte ein Labeling (3-D Bounding Boxen) der Punktwolken durch die Software MATLAB 3D Ground Truth Labeler. Abbildung 17 zeigt Beispiele verschiedener Objektklassen und der manuellen Bounding Box Labels, welche durch eine Erweiterung des Frameworks von Seiten ANavS entstanden sind.

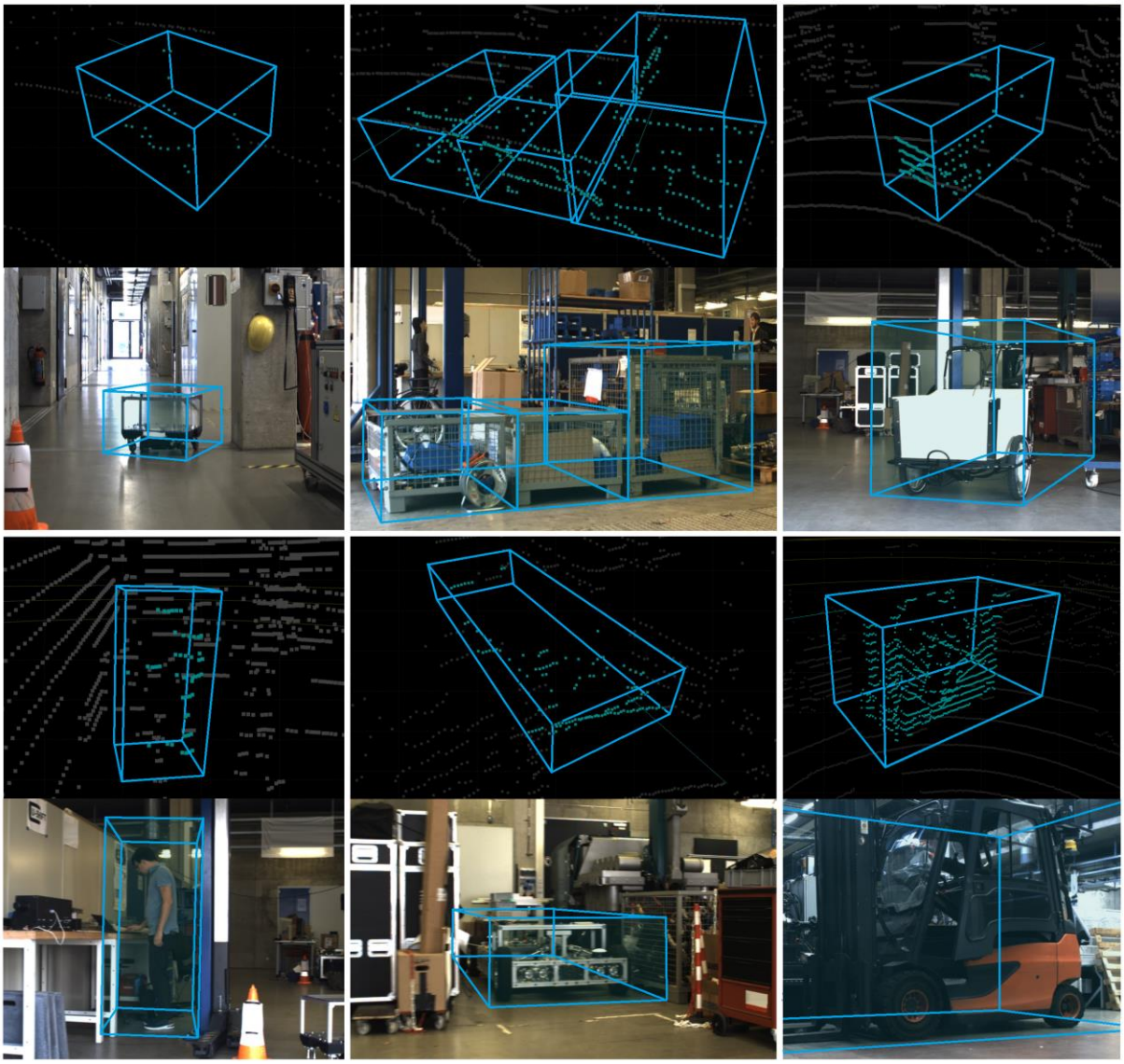

Abbildung 17: Beispiele der verschiedenen Objektklassen mit 3D Punktwolken- und den entsprechenden Kamerabildern.

Nach dem oben beschriebenen Labeling der Punktwolken durch die Software MATLAB 3D Ground Truth Labeler wurde ein Training und die Validierung eines open-source Objekterkennungsalgorithmus (Part A^2) auf die LiDAR-Daten durchgeführt, als Software hierfür kam Python bzw. die Bibliothek PyTorch zum Einsatz. Das Modell war bereits auf dem KITTI-Datensatz vortrainiert und wurde nun auf die ausgewählten Objektklassen Gabelstapler, Gitterbox und Demonstratorfahrzeug nachtrainiert. Die Ergebnisse verdeutlichen eine zuverlässige Detektion der drei Objektklassen. Die Average Precision sowie der Average Recall bei einem IoU von 50% liegt bei über 99%. Abbildung 18 verdeutlicht nochmals die Objekterkennung des trainierten Algorithmus.

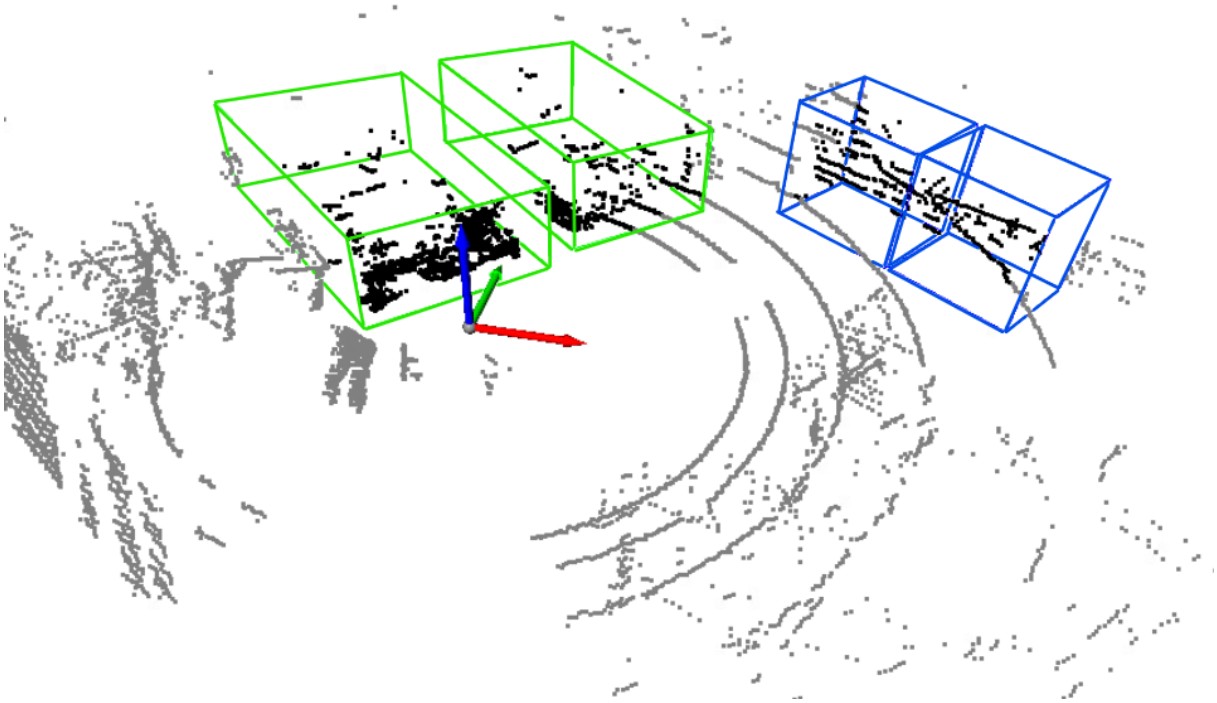

Abbildung 18: 3D-Bounding Boxen Prädiktion der LiDAR-Objekterkennung

Eine Übersicht der finalen Testergebnisse mit den aufgezeichneten LiDAR-Daten ist Tabelle 1 zu entnehmen. Hierbei ist für eine IoU von 50% bzw. 75% jeweils die Average Precision (AP) sowie Average Recall (AR) berechnet.

Tabelle 1: Testergebnisse Objekterkennung für die aufgezeichneten LiDAR-Daten in der Werkhalle.

Klassen	**AP_0.5 [%]**	**AR_0.5 [%]**	**AP_0.75 [%]**	**AR_0.75 [%]**
Gitterbox	100	100	87,97	88,41
Demonstratorfahrzeug	100	100	100	100
Gabelstapler	100	100	98,18	98,46
Gesamt	100	100	95,38	95,62

Final wurde das trainierte Modell in eine GUI eingebunden, mit deren Hilfe ein automatisches Labeling von neuen LiDAR-Rohdaten in Industrieumgebungen möglich ist. Der Benutzer kann weiterhin die Ergebnisse des Labelings manuell überprüfen, siehe Abbildung 19.

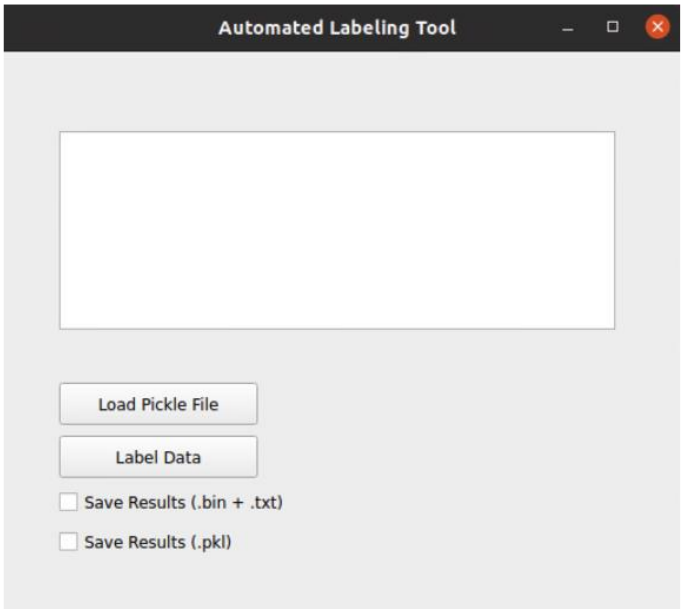

Abbildung 19: GUI für das Labeling des 3D LiDAR Datensatzes.

Parallel zu der oben beschriebenen Masterarbeit wurde der Ansatz des Annotations-Frameworks durch den Projektpartner ANavS erweitert. In Abbildung 20 ist das ursprüngliche Framework der Masterarbeit (blau) sowie die Erweiterungen von ANavS (gelb und grün) verdeutlicht.

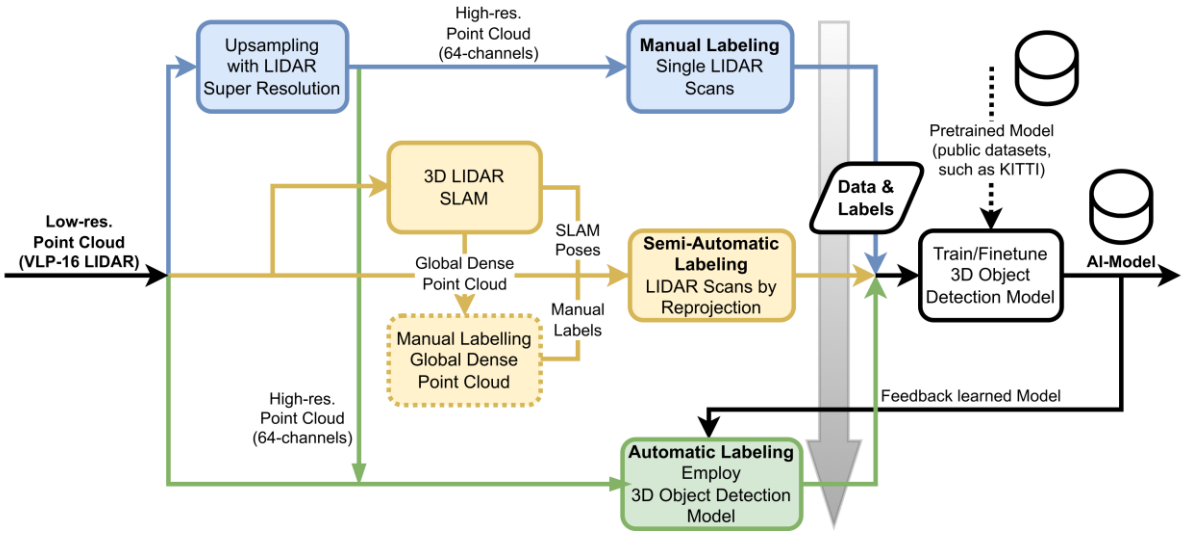

Abbildung 20: Automatisiertes KI-basiertes Annotations-Framework. Von manueller zur automatischen Annotation (von oben nach unten) basierend auf niedrig aufgelösten LiDAR-Daten.

Die oben beschriebene Masterarbeit sowie die einhergehenden Erweiterungen wurden von KIT und ANavS gemeinsam in der Veröffentlichung publiziert, siehe [6]. Weitergehende Informationen sind dieser Veröffentlichung zu entnehmen. Der gelabelte LiDAR-Datensatz wurde der Öffentlichkeit über die Plattform GitHub zugänglich gemacht, siehe Link.

Die Inhalte seitens des KIT-FAST für den Meilenstein 3.1 „Fertigstellung der Simulationsumgebung“ (Projektmonat 18) wurden in TAP3.1 vollständig erbracht.

2.1.5 AP4 – Entwicklung einer KI-basierten Manöver- und Trajektorienplanung

Das KIT-FAST war in AP4 nicht beteiligt.

2.1.6 AP5 – Entwicklung einer KI-Low-Power-Architektur für die HAD-Plattform

Das KIT-FAST war in AP5 nicht beteiligt.

2.1.7 AP6 – Fahrzeugplattformen

Das AP6 behandelte die verschiedenen Fahrzeugplattformen. Die Leitung oblag hierbei dem Konsortialpartner Schaeffler. Das AP teilte sich in folgende TAP‘s auf:

- TAP 6.1: Konzeptionierung Chassis-Systeme für Indoor- und Outdoorbereich
- TAP 6.2: Konzeptionierung von Indoor- und Outdoor-Regelungs- und Automatisierungsansätzen
- TAP 6.3: Adaption der Fahrzeugplattformen: FTS und Lastenrad
- TAP 6.4: Inbetriebnahme Fahrzeugplattformen

In AP6 war das KIT im Rahmen von TAP6.1 sowie TAP6.2 beteiligt, nahm zum Zwecke des Informationsaustauschs auch nach Abschluss dieser APs zusätzlich regelmäßig an den AP-Treffen teil.

Im Rahmen von TAP6.1 wurden die Anforderungen an die Demonstratorfahrzeuge hinsichtlich deren Eignung für den Einsatz im In- sowie Outdoorbereich diskutiert. Des Weiteren wurden die Art und die Positionen der zu verwendenden Sensoren auf den Demonstratorfahrzeugen finalisiert. Daraus ableitend wurden auch die in AP1 definierten Use-Cases noch mal überarbeitet und die Fahrstrecken in und vor der Werkhalle angepasst, alles in Vorbereitung für die Praxistests in AP8. Die entsprechenden Bereiche und möglichen Fahrwege sind in Abbildung 21 und Abbildung 22 aufgezeigt.

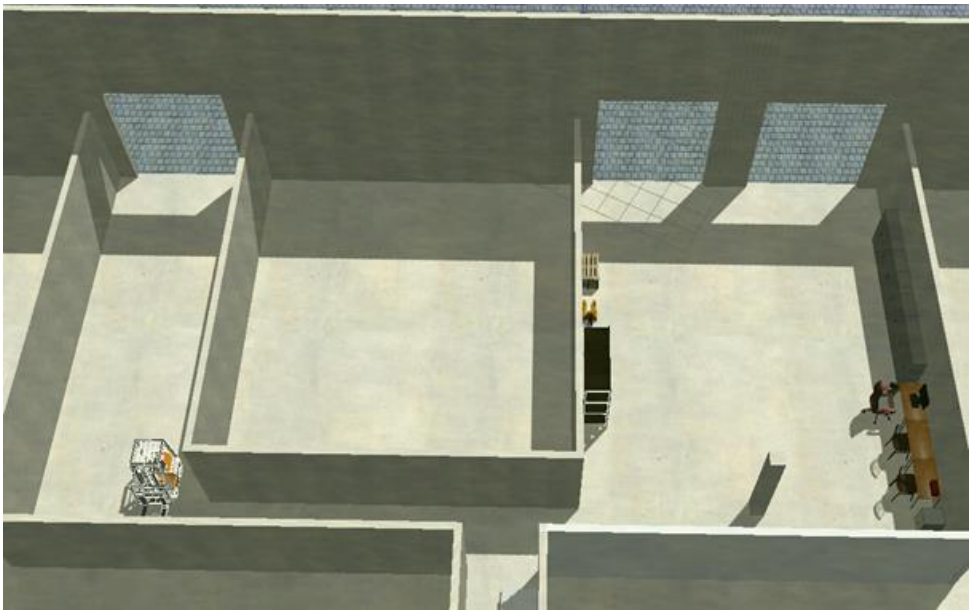

Abbildung 21: Testbereiche der Werkhalle in der Simulationsumgebung

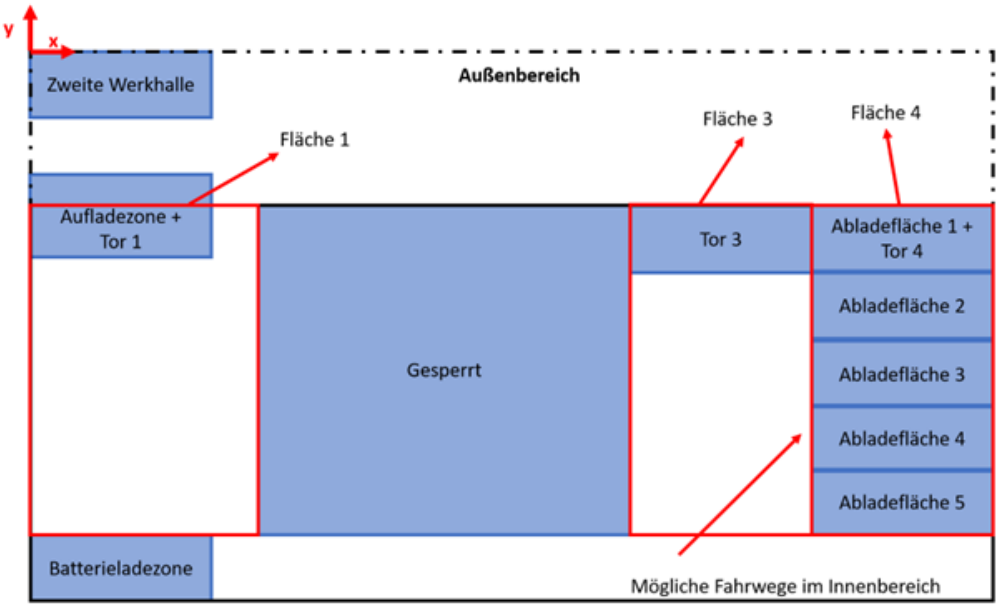

Abbildung 22: Testbereiche der Werkhalle im Detail

Final wurden sechs Szenarien definiert. Die Anfahrt zum Abladebereich erfolgte über Fläche 3 von innen oder durch Tor 4 von außen. Weiterhin fuhr das jeweilige Demonstratorfahrzeug zuerst die Aufladezone für die Waren an. Das Lastenfahrrad wurde hierbei nicht mehr betrachtet.

Nachfolgend sind die verschiedenen Use Cases aufgelistet:

- Use Case 1
  - Das ELF++ fährt durch den Bereich 3 in den Abladebereich und verlässt diesen Bereich wieder rückwärts und fährt zum Batterieladebereich
- Use Case 2
  - Das FTS fährt den Abladebereich an, dreht um und fährt den Batterieladebereich an
- Use Case 3
  - Das ELF++ fährt im Außenbereich um ein Hindernis, stoppt im Bereich der zweiten Werkhalle und fährt von außen durch Tor 4 den Abladebereich an und anschließend wieder den Batterieladebereich
- Use Case 4
  - Das ELF++ fährt über die Fläche 3 den Abladebereich an und muss im Flur vor einer Reinigungsmaschine anhalten. Ein „künstlicher“ Fehler des ELF++ wird gemeldet, der Fahrauftrag wird gelöscht
- Use Case 5
  - Der vorherige Fahrauftrag in Use Case 4 wird nun durch das FTS ausgeführt. Das FTS umfährt das ELF++ und die Reinigungsmaschine und fährt den Abladebereich an. Danach dreht es um und fährt den Batterieladebereich an. Beim ELF++ wird ein Restart durchgeführt, danach ist es wieder einsatzbereit
- Use Case 6
  - Ähnlich zu Use Case 3, allerdings muss nun das ELF++ einen kreuzenden Gabelstapler im Außenbereich umfahren. Danach steuert das Fahrzeug den Abladebereich durch das Tor 4 an und fährt danach zur Batterieladezone

Die in TAP6.2 für das KIT-FAST definierten Aufgaben überschnitten sich stark mit denen aus TAP3.1, weshalb beispielsweise der Aufbau sowie die Integration der Automatisierungs-algorithmen bereits in Kapitel „AP3 – Entwicklung einer KI-basierten Umfelderkennung“ thematisiert wurde. Die in TAP3.1 vorgestellte Simulationsumgebung wurde mit integrierten Automatisierungsansätzen sowie Schnittstellen für beispielsweise die Trajektorienplanung an die Konsortialpartner übergeben. Dabei konnte vor allem Optimierungspotential hinsichtlich der Auswahl der Simulationsumgebung bezüglich realistischer Darstellung identifiziert werden. Eine Verwendung der Simulationsumgebung zum Training von Umfelderkennungsalgorithmen konnte als nicht ausreichend festgestellt werden. Trotz erfolgreicher Optimierungen durch die

Integration von veränderlichen Lichtverhältnissen sowie Oberflächenbeschaffenheiten, war die optische Qualität der virtuellen Kamerasensorik nicht ausreichend, um einen synthetischen Datensatz für die Kameraobjekterkennung zu erstellen. Weitere Versuche, die Simulationsumgebung durch beispielsweise Unreal Engine oder Blender zu erweitern, um die optische Qualität zu erhöhen, waren mit nicht lösbaren Herausforderungen behaftet. Ebenfalls führte die Integration einer Karte im Lanelet2 Format zu Problemen. Nach Absprache mit den Konsortialpartnern wurden keine weiteren Optimierungsversuche der Simulationsumgebung hinsichtlich optischer Qualität und Kartenimplementierung unternommen. Für alle weiteren Anwendungszwecke (z.B. Trajektorienplanung) war die Simulationsumgebung ausreichend gut.

Die in AP6 fälligen Meilensteine 6.1 „Fertigstellung neuartiges Chassis-Konzept“ und 6.2 „Regelungs- und Automatisierungsalgorithmen erarbeitet“ wurden somit erbracht.

2.1.8 AP7 – Mobilitätssystem und Connectivity (KI)

Das AP7 behandelte die Implementierung einer Connectivity-Lösung für die Versuchsfahrzeuge ELF++, FTS und das Cargo-Bike (Lastenfahrrad). Als Testumgebung diente die Werkhalle am Campus Ost des KIT. Die Leitung oblag hierbei dem KIT-FAST. Das AP teilte sich in folgende TAP‘s auf:

- TAP 7.1: Implementierung eines User Interface mittels Touchscreen
- TAP 7.2: Entwicklung des KI-basierten Flottenmanagements
- TAP 7.3: Integration der KI-Services für HAD-Plattform
- TAP 7.4: Inbetriebnahme der Connectivity-Lösung und Einbindung in die Mobilitätsinfrastruktur

In AP7 war das KIT im Rahmen von TAP7.1, TAP7.2 sowie TAP7.4 beteiligt, nahm als AP-Leiter aber auch an TAP7.3 teil.

Implementierung eines User Interface mittels Touchscreen

In AP7.1 wurde zuerst ein Konzept einer GUI für verschiedene Anwendungsfälle in der Werkhalle ausgearbeitet. Die GUI ist Tablet-basiert und dient der Eingabe von Informationen für das Flottenmanagement, bspw. Größe des zu transportierenden Gutes, Zielort etc. Hierfür wurde als Programmiersprache Python verwendet. Zur finalen Entwicklung wurden Zustandsdiagramme für die Aufträge sowie die Fahrzeugzustände entworfen. Da die GUI in

englischer Sprache entwickelt wurde (international verständlich), sind die Zustandsdiagramme ebenfalls in Englisch verfasst, siehe Abbildung 23 und Abbildung 24.

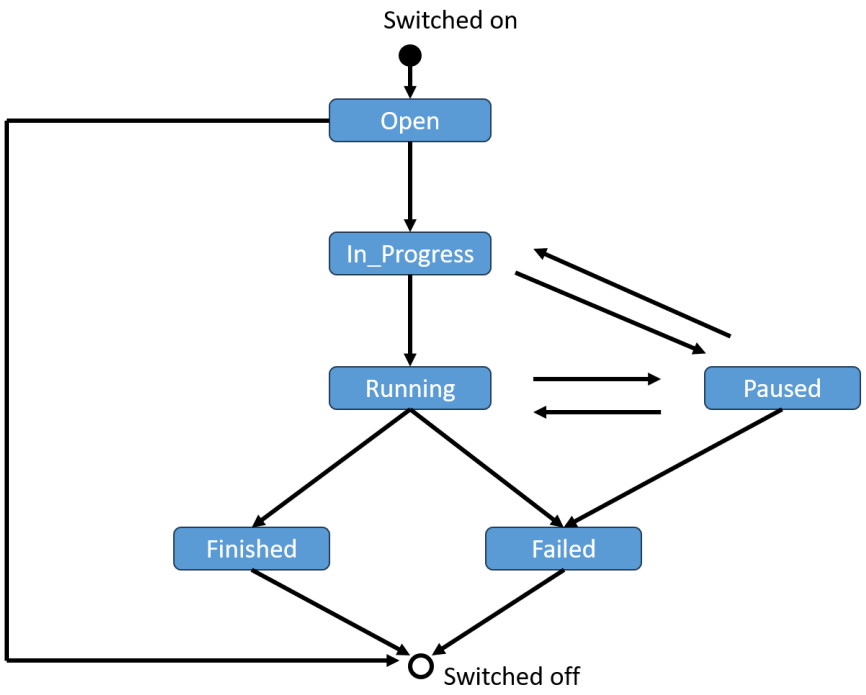

Abbildung 23: Zustandsdiagramm der Aufträge

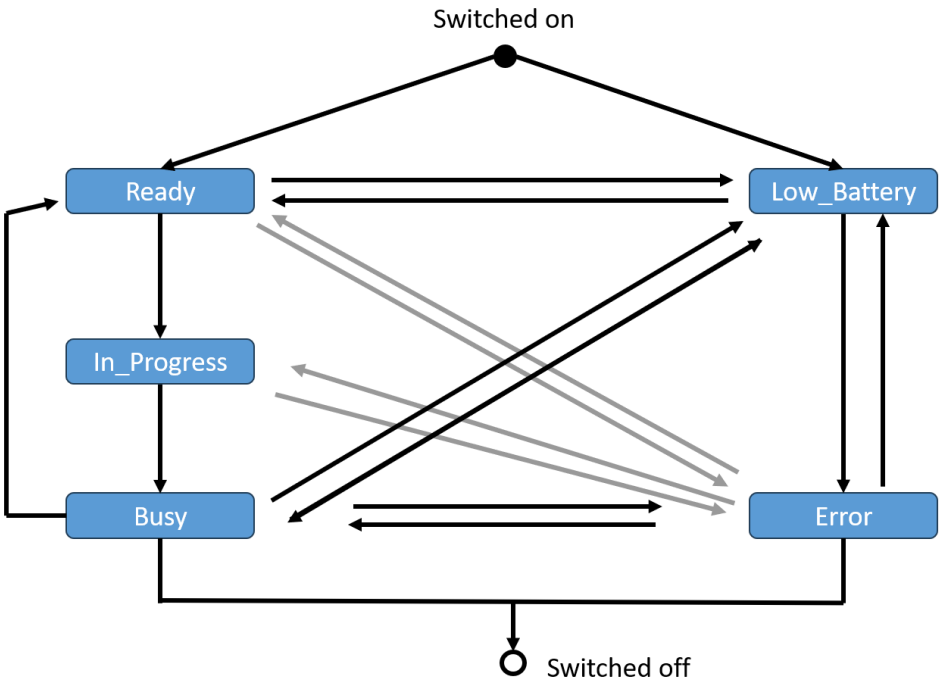

Abbildung 24: Zustandsdiagramm der Fahrzeugzustände

Die finale GUI besitzt eine Eingabemaske für Fahraufträge (Größe Paket/Ware, Zielort etc.), eine aktuelle Karte, welche bspw. die Position der Fahrzeuge in der Werkhalle visualisiert, eine Statusleiste mit einer Übersicht der aktuellen Fahraufträge samt Fahrzeuginformation (Dashboard), eine Übersicht der Aufträge, eine Konfigurationsmaske, eine Übersicht über den aktuellen Zustand der Fahrzeuge (Fleet Monitor) sowie ein Fenster für die Einstellungen. Nachfolgend sind das Dashboard (Abbildung 25), die Übersicht der aktuellen Fahraufträge (Abbildung 26), der Fleet Monitor (Abbildung 27) sowie die Live-Karte (Abbildung 28) dargestellt.

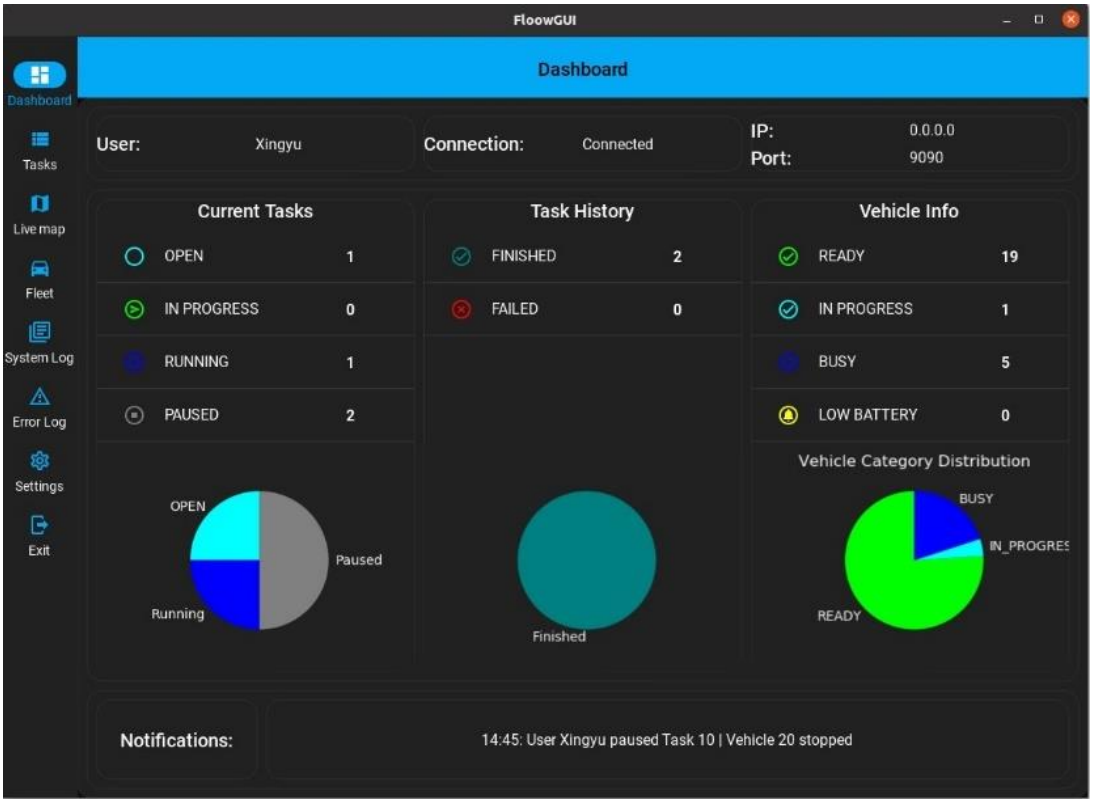

Abbildung 25: Dashboard der GUI

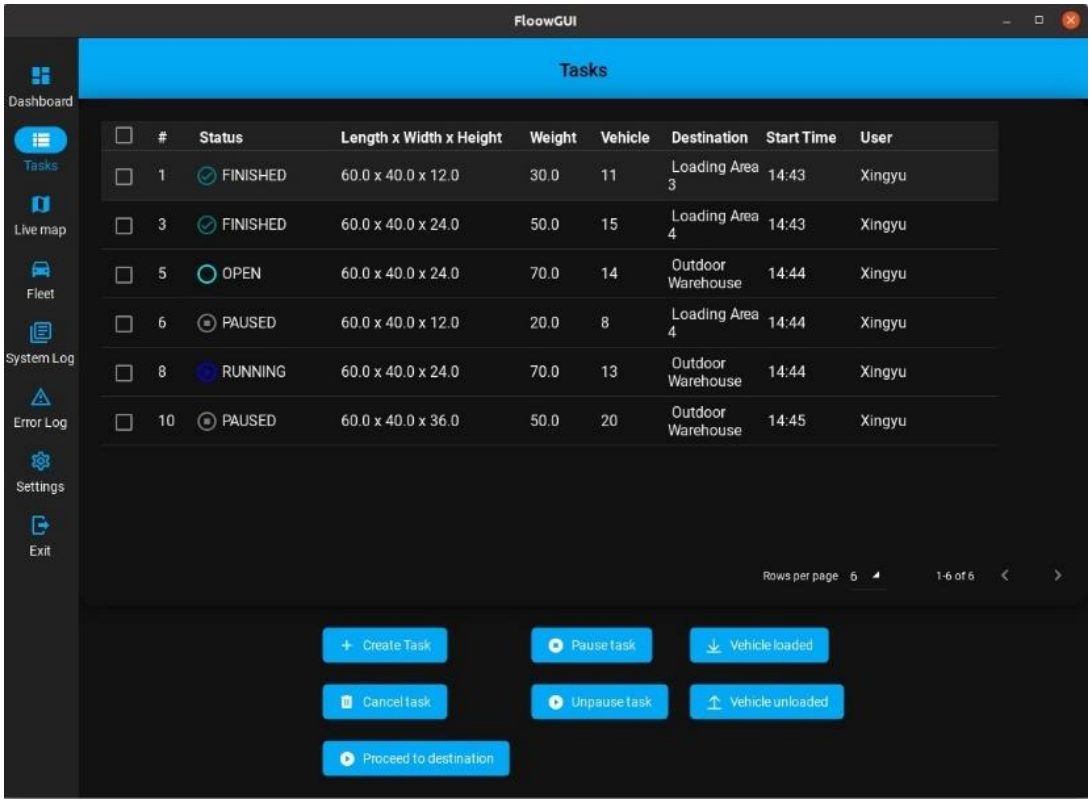

Abbildung 26: Übersicht der aktuellen Fahraufträge

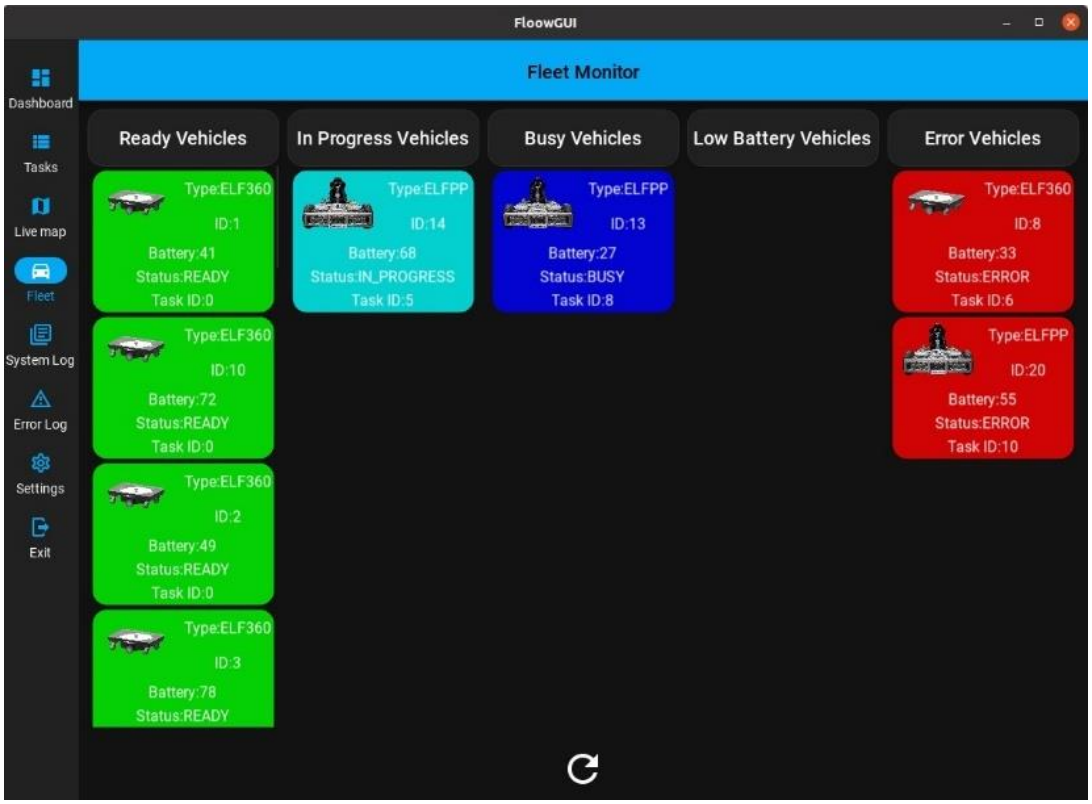

Abbildung 27: Fleet Monitor der GUI

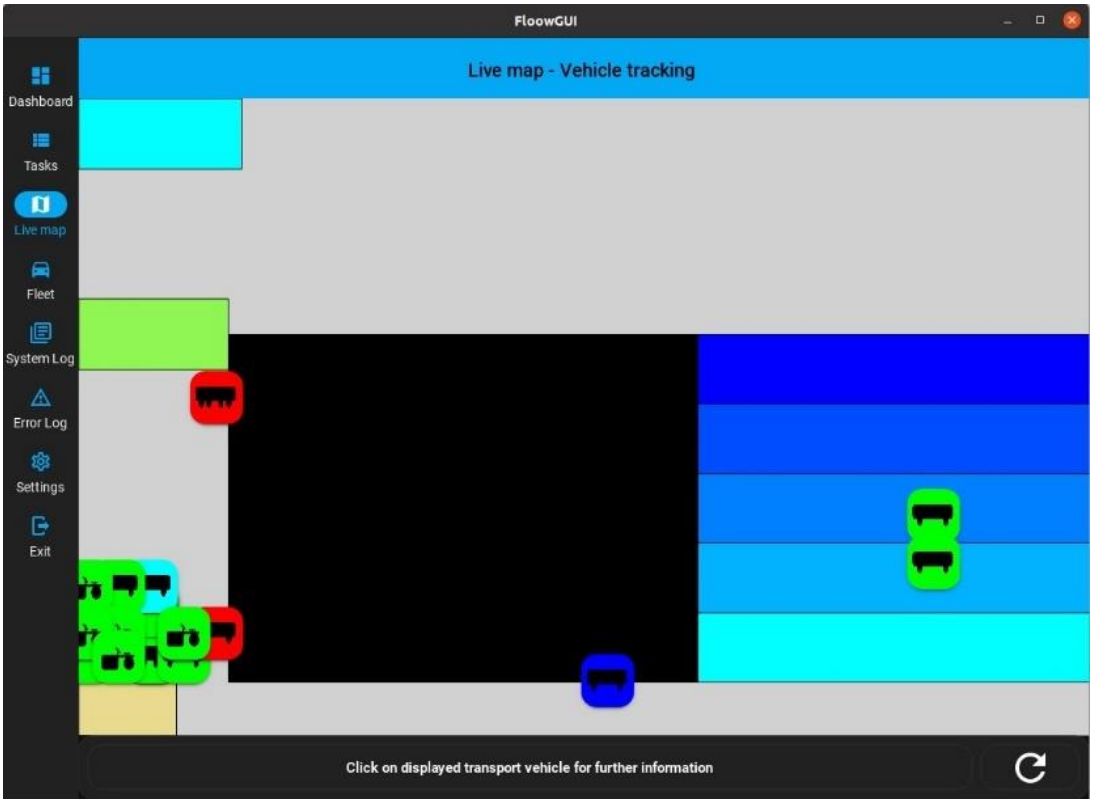

Abbildung 28: Live-Karte der GUI

Zur automatischen Erkennung der Paketgröße (Länge, Breite, Höhe) wurde eine Lösung mithilfe der im Tablet vorhandenen Kamera erarbeitet (basierend auf QR-Codes). Weiterhin wurde eine Lösung mittels eines KI-Algorithmus (automatische Größenerkennung durch Marker) in einer Testversion implementiert. Neben diesen zwei automatisierten Ansätzen, ist auch die Eingabe der Paketgrößen aufgrund standardisierter Euroboxen (Auswahl der Größe) oder auch per manueller Eingabe der Abmessungen möglich.

In Abbildung 29, Abbildung 30, Abbildung 31 und Abbildung 32 sind die verschiedenen Möglichkeiten zur Eingabe der Paketgröße verdeutlicht.

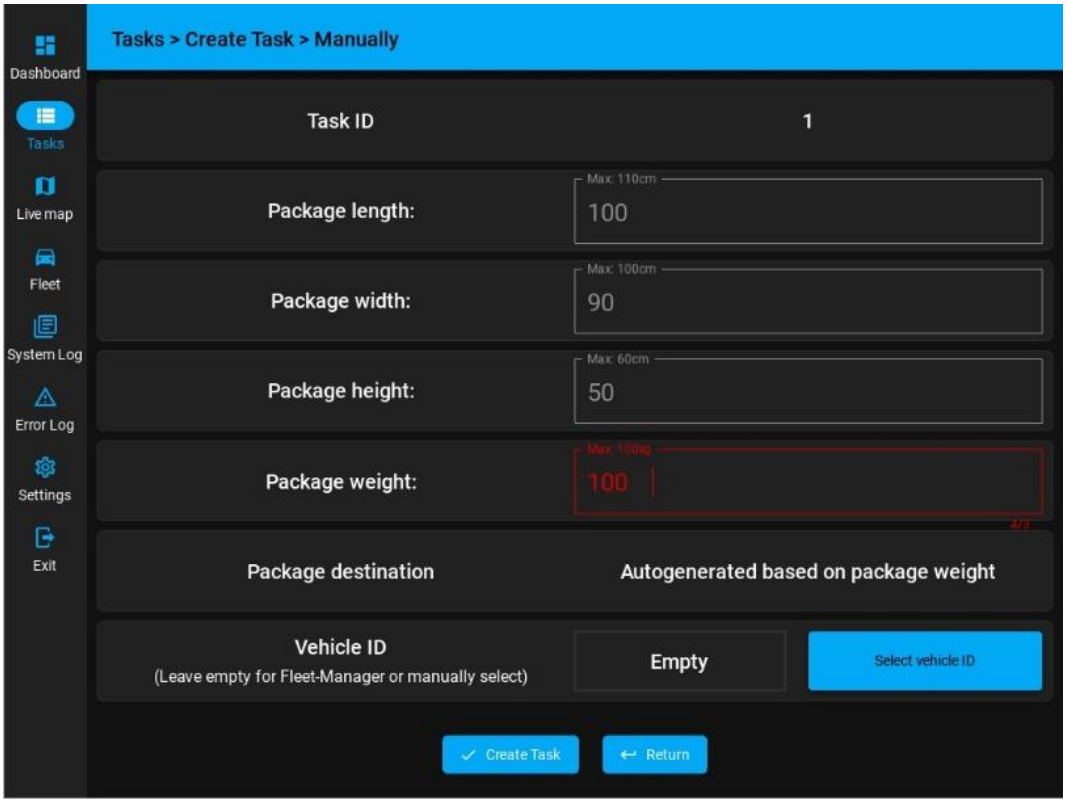

Abbildung 29: Manuelle Eingabe eines Fahrauftrages

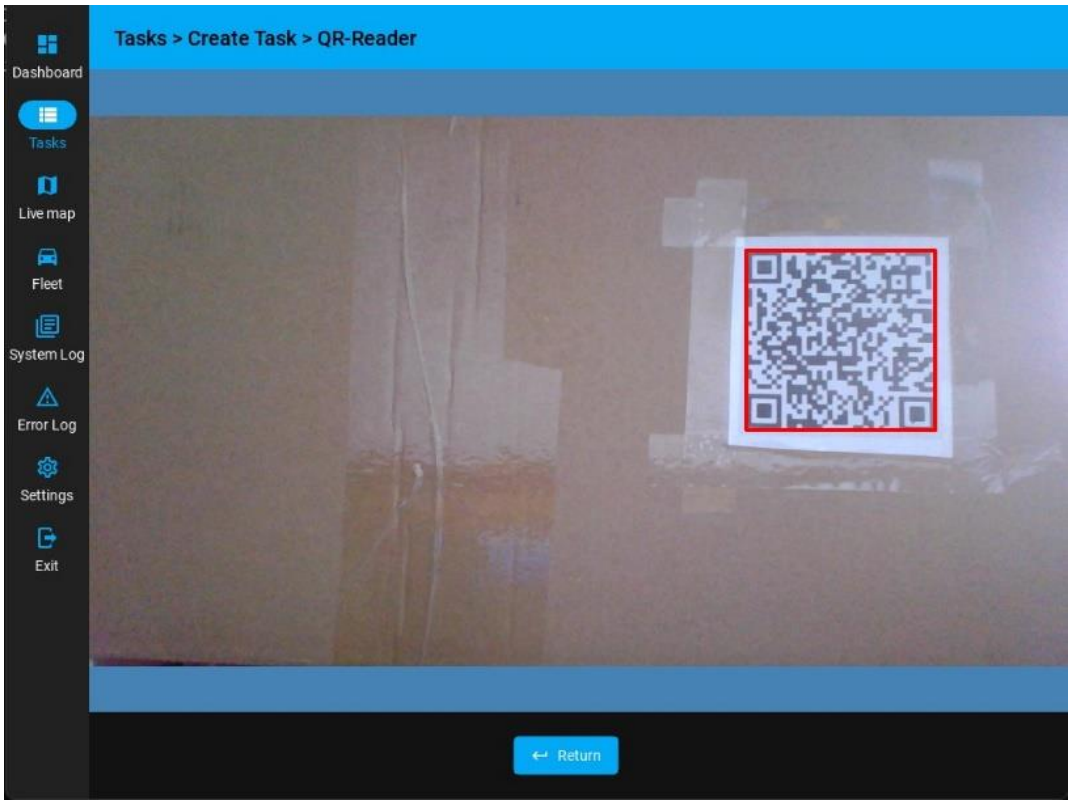

Abbildung 30: Scan eines QR-Codes zur Eingabe der Paketgröße

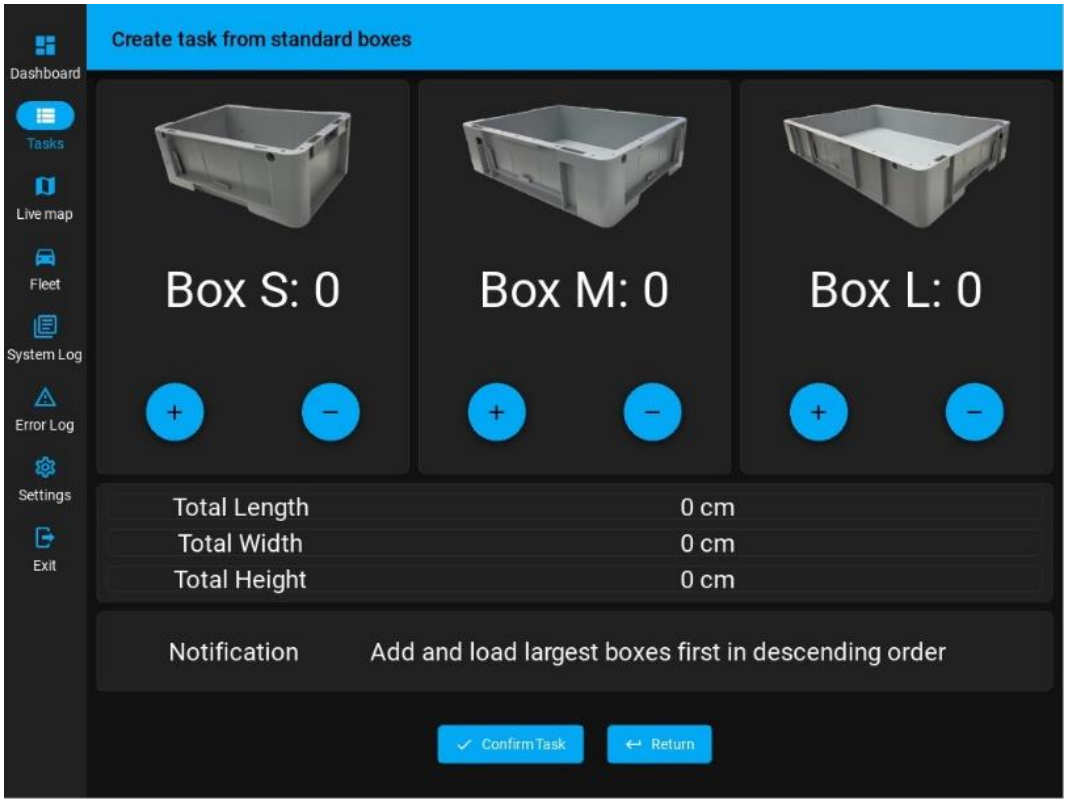

Abbildung 31: Eingabe der Paketgröße durch Auswahl genormter Euroboxen

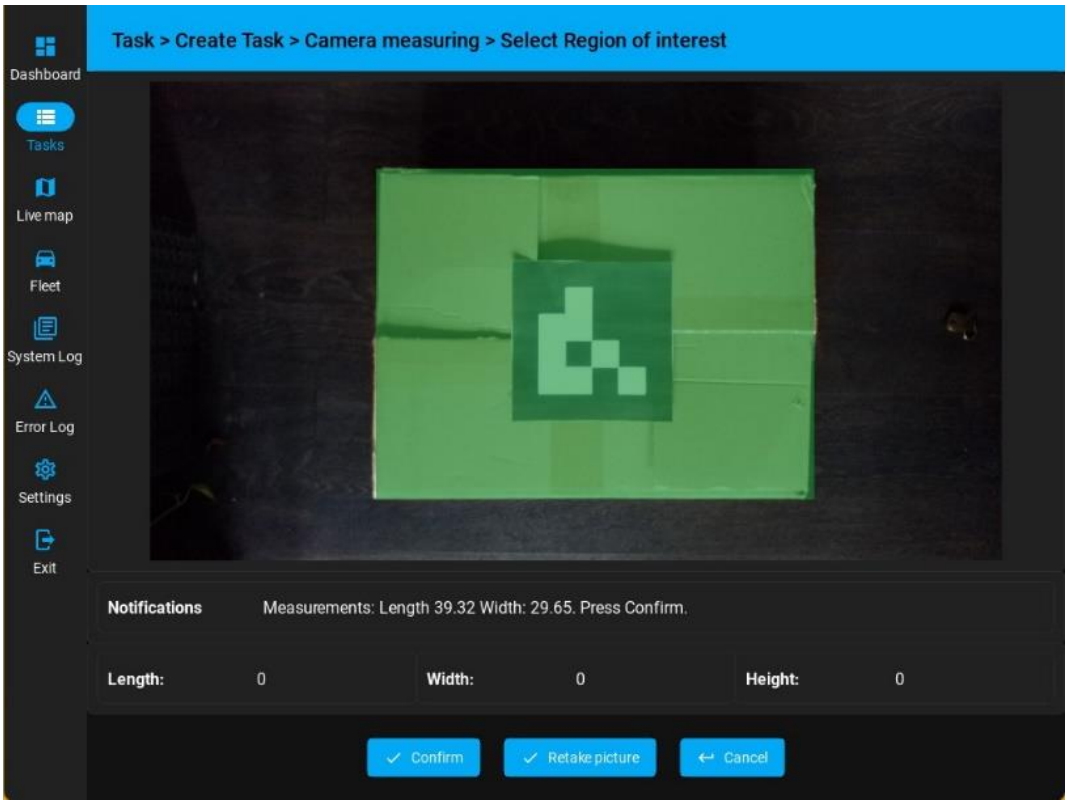

Abbildung 32: Eingabe der Paketgröße durch Kameraobjekterkennung (Aruco-Marker)

Die Funktion der GUI wurde in der Simulationsumgebung und auch in der Werkhalle validiert (Stabilität der App, Internetverbindung Werkhalle, Datenkommunikation etc.).

Entwicklung des KI-basierten Flottenmanagements

Im Anschluss wurde die GUI mit dem in AP7.2 entwickelten KI-basierten Flottenmanagement verknüpft. Für das Flottenmanagement wurde ein Reinforcement Learning Algorithmus gewählt, da dieser im Vergleich zu einem Supervised Ansatz, für eine interaktive System-umgebung geeignet ist. Grundlage des Algorithmus ist das Deep Q-Learning (Künstliche Neuronale Netze), welches per trial and error versucht, die Belohnung (Reward) für ausgeführte Aktionen zu erhöhen. Die Belohnung berücksichtigt die Einhaltung einer oberen (80%) und unteren Grenze (20%) des State of Charge (SoC) der Fahrzeugbatterie, die Auswahl des richtigen Fahrzeuges für die jeweilige Paketgröße/Ware (Transportfläche der Fahrzeuge), die Fahrzeit zur Erfüllung des Fahrauftrages sowie die Auswahl unterschiedlicher Fahrzeuge. In Tabelle 2 werden die jeweiligen technischen Parameter der Demonstrator-fahrzeuge aufgezeigt. Die Effizienz wurde beim CargoBike mit 100% angenommen, da dies durch einen Menschen betrieben wird.

Tabelle 2: Technische Parameter der Demonstratorfahrzeuge

Parameter	**ELF++**	**FTS**	**CargoBike**
Anzahl Fahrzeuge	10	10	5
Max. Geschwindigkeit [m/s]	8,0	0,5	5,6
Beschleunigungszeit [s]	3,0	0,5	4,0
Effizienz Antriebstrang [%]	96	76	100
Batteriekapazität [Ah]	100	6	50
Max. Cargogewicht [kg]	100	20	50

Die Validierung basierte auf zwei Simulationsdurchläufen, einerseits dem realen Energie-verbrauch, andererseits wurde der Energieverbrauch der Fahrzeuge künstlich erhöht, um eine Reaktion des Algorithmus zu provozieren (die Fahrstrecken sind zum Teil sehr kurz, sodass ein Energieverbrauch generell erst nach vielen aufeinanderfolgenden Fahraufträgen bzgl. dem

SoC von Bedeutung ist). Das Ergebnis bestätigte einen Lerneffekt des Agenten, so wurde der SoC der Batterien eingehalten und das Versuchsfahrzeug mit der niedrigsten Geschwindigkeit vermieden.

Der Algorithmus wurde im weiteren Verlauf optimiert und einem Stresstest unterzogen. Hierzu wurde die Python-Simulationsumgebung auf eine höhere Anzahl (25 Stück) an Versuchsfahrzeugen und Taktung der Fahraufträge (Steigerung der Komplexität) erweitert. Es erfolgte eine Variation der Hyperparameter, um eine bestmögliche Optimierung zu erreichen. Weiterhin wurde die eigentliche Aufgabe des Flottenmanagements (Auswahl der richtigen Fahrzeuge hinsichtlich Paketgröße, Ladezustand der Batterie etc.) nochmals untersucht und eine Anpassung der Randbedingungen vorgenommen.

Ein Vergleichs-Algorithmus, welcher rein regelbasiert agiert, wurde für die Erstellung eines Benchmarks entwickelt. Final fand ein Test beider Algorithmen für die sechs definierte Use-Cases aus AP6.1 statt. Der Vergleich ergab einen Vorteil des KI-basierten Ansatzes. Sowohl der SoC als auch die Transportzeit sowie die Nutzung passender Fahrzeuge entsprechend der Paketgröße wurden optimiert, der KI-basierte Algorithmus war in der Lage zu lernen und sich somit zu verbessern. Es wurde im Durchschnitt ein ca. 40% höherer Reward im Vergleich zum regelbasierten Algorithmus erzielt. Abbildung 33 zeigt den Lerneffekt des KI-basierten Algorithmus im Vergleich zum regelbasierten Algorithmus auf (Beschriftungen sind in Englisch aufgrund der internationalen Konventionen im Bereich des Reinforcement Learnings).

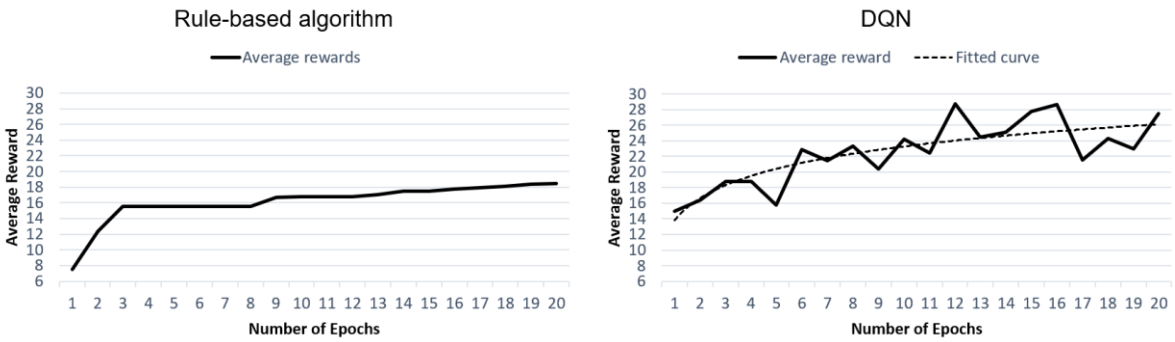

Abbildung 33: Vergleich regelbasierter Algorithmus (links) vs. KI-basierter Algorithmus (rechts)

In Abbildung 34 ist links die SoC Auslastung der jeweiligen Demonstratorfahrzeuge in einem Testdurchlauf aufgezeigt. Hierbei zeigt sich, dass die untere Grenze des SoC von 20% durchweg eingehalten wurde. Die höchste Auslastung weist das Demonstratorfahrzeug ELF++ auf. Korrespondierend hierzu verdeutlicht die rechte Abbildung den Verlauf der Transportzeit.

Unterschiede können aufgrund der verschiedenen Grundannahmen bzw. technischen Parametern der Demonstratorfahrzeuge (Höchstgeschwindigkeit, Batteriekapazität etc.) sowie des jeweiligen Paketgewichtes als auch der Fahrstrecke (Zielort) begründet sein.

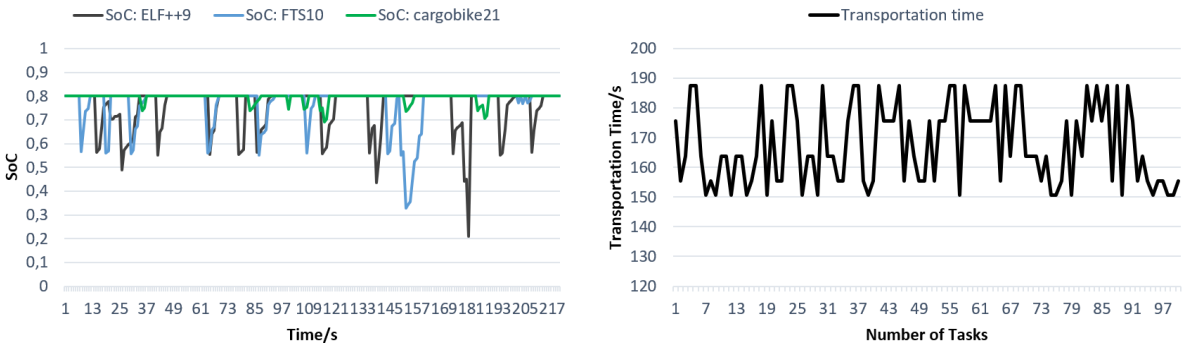

Abbildung 34: Vergleich der SoC Auslastung der Fahrzeuge (links) sowie der Transportzeit (rechts)

Bei den Ergebnissen muss die spezifische Testumgebung sowie die daraus jeweils resultierenden Randbedingungen beachtet werden. Bei sich verändernden Umgebungsbedingungen wäre ein neues Training bzw. eine Anpassung der Belohnungen und Bestrafungen notwendig. Weiterhin ist im vorliegenden Fall die Routenplanung vereinfacht angenommen sowie die Anzahl an Demonstratorfahrzeugen begrenzt. Eine Verbesserung des KI-basierten Algorithmus könnte durch eine Kombination mit einem regelbasierten Algorithmus oder eines anderen Reinforcement Learning Algorithmus (z.B. Proximal Policy Optimization) erreicht werden.

Sowohl die Entwicklung der GUI als auch die des KI-basierten Flottenmanagements wurden mit den Projektpartnern stetig abgesprochen. Hierbei lag der Fokus auf dem Austausch der verschiedenen Parameter (Position des Fahrzeuges, Ladezustand etc.) sowie auf der Implementierung in der realen Werkhalle. Zur Vorbereitung auf das Abschlussevent fanden im späteren Verlauf in regelmäßigen Abständen Realtests in der Werkhalle statt.

Integration der KI-Services in HAD-Plattform

In AP7.3 führten die Projektpartner Kommunikationstests zwischen den HAD-Plattformen, den Versuchsfahrzeugen und der Infrastruktur in der Werkhalle durch.

Die Entwicklung der Connectivity mit der Infrastruktur zielte darauf ab, effiziente Schnittstellen zwischen autonomen Fahrzeugen und einem Flottenmanagementsystem auf Serverseite zu implementieren. Hierbei wurde ROS-Websocket als Übertragungsschnittstelle verwendet. Die Schnittstellen ermöglichen die Kommunikation und Datenaustausch zwischen den autonomen Fahrzeugen und dem zentralen Flottenmanagementsystem.

Inbetriebnahme der Connectivity-Lösung und Einbindung in die Mobilitätsinfrastruktur

Für das AP7.4 erfolgte die Anpassung des KI-basierten Flottenmanagements an die finalen Systemumgebungen in der Werkhalle. Weiterhin wurden die Vorbereitungen in der Werkhalle zur Integration der Connectivity-Lösungen finalisiert (Anbringung eines WLAN Access Points außerhalb der Werkhalle). Der für den Projektmonat 30 datierte Meilenstein C („Integration erfolgt, Entwicklungsarbeiten evaluiert“) enthielt u.a. die Meilensteine 7.1 („User Interface implementiert und nutzbar“), 7.2 („KI-basiertes Flottenmanagement verfügbar“) sowie 7.3 („Connectivity-Lösung in die Mobilitätsinfrastruktur eingebunden“), welche alle erfolgreich abgeschlossen werden konnten. Die finale Validierung aller Software-Komponenten in Zusammenarbeit mit den beteiligten Projektpartnern in der Werkhalle wurde im Januar und Februar 2024 durchgeführt.

Die Anforderungsanalyse für die Entwicklung der Schnittstellen zwischen den autonomen Fahrzeugen und dem Flottenmanagementsystem basierte auf den spezifischen Anforderungen des Kunden, dem Projektpartner FAST. Zu den Hauptanforderungen gehörten:

- Übertragung von Fahrzeugzuständen wie SOC (State of Charge), Bereitschaft, Geschwindigkeit und Positionen in einem definierten Format.
- Implementierung eines Zustandsautomaten, um Zustandswechsel basierend auf Befehlen vom Server zu ermöglichen.

Es wurde eine ROS-Websocket-Schnittstelle entwickelt, um eine nahtlose Kommunikation zwischen den autonomen Fahrzeugen und dem Flottenmanagementsystem zu gewährleisten. Hierbei wurden ROS (Robot Operating System) und Websockets genutzt, um eine effiziente Übertragung der Daten zu ermöglichen. Ein Zustandsautomat wurde implementiert, um die Steuerung der Fahrzeugzustände basierend auf den Befehlen vom Server zu ermöglichen. Dieser Zustandsautomat berücksichtigt verschiedene Zustände wie Bereitschaft, Fahrmodi und Fehlerzustände und ermöglicht eine dynamische Anpassung der Fahrzeugsteuerung.

Zur Validierung der entwickelten Schnittstellen wurden umfangreiche Tests durchgeführt, die auf im Projekt definierten Use Cases basierten. Hierbei wurden sowohl automatisierte Tests als auch manuelle Tests durchgeführt, um die Funktionalität und Zuverlässigkeit der Schnittstellen sicherzustellen. Durch die Validierung wurde sichergestellt, dass die Schnittstellen den Anforderungen entsprechen und fehlerfrei funktionieren.

Die seitens des KIT-FAST zu erbringenden Meilensteine 7.1 „User Interface implementiert und nutzbar“, 7.2 „KI-basiertes Flottenmanagement verfügbar“ und 7.3 „Connectivity-Lösung in die Mobilitätinfrastruktur eingebunden“ wurden alle planmäßig erbracht.

2.1.9 AP8 – Integration und Evaluation

Das AP8 behandelte zum Abschluss die Integration und Evaluation aller Teilsysteme. Die Leitung oblag hierbei dem KIT-FAST. Das AP teilte sich in folgende TAP‘s auf:

- TAP 8.1: Konzeption des Anwendungstests
- TAP 8.2: Integration und Erprobung der Teilsysteme
- TAP 8.3: Durchführung Anwendungstest mit dem Gesamtsystem

In AP8 war das KIT-FAST in allen TAP’s beteiligt.

Konzeption der Anwendungstests

Im Rahmen von TAP8.1 wurden die Anwendungstest für TAP8.3 finalisiert. Dafür wurden die Aufgaben der verschiedenen Fahrzeuge definiert sowie die zur Demonstration zu verwendende Labor- beziehungsweise Werkhalle in Bereiche aufgeteilt und Fahrwege festgelegt (siehe ebenfalls AP6). Hierunter sind:

- ein Bereich zum Laden der Batterien (hypothetisch),
- ein Aufnahmebereich für die Güter,
- ein Outdoor-Bereich, welcher der Simulation einer zweiten Werkhalle dient, sowie
- fünf Zonen für das Abladen der Güter im Innenbereich der Werkhalle.

Durch den Outdoor-Bereich und den verschiedenen Eingangstoren der Werkhalle konnte zudem der Wechsel zwischen verschiedenen Werkhallen simuliert werden. Ebenfalls wurde ein x,y-Koordinatensystem eingeführt, welches mit den Projektpartnern bzgl. der Indoor-Lokalisierung besprochen wurde. Zur Georeferenzierung der Demonstratorfahrzeuge wurde approximativ eine GNSS-Karte erstellt. Hierbei wurde gemeinsam mit dem Projektpartner Schaeffler an zwei Positionen mithilfe eines GNSS-Receivers des Projektpartners ANavS reale GNSS-Marker gesetzt, siehe Abbildung 35.

Im nächsten Schritt wurden verschiedene Aufgaben definiert und den unterschiedlichen Fahrzeugen zugeordnet. Hierbei musste beachtet werden, dass aufgrund der unterschiedlichen Abmessungen und Fahreigenschaften nicht jedes Fahrzeug in der Lage war, jeden Fahrweg zu absolvieren. So konnte das FTS beispielsweise nur in der Halle betrieben werden, wohingegen das ELF++ vor allem für den Übergangsbereich zwischen Indoor und Outdoor ausgewählt wurde. Das Cargo-Bike (Lastenfahrrad) wurde für die finalen Versuche in der Werkhalle als Hindernis angenommen und war somit nicht mehr aktiv in die Tests eingebunden.

Abbildung 35: Aufzeichnung von GNSS-Marker zur Erstellung einer georeferenzierten Karte der Werkhalle

Bei den angewendeten Use-Cases (siehe AP6) startete jedes Fahrzeug im Batterieladebereich und fuhr von dort über unterschiedliche Wege zum Paketaufnahmebereich. Sobald das Paket aufgenommen wurde, fuhr das jeweilige Fahrzeug zu einer definierten Abladestelle und im Anschluss wieder zum Batterieladebereich.

Die Kommunikation zwischen dem Server, Tablet (GUI), KI-Flottenmanagement und den Versuchsfahrzeugen erfolgte über das KIT-WLAN und dem Betriebssystem ROS2 (Robot Operating System) sowie einer ROSbridge. Ein Konzept der Datenkommunikation wurde erarbeitet und mit den Projektpartnern abgestimmt. Die Versuchsfahrzeuge bzw. das Sensor Setup tauschen die internen Parameter wie Position des Fahrzeuges, Gierrate, Fehlermeldungen mit dem zentralen Server und der GUI aus. Dem KI-Flottenmanagement werden die aktuell verfügbaren Fahrzeuge sowie die Fahraufträge von der GUI gemeldet. In Abbildung 36 ist die Datenkommunikation der einzelnen Komponenten dargestellt.

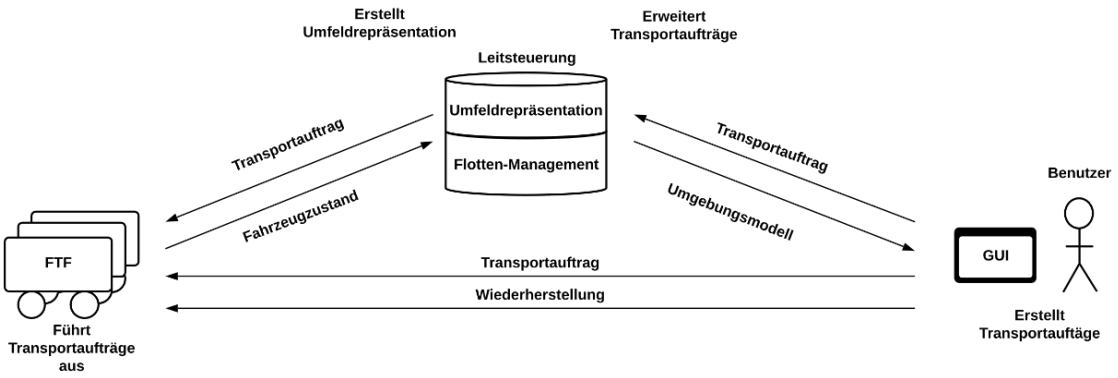

Abbildung 36: Datenkommunikation Fahrzeuge, Server und GUI

Integration und Erprobung der Teilsysteme

In TAP8.2 wurden mit dem entwickelten Konzept aus TAP8.1 Tests gemeinsam mit dem Projektpartner Schaeffler per Computersimulation durchgeführt. Weiterhin wurde die WLAN-Abdeckung in und vor der Werkhalle getestet sowie die Datenkommunikation mit dem Server validiert. Für eine bessere Abdeckung wurde daraufhin im Außenbereich der Laborhalle ein zusätzlicher WLAN Access Point angebracht, siehe Abbildung 37. Ebenfalls wurden die Termine für die Praxistests in der Werkhalle organisiert.

Abbildung 37: Neu installierter WLAN Access Point außerhalb der Werkhalle

Durchführung der Anwendungstest im Gesamtsystem

Im Rahmen des AP 8.3 wurden die zuvor erfolgreich integrierten Teilsysteme im Gesamtverbund getestet. Die Treffen zur Durchführung des AP fanden dabei primär nur noch vor Ort in der Laborhalle am Kit Campus Ost statt. Der Übergang von AP 8.2 zu AP 8.3 war dabei fließend. Während der Treffen vor Ort in der Laborhalle des KIT-FAST wurden die Anwendungsfälle aus AP 8.1 erprobt und mögliche Herausforderungen bei den Gesamtsystemtests identifiziert sowie Lösungsmöglichkeiten erarbeitet und umgesetzt. So wurden u.a. alle Anwendungstests schon vor dem Projektabschlussevent erfolgreich durchgeführt und u.a. mithilfe von Kameras aufgezeichnet, da im Falle von Regen (die Versuchsfahrzeuge sind nicht für Regen ausgelegt) beim Abschlussevent die Outdoor-Szenarien nicht durchgeführt werden konnten. Weiterhin wurde im Rahmen dieses AP das Abschlussevent vorbereitet und dafür u.a. die Laborhalle für die finalen Anwendungstests umgeräumt und somit die simulierte Werkhalle aufgebaut. Das KIT-FAST unterstützte die Arbeiten der Projektpartner vor Ort und gewährleistete ausreichend Testmöglichkeiten.

Aufgrund des hohen Maßes an Zusammenarbeit und Abhängigkeiten in AP8 sei für eine detaillierte Übersicht auf den gemeinsamen Schlussbericht des Konsortiums verwiesen.

Die Meilensteine 8.1 „Konzept für Anwendungstests liegt vor“, 8.2 „Teilsysteme in Gesamtsystem integriert und erfolgreich getestet“ sowie 8.3 „Anwendungstests des Gesamtsystems erfolgreich abgeschlossen“ wurden in AP8 seitens des KIT-FAST erbracht.

2.2 Wichtigste Positionen des zahlenmäßigen Nachweises

Die wesentlichen Kosten des Projektes entfallen auf die Personalkosten (ca. 94%). Einen weiteren großen Posten stellt mit 5% die Beschaffung eines High-Performance Computers dar, welcher zur Datenaufzeichnung, Datenspeicherung und insbesondere zum Training von Objekterkennungsalgorithmen mittels Kamera zum Pre-Processing der Daten für das Testen der Kritikalitätsalgorithmen eingesetzt wurde.

2.3 Notwendigkeit und Angemessenheit der geleisteten Arbeit

Der Verlauf der Arbeiten im Projekt folgte der im Projektantrag formulierten Planung, so dass alle im Arbeitsplan formulierten Aufgaben erfolgreich bearbeitet wurden. Dabei waren gegenüber der ursprünglichen Planung keine zusätzlichen Ressourcen für das Projekt notwendig.

2.4 Voraussichtlicher Nutzen

Die im Rahmen des Verbundvorhabens FLOOW erarbeiteten Ergebnisse und Erfahrungen haben für das KIT-FAST folgende zukünftige Nutzen:

- Durch die bei der Erforschung des KI-basierten Flottenmanagements gewonnenen Erkenntnisse im Bereich des Reinforcement Learning für Mobilitätssysteme, findet diese Art von Algorithmen auch in zukünftigen Forschungen wie bspw. Fahrzeugregelung Anwendung. Hierbei ist der grundlegende Ansatz des Algorithmus nutzbar und wird dementsprechend entsprechend modifiziert.
- Das Wissen, welches bei der Entwicklung der GUI gewonnen wurde, wird in zukünftigen Projekten bspw. beim Labeling von Datensätzen für das autonome Fahren eingesetzt.
- Auswahl von Simulationssoftware für zukünftige Projekte und Forschungen auf Basis der Erfahrungen mit der Software Gazebo und MATLAB/Simulink. Insbesondere hinsichtlich des autonomen Fahrens wie bspw. realitätsnahe synthetische Datensätzen für die Kameraobjekterkennung oder Simulation der Fahrphysik.
- Möglichkeit für Studierende (Abschlussarbeiter und studentische Hilfskräfte) KI-Algorithmen, Simulationsumgebungen sowie eine GUI für Mobilitätsysteme zu entwickeln und in einem realen Umfeld zu testen.
- Die gewonnen Erkenntnisse aus dem Projekt werden in die Inhalte bestehender Lehrveranstaltungen des Instituts (bspw. „Grundlagen der Fahrzeugtechnik“) einfließen.
- Auf Basis der erarbeiteten Softwarelösungen sowie des gewonnenen Wissens können zukünftig weitere Forschungsprojekte oder Industriekooperationen initiiert bzw. gestaltet werden.

2.5 Fortschritte bei anderen Stellen

Außer dem im Kapitel 1.4 „Wissenschaftlicher und technischer Stand“ dargestellten Inhalten wurden im weiteren Projektverlauf keine weiteren Fortschritte auf dem Gebiet des Vorhabens, insbesondere beim für das Teilvorhaben des KIT-FAST relevanten Flottenmanagement identifiziert.

2.6 Veröffentlichung der Ergebnisse

Teile der Inhalte des Verbundprojektes FLOOW wurden in einer wissenschaftlichen Veröffentlichung im Anschluss an das Projekt veröffentlicht. Patente oder Erfindungsmeldungen wurden nicht eingereicht. Im Rahmen des Vorhabens wurden jedoch mehrere Abschlussarbeiten betreut.

2.6.1 Veröffentlichungen in wissenschaftlichen Zeitschriften und bei Konferenzen

- Abdelhalim, G., Simon, K., Bensch, R., Parimi, S., Qureshi, B.-A., „Automated AI-based Annotation Framework for 3D Object Detection from LIDAR Data in Industrial Areas“, Society of Automotive Engineers (SAE) Technical Paper, Bd. 2024-01–2999, S. 14, Juli 2024, doi: 10.4271/2024-01-2999.

2.6.2 Veröffentlichungen im Kontext der Öffentlichkeitsarbeit

Gemeinsam mit den Projektpartnern wurden regelmäßig Informationen über den Projektfortschritt und besondere Ereignisse (bspw. Konsortialtreffen) auf der gemeinsamen Projektwebsite (https://floow-project.de/) veröffentlicht. Darüber hinaus gab es auch auf dem Internetauftritt des Instituts eine Seite (https://www.fast.kit.edu/lff/Projekte_FLOOW.php) zum Verbundprojekt FLOOW. Außerdem waren die Konsortialpartner FZI sowie KIT-FAST gemeinsam bei der Hannover Messe 2023 vertreten und haben dabei das Verbundvorhaben FLOOW präsentiert.(https://www.sts.kit.edu/hannovermesse2023.php). Unter den zahlreichen Besuchern befanden sich beispielsweise auch der Bundesminister für Digitales und Verkehr, Volker Wissing, die Ministerin für Wirtschaft, Arbeit und Tourismus des Landes Baden-Württemberg, Dr. Nicole Hoffmeister-Kraut sowie Maive Rute, Directorate-General for Internal Market, Industry, Entrepreneurship and SMEs der Europäischen Kommission.

Abbildung 38: Impressionen vom Gemeinschaftsstand auf der Hannover Messe 2023

2.6.3 Abschlussarbeiten

Im Rahmen des Projektes wurden insgesamt sieben Abschlussarbeiten am KIT-FAST mit Bezug zum Verbundprojekt FLOOW angefertigt:

- Entwicklung eines KI-basierten Flottenmanagements für autonome Logistikfahrzeuge (Wang, 2022)
- Weiterentwicklung eines KI-basierten Flottenmanagements für autonome Logistikfahrzeuge (Pernice, 2023)
- Entwicklung eines KI-basierten Annotationstools für das Labeling von Objektdatensätzen (Abdelhalim, 2023)
- Entwicklung eines Touchscreen User-Interface für ein KI-basiertes Mobilitätssystem (Al Ayoubi, 2023)
- Implementierung einer Connectivity-Lösung für ein autonomes Mobilitätssystem (Liao, 2024)

3 Verzeichnisse

3.1 Literaturverzeichnis

[1] T. Xue, P. Zeng und H. Yu, „A reinforcement learning method for multi-AGV scheduling in manufacturing," in *2018 IEEE International Conference on Industrial Technology (ICIT)*, Lyon, 2018.

[2] J. Cestero, M. Quartulli, A. M. Metelli und M. Restelli, „Storehouse: a Reinforcement Learning Environment for Optimizing Warehouse Management," in *2022 International Joint Conference on Neural Networks (IJCNN)*, Padua, 2022.

[3] W. Zhang, Q. Wang, J. Li und C. Xu, „Dynamic Fleet Management With Rewriting Deep Reinforcement Learning," *IEEE Access,* Bd. 8, pp. 143333-143341, 2020.

[4] K. Lin, R. Zhao, Z. Xu und J. Zhou, „Efficient Large-Scale Fleet Management via Multi-Agent Deep Reinforcement Learning," *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,* pp. 1774-1783, 2018.

[5] C. Han, M. Frey und F. Gauterin, „Modular Approach for Odometry Localization Method for Vehicles with Increased Maneuverability," *Sensors,* 2021.

3.2 Abbildungsverzeichnis

Abbildung 1: Die Demonstratorfahrzeuge in einer Werkhalle. (Bilder: Alexander Seiffer, SHARE) 8

Abbildung 2: Vergleich der realen Prozesskette (oben) mit der virtuellen Prozesskette (unten). 12

Abbildung 3: Außenansicht der virtuellen Werkhalle. 13

Abbildung 4: Außenansicht der realen Werkhalle am Campus Ost des KIT in Karlsruhe. 13

Abbildung 5: Innenansicht der Werkhalle mit statischen und dynamischen Objekten. 14

Abbildung 6: Darstellung der drei Demonstratorfahrzeuge ELF++, FTS und Lastenrad ... 14

Abbildung 7: Außenansicht der Werkhalle mit einem FTS, zwei ELF++ sowie zwei Gabelstaplern im Außenbereich vor der Werkhalle. 15

Abbildung 8: Erstellte Straßennetze im Bereich der Werkhalle mithilfe von Lanelet2. 15

Abbildung 9: Aufgezeichnete Punktwolke eines LiDAR-Sensors in Gazebo. 16

Abbildung 10: Schematische Darstellung der generellen Modellstruktur des ELF++. 17

Abbildung 11: Schematische Darstellung des ELF++-Fahrzeugmodells. 19

Abbildung 12: Schematische Darstellung der generellen Modellstruktur des FTS sowie Lastenrads. 20

Abbildung 13: Schematische Darstellung des Fahrzeugmodells für FTS und Lastenrad. . 21

Abbildung 14: Signalverknüpfungen in der gesamtheitlichen Simulationsumgebung 22

Abbildung 15: Setup zur Aufzeichnung der LiDAR-Daten in der Werkhalle des KIT 23

Abbildung 16: Aufgezeichnete Punktwolke mit 16-Channel LiDAR (links) und mit dem LiDAR Super Resolution Algorithmus erzeugte upsampled Punkwolke (rechts) 24

Abbildung 17: Beispiele der verschiedenen Objektklassen mit 3D Punktwolken- und den entsprechenden Kamerabildern. 25

Abbildung 18: 3D-Bounding Boxen Prädiktion der LiDAR-Objekterkennung 26

Abbildung 19: GUI für das Labeling des 3D LiDAR Datensatzes. 27

Abbildung 20: Automatisiertes KI-basiertes Annotations-Framework. Von manueller zur automatischen Annotation (von oben nach unten) basierend auf niedrig aufgelösten LiDAR-Daten. 27

Abbildung 21: Testbereiche der Werkhalle in der Simulationsumgebung 29

Abbildung 22: Testbereiche der Werkhalle im Detail 29

Abbildung 23: Zustandsdiagramm der Aufträge 32

Abbildung 24: Zustandsdiagramm der Fahrzeugzustände 32

Abbildung 25: Dashboard der GUI 33
Abbildung 26: Übersicht der aktuellen Fahraufträge 34
Abbildung 27: Fleet Monitor der GUI 34
Abbildung 28: Live-Karte der GUI 35
Abbildung 29: Manuelle Eingabe eines Fahrauftrages 36
Abbildung 30: Scan eines QR-Codes zur Eingabe der Paketgröße 36
Abbildung 31: Eingabe der Paketgröße durch Auswahl genormter Euroboxen 37
Abbildung 32: Eingabe der Paketgröße durch Kameraobjekterkennung (Aruco-Marker) 37
Abbildung 33: Vergleich regelbasierter Algorithmus (links) vs. KI-basierter Algorithmus (rechts) 39
Abbildung 34: Vergleich der SoC Auslastung der Fahrzeuge (links) sowie der Transportzeit (rechts) 40
Abbildung 35: Aufzeichnung von GNSS-Marker zur Erstellung einer georeferenzierten Karte der Werkhalle 43
Abbildung 36: Datenkommunikation Fahrzeuge, Server und GUI 44
Abbildung 37: Neu installierter WLAN Access Point außerhalb der Werkhalle 44
Abbildung 38: Impressionen vom Gemeinschaftsstand auf der Hannover Messe 2023 47

3.3 Tabellenverzeichnis

Tabelle 1: Testergebnisse Objekterkennung für die aufgezeichneten LiDAR-Daten in der Werkhalle. 26

Tabelle 2: Technische Parameter der Demonstratorfahrzeuge 38

Berichtsblatt

1. ISBN oder ISSN noch zu vergeben	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht
3. Titel Schlussbericht zum Verbundvorhaben: Flexibles Mobilitäts- und Cargo-System für den Werksverkehr (FLOOW), Teilvorhaben: Fahrzeugplattformen und Connectivity	
4. Autor(en) [Name(n), Vorname(n)] Bearbeiter Frey, Michael; Rautenberg, Philip; Simon, Kevin; Ellwanger, Garvin	5. Abschlussdatum des Vorhabens 29.02.2024 6. Veröffentlichungsdatum 7. Form der Publikation Schlussbericht
8. Durchführende Institution(en) (Name, Adresse) Karlsruher Institut für Technologie (KIT) Institut für Fahrzeugsystemtechnik (FAST) Kaiserstr. 12 76131 Karlsruhe	9. Ber. Nr. Durchführende Institution 10. Förderkennzeichen 19A20025C 11. Seitenzahl 51
12. Fördernde Institution (Name, Adresse) Bundesministerium für Wirtschaft und Klimaschutz (BMWK) 53107 Bonn	13. Literaturangaben 5 14. Tabellen 2 15. Abbildungen 38
16. Zusätzliche Angaben	
17. Vorgelegt bei (Titel, Ort, Datum)	

18. Kurzfassung

Im Projekt FLOOW wurden die zentralen Vorteile der Künstlichen Intelligenz (KI) zur Schaffung neuer Lösungen für die Mobilität von Menschen und Gütern genutzt. Der Fokus lag dabei auf einer robusten und hochgenauen Lokalisierung von Fahrzeugen im Mobilitätssystem, welche für Güter- oder Personentransport im Werksverkehr eingesetzt werden, sowie einer generalisierten Umfelderkennung als auch einer risikobewussten Manöverplanung auf dedizierter Hardware zur energieeffizienten Lösung der komplexen Teilprobleme auf den Fahrzeugen.

Das hier beschriebene Teilvorhaben trug zur Erarbeitung der nachfolgend genannten Projektergebnisse bei.

Zunächst erfolgte eine Analyse von Anwendungsfällen zur Evaluation der Multi-Sensor-Lokalisierung und der KI-basierten Umfelderkennung. Zusätzlich wurden Anforderungen gesammelt sowie eine Methodik zur Bewertung sowie bei der methodischen Analyse von Systemabhängigkeiten und der Definition von Schnittstellen erarbeitet. Zur Entwicklung einer hochgenauen und zuverlässigen Outdoor-Lokalisierung wurde vom Teilvorhaben die Erstellung der Odometrie-Modelle eingebracht. Als Grundlage für die Entwicklung und Erprobung der KI-basierten Algorithmen wurden eine dedizierte Simulationsumgebung für die im Projekt verwendeten Demonstratorfahrzeuge sowie digitale Karten des Campusgeländes und der Laborhalle erstellt. Dabei wurden auch die notwendigen Schnittstellen für die Kommunikation und Interaktion der Fahrzeugmodelle mit den Algorithmen der diversen anderen Arbeitspakete bereitgestellt. Bei der Konzeptionierung von Chassis-Systemen für kombinierten Indoor- und Outdoorbetrieb konnten u.a. die Erfahrungen aus den Verbundprojekten e²-Lenk und OmniSteer eingebracht werden.

Des Weiteren wurden die funktionalen Anforderungen an derartige Chassis-Systeme abgeleitet und die Auslegung der zugehörigen Regelungs- und Automatisierungsansätze unterstützt. Diese wurden dann in die Simulationsumgebung integriert. Im Arbeitspaket „Mobilitätssystem und Connectivity (KI)“ wurde die entsprechende Rechner- und Kommunikationstechnik aufgebaut. Zusätzlich wurde ein User-Interface für eine intuitive Bedienung zur Eingabe der Aufgabe bzw. des Fahrziels auch durch ungeübte Nutzer erstellt. Um die Daten für die beim Flottenmanagement eingesetzten KI-Verfahren zu generieren, wurde ein kombiniertes Belegungs- bzw. Nachfragemodell für die in der vereinfachten Erprobungsumgebung vorhandenen Streckenabschnitte erstellt. Dieses bildete die Basis für die Bereitstellung bzw. Verwaltung der notwendigen Daten, um daraus mit Hilfe einer KI-basierten Auswertung bzw. Optimierung eine Verteilungsstrategie der Flottenfahrzeuge hinsichtlich der Verfügbarkeit, der Bereitstellungszeit und der Gesamtauslastung zu erarbeiten. Im Rahmen der „Integration und Evaluation“ fanden eine kontinuierliche und regelmäßige Integration sowie Tests von Teilsystemen statt. Mit fortlaufender Projektdauer wurden die einzelnen Teilsysteme größer und umfassender, bis letztlich am Ende die Integration sowie die Tests aller Teilsysteme in das Gesamtsystem erfolgte. Schließlich erfolgte die Evaluation des Gesamtsystems anhand eines realitätsnahen Anwendungstests in der vereinfachten Einsatzumgebung (Simulation eines Werksgeländes auf dem Campus Ost des KIT).

19. Schlagwörter	
automatisierte Fahrzeuge, Lokalisierung, künstliche Intelligenz (KI), Flottenmanagement	
20. Verlag	21. Preis
nicht zutreffend	nicht zutreffend

xDocument Control Sheet

1. ISBN or ISSN to be defined	2. type of document (e.g. report, publication) Final report
3. title Schlussbericht zum Verbundvorhaben: Flexibles Mobilitäts- und Cargo-System für den Werksverkehr (FLOOW), Teilvorhaben: Fahrzeugplattformen und Connectivity (in German language) Translation: Final report on the joint project: Flexible mobility and cargo system for plant transport (FLOOW), sub-project: Vehicle platforms and connectivity	
4. author(s) (family name, first name(s)) Frey, Michael; Rautenberg, Philip; Simon, Kevin; Ellwanger, Garvin	5. end of project 29.02.2024 6. publication date 7. form of publication Final report
8. performing organization(s) (name, address) Karlsruher Institut für Technologie (KIT) Institut für Fahrzeugsystemtechnik (FAST) Kaiserstr. 12 76131 Karlsruhe	9. originator's report no. 10. reference no. 19A20025C 11. no. of pages 51
12. sponsoring agency (name, address) Bundesministerium für Wirtschaft und Klimaschutz (BMWK) 53107 Bonn	13. no. of references 5 14. no. of tables 2 15. no. of figures 38
16. supplementary notes	
17. presented at (title, place, date)	

18. abstract

The FLOOW project utilised the key advantages of artificial intelligence (AI) to create new solutions for the mobility of people and goods. The focus was on robust and highly accurate localisation of vehicles in the mobility system that are used for goods or passenger transport in factory traffic, as well as generalised environment recognition and risk-aware manoeuvre planning on dedicated hardware for energy-efficient solutions to the complex sub-problems on the vehicles.

The sub-project described here contributed to the development of the following project results.

The first step was to analyse use cases for evaluating multi-sensor localisation and AI-based environment recognition. In addition, requirements were collected and a methodology was developed for the evaluation and methodical analysis of system dependencies and the definition of interfaces. The sub-project contributed the creation of odometry models for the development of highly accurate and reliable outdoor localisation. A dedicated simulation environment for the demonstrator vehicles used in the project and digital maps of the campus grounds and the laboratory hall were created as the basis for the development and testing of the AI-based algorithms. The necessary interfaces for the communication and interaction of the vehicle models with the algorithms of the various other work packages were also provided. Experience from the joint projects e²-Lenk and OmniSteer was utilised in the design of chassis systems for combined indoor and outdoor operation. Furthermore, the functional requirements for such chassis systems were derived and the design of the associated control and automation approaches was supported. These were then integrated into the simulation environment. The corresponding computer and communication technology was set up in the 'Mobility system and connectivity (AI)' work package. In addition, a user interface was created for intuitive operation for entering the task or driving destination, even by inexperienced users.

In order to generate the data for the AI procedures used in fleet management, a combined occupancy and demand model was created for the existing route sections in the simplified test environment. This formed the basis for the provision and management of the necessary data in order to develop a distribution strategy for the fleet vehicles in terms of availability, provision time and overall utilisation with the help of an AI-based evaluation and optimisation. Continuous and regular integration and testing of subsystems took place as part of the ‘integration and evaluation’ process. As the project progressed, the individual subsystems became larger and more comprehensive until finally all subsystems were integrated and tested in the overall system. Finally, the overall system was evaluated on the basis of a realistic application test in the simplified operating environment (simulation of a factory site on the East Campus of the KIT).

19. keywords
Automated vehicles, localisation, artificial intelligence (AI), fleet management

20. publisher	21. price
not applicable	not applicable