

► VVM - Schlussbericht

VVMethoden - Verifikations- und Validierungsmethoden automatisierter Fahrzeuge Level 4 und 5



FZI Forschungszentrum Informatik

Version: 1.0

Zuwendungsempfänger: FZI Forschungszentrum Informatik

Förderkennzeichen: 19A19002L

Laufzeit des Vorhabens: 07/2019 - 12/2023



Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

Dokumenteninformation

Autoren

Thilo Braun, FZI
Julian Fuchs, FZI
Helen Gremmelmaier, FZI
Stefan Otten, FZI
Hubert Padusinski, FZI
Sebastian Reiter, FZI
Barbara Schütt, FZI
Patrick Schulz, FZI
Stefan Schwab, FZI
Maximilian Zipfl, FZI

Kontakt

Stefan Otten
FZI Forschungszentrum Informatik
Haid-und-Neu-Str. 10-14
76131 Karlsruhe
Tel.: +49 (0) 721 9654-156
Email: otten@fzi.de

Inhaltsverzeichnis

1 Übersicht	4
1.1 Aufgabenstellung	4
1.2 Voraussetzungen für das Vorhaben	8
1.3 Planung und Ablauf	9
1.3.1 Planung	9
1.3.2 Projektphasen	11
1.3.3 Ablauf	12
1.4 Stand der Wissenschaft und Technik	13
1.5 Zusammenarbeit mit anderen Stellen	16
2 Eingehende Darstellung	18
2.1 Erzielte Ergebnisse	18
2.1.1 TP3: Sicherheitsanalyse und funktionale Architektur	18
2.1.2 TP4: Regelwerk System- und Testanforderungen	18
2.1.3 TP6: Datenbereitstellung	18
2.1.4 TP7: Testvorgehen	28
2.1.5 AP7.1: Stand der Technik und Anforderungsanalyse	28
2.1.6 AP7.2: Entwicklung Testprozesse und -methoden	36
2.1.7 AP7.3: Exemplarische Referenzumsetzung	66
2.2 Notwendigkeit der Förderung	77
2.3 Bekannte Fortschritte auf dem Gebiet des Vorhabens bei anderen Stellen	77
2.4 Erfolgte Veröffentlichungen	77
3 Abbildungsverzeichnis	80
4 Tabellenverzeichnis	83
5 Literaturverzeichnis	84

1 Übersicht

1.1 Aufgabenstellung

Das Gesamtziel des Projektes war die Entwicklung von Systematik und Methoden für den Sicherheitsnachweis von vollautomatisierten und fahrerlosen Fahrfunktionen und Fahrzeugen (Level 4/5 nach VDA-Definition) zur Homologation im urbanen Umfeld.

Klassische Absicherungsmethoden beruhen vielfach auf der Annahme, dass neue Funktionen hinreichend sicher sind, wenn sie unter Realbedingungen über hunderttausende Kilometer ohne Probleme funktioniert haben. Versucht man, diesen Ansatz auf L4/L5 Systeme zu übertragen, stößt man schnell auf Probleme. Die Extrapolation herkömmlicher Absicherungsmethoden erfordern für ein vollautomatisiertes System, dass alle möglichen Kombinationen von konkreten Verkehrssituationen, Umfeldfaktoren und Systemzuständen einschließlich Systemfehlern vollständig zu betrachten sind. Der dadurch entstehende Kombinationsraum ist so groß, dass eine statistisch hinreichend vollständige Untersuchung mit aktuell etablierten Testmethoden nicht wirtschaftlich darstellbar ist. Zudem sind die für das System relevanten und häufig auftretenden Situationen nicht vollumfänglich bekannt (Vollständigkeitsdilemma).

Die dem Projekt zugrunde liegende Annahme war, dass diese technisch/wissenschaftlichen Aufgabenstellungen gelöst werden können, indem zunächst der für den Sicherheitsnachweis ausreichende Kombinationsraum (Topologie, Verkehrssituation, Wetter,...) über eine strukturierte Analyse durch eine handhabbare Anzahl von Szenarien darstellbar wird. Diese Szenarien müssen dabei

1. alle statistisch relevanten kritischen Ereignisse umfassen,
2. so beschrieben sein, dass sie für menschliche Experten verständlich sind und
3. die Vielzahl der tatsächlich vorkommenden und denkbaren Szenarien in einer handhabbaren Anzahl von parametrisierten Szenarien zusammenfassen.

Würden alle notwendigen Prüfungen eines L4/L5-Fahrzeuges nur auf Basis von Gesamtfahrzeugtests durchgeführt entstünde ein gigantischer physischer Prüfaufwand und entsprechender Zeitaufwand. Stattdessen muss Aufwand und Zeitdauer in einem durchführbaren Rahmen gehalten werden, der es ermöglicht, die Vielzahl der möglichen Szenarien mit hinreichender Variation effizient und kostengünstig sowohl für Gesamtfahrzeuge als auch Teile eines Fahrzeuges separat abzusichern.

Um die Durchführbarkeit zu erhalten, muss der Sicherheitsnachweis Tests und Testergebnisse unterschiedlicher Testinfrastrukturen und Integrationsstufen (Komponente bis System) optimal und nachvollziehbar zusammenführen. Zudem sind neuartige Testbeschreibungen sowie Metriken zur Bewertung der Systeme zu entwickeln. Für den Sicherheitsnachweis ist eine Kombination aus Simulations- und Realtests zu erarbeiten, durch die Testaufwände mit Hilfe von Simulationstechnologien zunehmend in die virtuelle Welt verschoben werden können.

Das Grundkonzept der im Projekt verfolgten Lösung war es, aufbauend auf etablierten Nachweismethoden, innovativen Forschungsergebnissen und existierenden Forschungsprojekten (u.a. PEGASUS, ENABLE-S3, SAFE) eine Methodik für die Freigabe / Homologation von L4/L5-Fahrfunktionen zu entwickeln.

Vor diesem Hintergrund und der Gesamtzielsetzung – der Entwicklung von Systematik und Methoden für den Sicherheitsnachweis von vollautomatisierten und fahrerlosen Fahrfunktionen und Fahrzeugen (Level 4/5 nach VDA-Definition) zur Homologation im urbanen Umfeld – wurden drei zentrale technisch/wissenschaftlichen Aufgabenstellungen abgeleitet, die im Projekt bearbeitet wurden:

Aufgabenstellung I: Beherrschung der Vielzahl möglicher Szenarien

Bisher fährt der Mensch selbst – typischerweise risikomindernd. Vom Grundsatz her hat der Mensch in allen Situationen die Kontrolle über die Fahrfunktionen inne. Für die Entwicklung von Sicherheitsnachweisen greift heute insbesondere die ISO 26262 mit der im Wesentlichen zu zeigen ist, dass Lenkung, Bremsen, Beschleunigung etc. verlässlich funktionieren, damit der Mensch in allen Situationen die sichere Kontrolle über das Fahrzeug hat.

In vollautomatisierten Fahrzeugen ist der Mensch nicht mehr als (rechtzeitige) Rückfallebene verfügbar. Der Verhaltensanspruch ist somit, dass die Automation selbst alle potentiellen Fahrsituationen und Einflussfaktoren sicher behandeln muss. Der Sicherheitsnachweise für vollautomatisierte und fahrerlose Fahrzeuge muss also nachweisen, dass die Automation in „so gut wie allen“ (d.h. statistisch relevanten) Fahrsituationen sicher und risikomindernd agiert.

Wie eingangs erläutert gibt es eine unüberschaubar große Menge von Szenarien (viele Millionen), deren Prüfung und Untersuchung mit herkömmlichen Mitteln einen unbeherrschbaren Aufwand fordert. Noch schwieriger ist es, diese Szenarien korrekt zu definieren und den Nachweis ihrer statistischen Relevanz / Vollständigkeit zu führen. Somit ergab sich die erste technisch/wissenschaftliche Aufgabenstellung für VVMethoden (Aufgabenstellung I):

Welches sind die sicherheitsrelevanten Verhaltensansprüche an Fahrzeuge im urbanen Verkehr und wie können diese so erfasst werden, dass sie umfassend und gleichzeitig effizient getestet werden können?

Ansätze zur Beantwortung von Aufgabenstellung I in VVMethoden

Ansatz	Ergebnisse
Entwicklung einer Methodik zur handhabbaren Reduktion des L4/L5-Testraums;	I.1: Darstellung eines repräsentativen Szenarienraumes und Entwicklung einer Ontologie für Klassifizierung, Einflussfaktoren und Entwicklung der Wirkzusammenhänge
Ermittlung und Beschreibung der Klassen mittels repräsentativer, generischer Szenarien;	I.2: Identifikation und tiefgehendes Verständnis der relevanten Wirkzusammenhänge in Verkehrsgefährdungen und systematische Zuordnung von Sicherheitsprinzipien und Sicherheitskonzepten.
Ermittlung und Beschreibung der Einflussfaktoren; Analyse der Wirkzusammenhänge; systematische Ermittlung kritischer Situationen und Gefährdungen;	I.3: Gesamtmethodik inklusive Abdeckungsmetrik für die Entwicklung von
Identifikation / Ableitung zugehöriger Sicherheitskonzepte zur Unfallvermeidung;	

Ableitung einer handhabbaren Menge standardisierbaren, hinreichend vollständigen repräsentativer, einhüllender Tests. Szenarienkatalogen

Aufgabenstellung II: Inkrementelles Testen von Subsystemen und Gesamtsystemen

Die Summe der Testanforderungen an L4/L5-Fahrzeuge wird trotz der skizzierten Reduktionen gegenüber den heutigen Tests signifikant ansteigen. Gleichzeitig steigt die Komplexität der Systeme an. Um wettbewerbsfähig zu bleiben, müssen Hersteller deswegen zukünftig in der Lage sein, erhobene Testergebnisse und Sicherheitsnachweise wiederzuverwenden und Komponenten bzw. Subsysteme unterschiedlicher Zulieferer zusammenzuführen – ohne Brüche in der Test- und Sicherheitsargumentation für das Gesamtsystem. Für die Freigabe des Gesamtfahrzeugs hat daher die Art der Absicherung von Komponenten und Subsystemen eine zentrale Bedeutung. Die zentrale Forschungsfrage zur technisch/wissenschaftlichen Aufgabenstellung II war also:

Wie können Testmittel und -ergebnisse wiederverwendet bzw. systematisch aggregiert werden, wie können Hersteller und Zulieferer getestete Systeme sowie Tests austauschen und wie kann die Testbarkeit der Sicherheit von Anfang an in das System designed werden?

Ansätze zur Beantwortung von Aufgabenstellung II in VVMMethoden

Ansatz	Ergebnisse
Entwicklung einer Absicherungsmethodik über alle Systemebenen, die hierarchisch auf die System- und Testanforderungen für verschiedene (Teil-) Komponenten heruntergebrochen werden kann.	II.1: Methoden zur Ableitung von Sicherheitsanforderungen auf System/Subsystemen/Komponenten-Ebene aus den Sicherheitsanforderungen auf Fahrzeugebene auf Basis einer Sicherheitsargumentation. II.2: Strukturierte Anforderungsdefinition an Systeme und Tests; Definition eines Testprozesses und Testvorgehens für die Systemelemente und Herunterbrechen auf Absicherung von Level4/5-Systemen; Traceability, Sub-Systeme auf Basis einer durchgehenden Nachvollziehbarkeit und Vergleichbarkeit von Sicherheitsargumentation. Testergebnissen über den gesamten Testprozess hinweg; Vorbereitung für die Standardisierung von Freigabetests für Level4/5-Systeme. II.3: Metriken und Bewertungskriterien zur Freigabe von Level4/5-Systemen.
Ableitung von Testanforderungen inklusive Metriken	

Aufgabenstellung III: Nahtloses Testen über alle Testinstanzen

Der reine Realfahrzeugversuch, wie er bei teilautomatisierten Fahrfunktionen zur Absicherung Anwendung findet, ist in seinem Umfang für diesen großen Testraum entweder nicht ausreichend oder in seinem Aufwand nicht leistbar. Die Simulation ermöglicht hier ergänzend einen skalierbaren Ansatz, um eine sehr große Anzahl von Szenarien in endlicher Zeit abtesten zu können. Zudem

können z.B. konkrete Fahrsituationen bezüglich Einflussfaktoren simulativ variiert werden oder neue Kombinationen von Einflussfaktoren erzeugt werden, die so nicht in Realfahrten ermittelt wurden. Des Weiteren kann die simulative Erzeugung von Variationen real eingefahrener Fahrsituationen gezielt dafür genutzt werden, gefährliche Situationen zu erzeugen, die man mit einem Fahrzeug aus Sicherheitsgründen nicht einfahren kann.

Neben der Bewältigung eines enormen Testvolumens ist eine zentrale Herausforderung der Nachweis der fehlerfreien Funktion innerhalb dieser großen Anzahl von Tests, aber auch der Nachweis, dass die so gewonnenen Testergebnisse die Realität hinreichend gut abbilden. Hierbei ist das Zusammenspiel der Testinstanzen entscheidend. Die Nutzung der Simulation zur Prüfung hat gegenüber Realtests nicht nur den Vorteil, dass sie deutlich aufwandsärmer und zeitsparender ist, sie ist auch bezüglich der Variationsmöglichkeiten deutlich flexibler. Die zentrale Forschungsfrage von Aufgabenstellung III war:

Wie kann eine nahtlose Argumentationskette für den Sicherheitsnachweis aufgebaut werden in der sich Simulation und Realtest so ergänzen, dass Aufwände für Realtests minimiert und der Nutzen der Simulation für die Absicherung optimiert werden?

Ansätze zur Beantwortung von Aufgabenstellung III in VVMMethoden

Ansatz	Ergebnisse
Erzeugung einer durchgängigen, effizienten Prüfkette von Simulation bis Realfahrten, die die jeweiligen Stärken der einzelnen Testinstanzen nutzt und deren Ergebnisse effizient zu einer Gesamtsicherheitsaussage zusammengefasst werden können.	III.1: Sicherheitsnachweis für L4/L5-Fahrzeuge durch optimale und effiziente Verzahnung von Simulationstests und Realtests.
Effiziente Verifikation und Validierung durch Skalierbarkeit verschiedener Testinstanzen wie z.B. Wiederverwendbarkeit von Testeingangsdaten, Vergleich der Testergebnisse der jeweiligen Testinstanzen untereinander.	III.2: Methoden zum durchgängigen Testen über alle Testinstanzen inkl. aufeinander abgestimmte Testbeschreibungen sowie zugehörige Bewertungsmetriken.
Effizientes Zusammenfügen zu einer umfassenden Sicherheitsargumentation über einheitliche Schnittstellen, Formate und Bewertungsmetriken, zugehöriger Methodenentwicklung	III.3: Konzept und Demonstration eines durchgängigen Sicherheitsnachweises für ein L4/L5-Fahrzeug
Aufbau einer Absicherungsplattform zur Demonstration der Umsetzbarkeit und Nachweis der Methode	

1.2 Voraussetzungen für das Vorhaben

Die Voraussetzungen für eine effektive, effiziente und erfolgreiche Umsetzung der Projektziele des Vorhabens VVMethoden waren hervorragend, da das Gros der Konsortialpartner auf umfassende Erfahrungen in der Zusammenarbeit in vorwettbewerblichen Forschungs- und Entwicklungsprojekten zurückgreifen konnte, viele Konsortialpartner in weiteren Forschungs- und Entwicklungsprojekten aktiv waren, innerhalb derer weitere relevante Aspekte der Verifikation und Validierung hochautomatisierter und autonomer Fahrzeuge untersucht wurden und derart der Transfer aktueller Ergebnisse - zur Projektlaufzeit - in das Vorhaben VVMethoden sichergestellt war und das Konsortium auf eine Vielzahl von – insbesondere auf Seiten der wissenschaftlichen Partner – verfügbaren digitalen Infrastrukturen und Datenquellen zurückgreifen konnte, auf diese aufgesetzt hat und im Rahmen des Vorhabens VVMethoden weiterausgebaut und qualifiziert hat.

Das Projekt VVMethoden konnte auf eine breite Beteiligung aller relevanten Partner aus Industrie und Wissenschaft bauen, die ihre Kompetenzen in Forschung und Umsetzung eingebracht haben. Damit wurden zum einen die Voraussetzungen für eine qualitativ hochwertige, termin- und budgetgerechte Umsetzung der Projektziele geschaffen. Zum anderen war dies der Garant dafür, dass die im Projekt VVMethoden erarbeiteten Lösungen relevant für die Industrie sind und langfristig eingesetzt werden.

Auf Seiten der Industrie beteiligten sich Unternehmen, die die Perspektive der OEM (Audi, BMW, Daimler, FORD, Opel, VW) und der Tier1 (BOSCH, Continental, Valeo, ZF AG) einnehmen. Darüber hinaus waren relevante Technologieprovider (AVL, dSPACE, PROSTEP, understandAI) mit eingebunden. Mit dem TÜV SÜD wurde ein Projektpartner in das Konsortium integriert, der über zentrales Know-how in der Homologation und in der regulativen Freigabe von Systemen im Fahrzeug verfügt.

Darüber hinaus waren die relevanten Wissenschaftspartner, die über zentrale methodische Kompetenzen für die Verifikation und Validierung automatisierter Fahrzeuge verfügen, in das Projekt eingebunden. Mit den Partnern DLR, ika, FZD, FZI, IfR, FhG und OFFIS, die während der Projektlaufzeit ein Institut des DLR wurden, konnten namhafte Wissenschaftsakteure aus dem universitären und außeruniversitären Bereich für die Arbeit am Projekt VVMethoden gewonnen werden.

Die Wissenschaftspartner haben einen entscheidenden Teil dazu beigetragen, dass die nötige, disruptive Effizienzsteigerung der Prüf- und Instanziierungsprozesse erreicht wurde. Ihnen kamen zentrale Aufgaben im Bereich der Entwicklung der Kritikalitätsanalysen, der Entwicklung von Sicherheitskonzepten sowie der Entwicklung von Testinfrastrukturen und Datendanken (nicht wettbewerbsdifferenzierend) zu. Damit konnte gewährleistet werden, dass der vorwettbewerbliche Ansatz von VVMethoden diskriminierungsfrei umgesetzt wurde und dass die Ergebnisse bei allen Industriepartnern zeitnah und effizient Eingang in die Wertschöpfungskette gefunden haben.

Nur im Zusammenspiel aller beteiligten Partner im Wertschöpfungsprozess konnte sichergestellt werden, dass die zu erarbeitenden Lösungen für den Einsatz in der Industrie geeignet sind. Die große Zahl an Varianten von Fahrzeugen mit unterschiedlichen Ausstattungen an Sensoren und Komponenten sowie die hohe Geschwindigkeit mit der neue Komponenten entwickelt werden, bedingt neue Konzepte für Verifikation und Validierung. Bereits für das Zusammenspiel zwischen Zulieferern verschiedener Ebenen und Fahrzeugherstellern während der Entwicklung neuer Systeme mussten daher gemeinsame Methoden und akzeptierte Systematiken gefunden werden.

Das starke Zusammenwirken der Partner wurde auch in der Projektorganisation deutlich: Industrie und Wissenschaftspartnern haben die Verantwortung und Führung des Projektes gemeinsam („Doppelspitze“) wahrgenommen.

Teilprojekt	TP-Leitung Industrie	TP-Leitung Wissenschaft
TP1: Rahmenbedingungen und Impulsgeber	Opel	IfR
TP2: Kritikalitätsanalyse	Mercedes-Benz	DLR (ehemals OFFIS)
TP3: Sicherheitsanalyse und funktionale Architektur	BOSCH	IfR
TP4: Regelwerk System- und Testanforderungen	Mercedes-Benz	FhG
TP5: Simulationssteuerung	TÜV Süd	DLR
TP6: Datenbereitstellung	AVL	FZD
TP7: Exemplarische Testumgebung und -Anwendung	ZF AG	FZI
TP8: Datenbanken	AVL	ika
TP9: Prozess Instanziierung, kontinuierlicher Verbesserungsprozess, Anwendung und Bewertung	PROSTEP	-

1.3 Planung und Ablauf

1.3.1 Planung

VVMethoden hat als vorwettbewerbliches Verbundprojekt die nötigen Stakeholder und Kompetenzträger aus Industrie und Wissenschaft zusammengebracht, um die Projektziele effizient zu verfolgen und die entstandenen Ergebnisse nah an den Bedürfnissen der Industrie bereitzustellen. VVM ist ein Projekt der VDA Leitinitiative autonomes und vernetztes Fahren und als solches Mitglied der Pegasus Projektfamilie zu der auch die Projekte Pegasus und Set Level gehören. Im Vorläuferprojekt Pegasus wurden wichtige Ansätze und Ergebnisse erarbeitet auf denen VVM aufbaut, Set Level als zeitlich überlappendes Projekt war inhaltlich direkt mit VVMethoden verbunden, dort wurden Simulationen und Simulationsmodelle entwickelt, deren Anforderungen in VVMethoden erarbeitet wurden.

1.3.1.1 Teilprojekte

Die relativ große Zahl von neun Teilprojekten spiegelt die vielfältigen Arbeitsgebiete wider, die zur Bewältigung der komplexen Aufgaben beigetragen haben. Die Teilprojekte 1 "Rahmenbedingungen

und Impulsgeber" und 9 "Prozess Instanziierung, kontinuierlicher Verbesserungsprozess, Anwendung" hatten dabei projektübergreifende Sonderfunktionen und waren mit allen anderen TPs in kontinuierlichem Austausch.

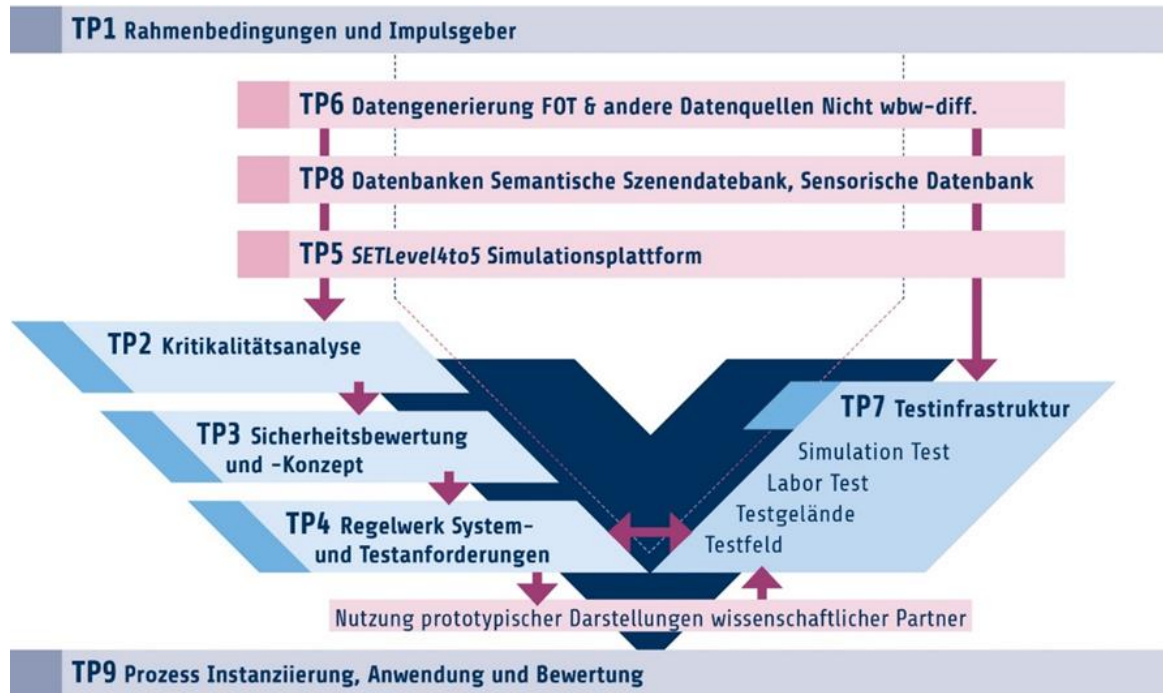


Abbildung 1: Die TPs aus VVM und deren Zusammenhang

TP1 Rahmenbedingungen und Impulsgeber fungierte als Befähigungs-TP, das zu Beginn umfassende Rahmenbedingungen sicherstellt und im weiteren Verlauf darauf achtet, dass eine gesamtheitliche Projektperspektive beibehalten wird. Die unterschiedlichen Erkenntnisstränge wurden zusammengefügt und -gehalten um ein kohärentes und nutzbares Gesamt-Projektergebnis zu erreichen. Dieser starke inhaltliche Aspekt wurde durch die in einem Projekt dieser Größe nötigen, vielfältigen administrativen Aufgaben ergänzt. Eine weitere wichtige Aufgabe von TP1 bestand in der Ergebnisverbreitung um zu gewährleisten, dass die VVM-Ergebnisse in der internationalen Fachcommunity kommuniziert wurden, diskutiert werden und letztlich Eingang in die nötigen Gremien finden.

In TP2 Kritikalitätsanalyse wurden, aufbauend auf den Anforderungen, Anwendungsfällen und Metriken aus TP1 die Methoden der Kritikalitätsanalyse entwickelt, die in die Sicherheitsanalysen von TP3 und in die Erstellung der Gesamtmethodik von VVMethoden (TP1) einfließen. Weiterhin lieferte TP2 Anforderungen an Simulationsplattformen (TP5) und an die Datenerhebung aus Realfahrten (TP6) aus Sicht der Kritikalitätsanalyse.

In TP3 Sicherheitsanalyse und funktionale Architektur wurden aus den Kritikalitätsanalysen von TP2 Sicherheitsanalysen entwickelt aus denen wiederum Sicherheitskonzepte für TP4 abgeleitet wurden. Das beinhaltete globale und lokale Betrachtungen sowie deren Interaktion.

TP4 Regelwerk System- und Testanforderungen hat, basierend auf den in TP 3 durchgeführten Sicherheitsanalysen, den daraus entwickelten Sicherheitskonzepten und der funktionalen Systemarchitektur dann System- und Testanforderungen für die Homologation abgeleitet. Ziel war es, Methodiken zur Prüfung des Gesamtfahrzeuges über die Prüfung von Ansprüchen an

Systeme/Sub-Systeme/Komponenten zu entwickeln, so dass Hersteller und Zulieferer getestete Systeme und Tests austauschen sowie Tests wiederverwenden können. Ebenso sollte die Methodik erlauben, die Testbarkeit der Sicherheit von Anfang an im Systemdesign zu berücksichtigen. TP4 widmete sich damit vornehmlich den Grundkonzepten II und III für die System- und Testanforderungen, die basierend auf einer Sicherheitsargumentation hierarchisch herunter gebrochen wurden.

TP5 Simulationssteuerung hat im Rahmen des Projektes VVMethoden die Simulationstechnologie und die Methoden zu ihrer Anwendung bereitgestellt. Dabei wurden zwei Hauptanwendungszwecke verfolgt, die Kritikalitätsanalyse im TP2 und die Absicherung von automatisierten Fahrzeugen und deren Komponenten (Sensorik, Fahrfunktionen etc.) im TP7. In der Kritikalitätsanalyse wurde sie zur Ableitung und Analyse relevanter Szenarien genutzt, an denen die Kritikalitätsphänomene studiert, präzisiert und bewertet werden können. Bei der Absicherung diente sie dem umfassenden Testen von Systemen oder Komponenten von Systemen.

Im TP6 Datenbereitstellung wurden Messdaten generiert, aufbereitet und im Projekt zur Verfügung gestellt. Die Daten wurden dabei durch wissenschaftliche Partner mit bestehenden Fahrzeugen sowie durch AVL mit einem für das Projekt aufgebauten Versuchsträger erzeugt. Zusätzlich wurden in der Anwendungsplattform Intelligente Mobilität (AIM) Daten einer Braunschweiger Kreuzung erhoben. Weitere Daten lieferte das Karlsruher Testfeld automatisiertes Fahren (TAF) sowie die mobile Messtechnik des FhG-IVI.

In TP7 Testvorgehen wurden die Testanforderungen aus TP4, die Simulationswerkzeuge aus TP5, die in TP9 ermittelten wettbewerbsdifferenzierenden Anforderungen sowie Anforderungen und Vorarbeiten aus Forschungsprojekten wie bspw. PEGASUS verwendet, um daraus ein Testvorgehen für die Absicherung von L4/L5-Systemen zu entwickeln. Dies hat insbesondere das identifizierte technisch/wissenschaftliche Arbeitsziel III „Nahtloses Testen über alle Testinstanzen“ adressiert.

Kernziel von TP8 Datenbanken war die Verarbeitung der im Projekt benötigten und erzeugten Daten in einer zentralen Datenbank. TP8 hat somit die technische Basis für die Erzeugung einer durchgängigen, effizienten Prüfkette von Simulation bis Realität zur Verfügung gestellt.

TP9 Prozess Instanziierung, kontinuierlicher Verbesserungsprozess, Anwendung hatte zu Aufgabe der Anforderungen der Industriepartner zu sammeln und zu bündeln, insbesondere die der Projektpartner, die nicht an den entsprechenden TPs 2-8 beteiligt waren ("Brückenfunktion"). Außerdem wurden dort die Erkenntnisse aus der praktischen Erprobung der im VVMethoden-Projekt entwickelten Methodik durch die Industriepartner in das VVMethoden-Projekt konsolidiert und zurückgemeldet, („Erkenntnismotor aus industrieller Anwendung“).

1.3.2 Projektphasen

Die Arbeiten in VVMethoden wurden in drei Phasen konzipiert, wie in Abbildung 2Abbildung 1 dargestellt:

- **Phase 1 – Ansatzphase** – Aufstellen der zentralen Ansätze auf Basis der Stand der Technik und im Wesentlichen auf Basis bestehender Daten.
- **Phase 2 – Vertiefungsphase** – Ausarbeitung und Konkretisierung der Ansätze auf Basis nicht automatisierter Fahrten und Einbeziehung der Simulation.

- **Phase 3 – Komplettierungsphase** – industriegerechte, effiziente Verknüpfung der Ansätze, Quercheck. Finale Einbeziehung von Automationsrisiken über Datenanreicherung auf Basis von Fahrten mit SAE-Level 4/5 Fahrfunktionen.

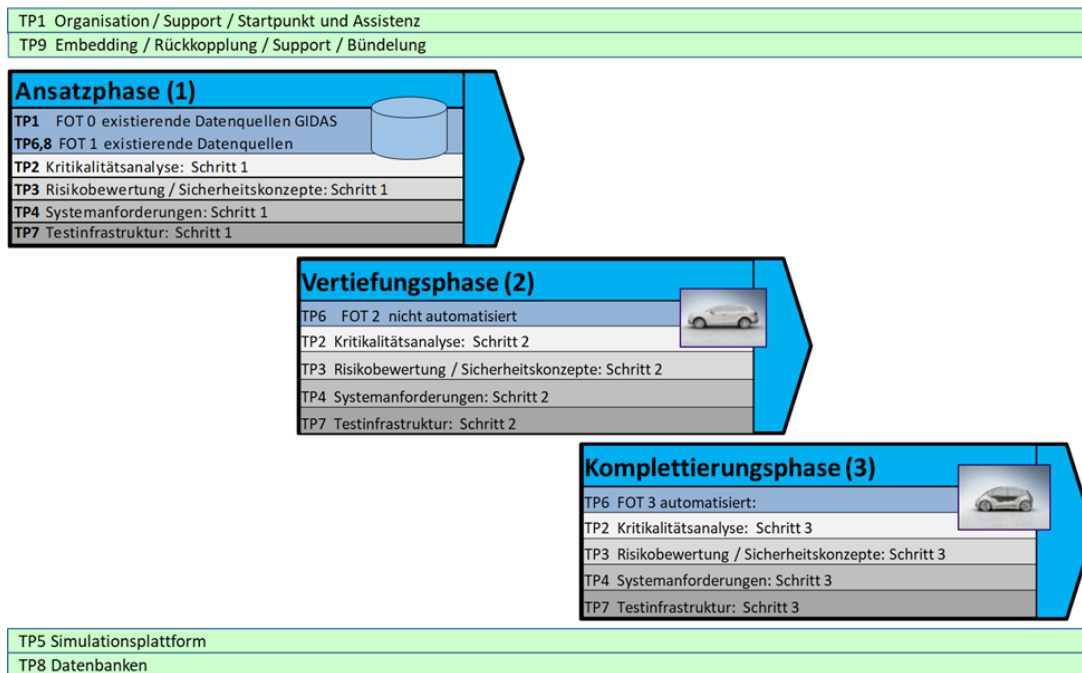


Abbildung 2: Die drei Arbeitsphasen in VVM-Methoden.

Durch diesen iterativen Ansatz wurde zum einen gewährleistet, dass neue Ansätze und Erkenntnisse die außerhalb des Projektes entstanden sind auch während der Projektlaufzeit aufgenommen werden konnten. Zum anderen konnten so schon frühe Versionen der Ergebnisse ausgeleitet werden, um sie bei den beteiligten Herstellern und Zulieferern zu erproben oder in externen Kreisen diskutiert und reflektiert werden.

1.3.3 Ablauf

Die Projektlaufzeit wurde initial von 07/2019 bis 06/2023 geplant. Damit fiel der Beginn der Covid-19-Pandemie in das dritte Projektquartal. Diese hatte für VVM verschiedene Auswirkungen: Die kurzfristige Umstellung auf vollständiges Homeoffice hat zu Ausfällen, Nichtverfügbarkeiten und reduzierter Effizienz geführt, welche die Projektarbeit merklich verzögert haben.

Die physischen Treffen wurden eingestellt und waren erst sehr spät wieder in der geplanten Breite möglich. Dies betraf nicht nur die klassischen Arbeitsmeetings und Konsortialtreffen, die einen großen Teil der Projektlaufzeit als Videokonferenzen abgehalten werden mussten, oder die Halbzeitpräsentation bei der nur ein Redner-Rumpfteam zusammenkam während alle Zuschauer remote teilnehmen mussten.

Zur tieferen Reflektion der Absätze und um früh und mehrfach die Projektergebnisse mit der Fachcommunity zu diskutieren waren am Ende jeder Projektphase "Offene Treffen zur Ergebnisverbreitung" vorgesehen, die in dieser Form nicht stattfinden konnten.

Außerdem war in TP9 verankert die Industriepartner aufzusuchen und deren Feedback zu (1) Anforderungsmanagement, (2) Harmonisierung Werkzeugschnittstellen und Dateiformate und (3)

Instanziierung der Verifikations- und Validierungsmethodik einzuholen, um es aufzubereiten und in das Projekt zurückzuspielen. Auch das musste Remote durchgeführt werden.

Um die ambitionierten Projektziele auch unter diesen Umständen zu erreichen wurde eine kostenneutrale Projektverlängerung von sechs Monaten beantragt, die vom Projektträger bewilligt wurde.

1.4 Stand der Wissenschaft und Technik

Das Projekt VVMethoden knüpfte an den aktuellen Stand der Wissenschaft und Technik betreffend Methoden und Werkzeuge des Testens automatisierter Fahrzeugfunktionen an. Insbesondere lieferten die Ergebnisse aus den Projekten PEGASUS¹ sowie ENABLE S3² dafür eine gute Ausgangsbasis.

Die industrieseitige Freigabe von autonomen Fahrfunktionen SAE-Level 4/5 konfrontiert die Automobilindustrie mit der Herausforderung, dass etablierte Verifikations- und Validierungsmethoden über Realfahrten durch die immer höheren Innovationsgeschwindigkeiten, die mit der Digitalisierung einher gehen, nicht mehr einsetzbar und nicht mehr effizient sind.

Gleichzeitig liefern reine Computersimulationen nicht die notwendige Sicherheit für die Verifikation und Validierung. Das Grundkonzept der im Projekt verfolgten Lösung ist es, eine Methodik für eine Kombination von Simulationstests bis Realtests zu entwickeln, die der Komplexität des erforderlichen Nachweises gerecht wird, und es erlaubt, einen Großteil der Tests simulativ oder mit Simulationsunterstützung durchzuführen.

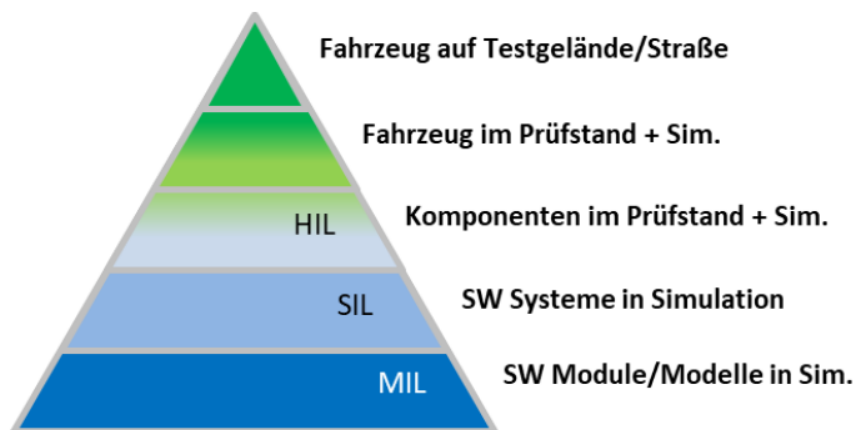


Abbildung 3: Testpyramide für die Absicherung

Für diese Kombination wird eine Hierarchie von ineinandergreifenden Testumgebungen konzipiert, mit denen das Verhalten von L4/L5-Fahrzeuge und ihrer Systeme und Komponenten durch eine Kombination verschiedener Testinstanzen valide beurteilt werden kann. Dafür werden geeignete Methoden entwickelt, passende Werkzeuge zusammengestellt und deren Anwendung durch Prozesse beschrieben. Die Wirksamkeit des Verfahrens und seiner Übertragbarkeit in praktische Entwicklungsprozesse wird an Hand eines gemeinsamen Use Cases Urbane Kreuzung untersucht. Die Urbane Kreuzung ist mit ihrem Mischverkehr aus nicht-automatisierten und automatisierten Fahrzeugen, Radfahrern, Fußgängern, schienenengebundenem und schienenlosem OPNV, etc. einer

¹ <https://www.pegasusprojekt.de/de/>

² <https://enable-s3.eu/>

der komplexesten Use Cases für urbanes L4/L5-Fahren, der den Großteil der Herausforderungen, potentiellen Gefahren und notwendigen L4/L5-Sicherheitskonzepte abdeckt.

In der Vergangenheit war es bislang praktisch ausschließlich der Fahrer, der die Entstehung kritischer Verkehrssituationen zu vermeiden hatte, während die im Fahrzeug realisierten Fahrfunktionen ihn dabei unterstützen mussten und selbst keine Gefahrenentstehung begünstigen durften. In der Außensicht bildeten Fahrer und Fahrzeug damit eine „Containment-Region“, in welcher kontinuierlich die Risiken außerhalb des Fahrzeugs minimiert werden. Um hier ein aus Sicherheitssicht mindestens gleichwertiges Verhalten bei der Umsetzung von L4/L5 Fahrfunktionen zu erlangen, ist die Erzeugung eines tiefgreifenden Verständnisses zur Gefahrenentstehung in einer komplexen urbanen Verkehrsumwelt erforderlich, welches bis heute nicht existiert. Eine solche Durchdringung ist aber für die Absicherung von Perzeptions- und Entscheidungsfunktionen in vollautomatisierten und autonomen Fahrzeugen unabdingbar, da andernfalls nicht einmal die Verifikations- und Testziele für die Absicherungsschritte bestimmt werden können.

Als grundlegendes Vorgehensmodell für die Produktentwicklung in der Automobilindustrie ist das in dargestellte V-Modell etabliert [1]. Das V-Modell beschreibt ein strukturiertes und sequentielles Vorgehen in der Entwicklung mit definierten Entwicklungsphasen. Der linke absteigende Ast beschreibt Top-Down-orientierte Dekompositionsphasen in der Systementwicklung zur Spezifikation und Realisierung des Systems. Entsprechende Aktivitäten im Bereich Verifikation und Validierung sind zugehörig gegenüber als Bottom-Up-orientierte Integrations- und Prüfphasen eingeordnet. Die Aktivitäten im rechten Schenkel des V-Modells prüfen entsprechend ihrer Stufe die Umsetzung der im linken Schenkel erarbeiteten Spezifikationen und Implementierungen [2]. Automotive SPICE [3] ist als Prozessreferenzmodell an das Vorgehen im V-Modell angelehnt und dient als Rahmenwerk für produktbezogene Entwicklungs- und Managementprozesse im Bereich der Elektrik/Elektronik für die Automobilindustrie.

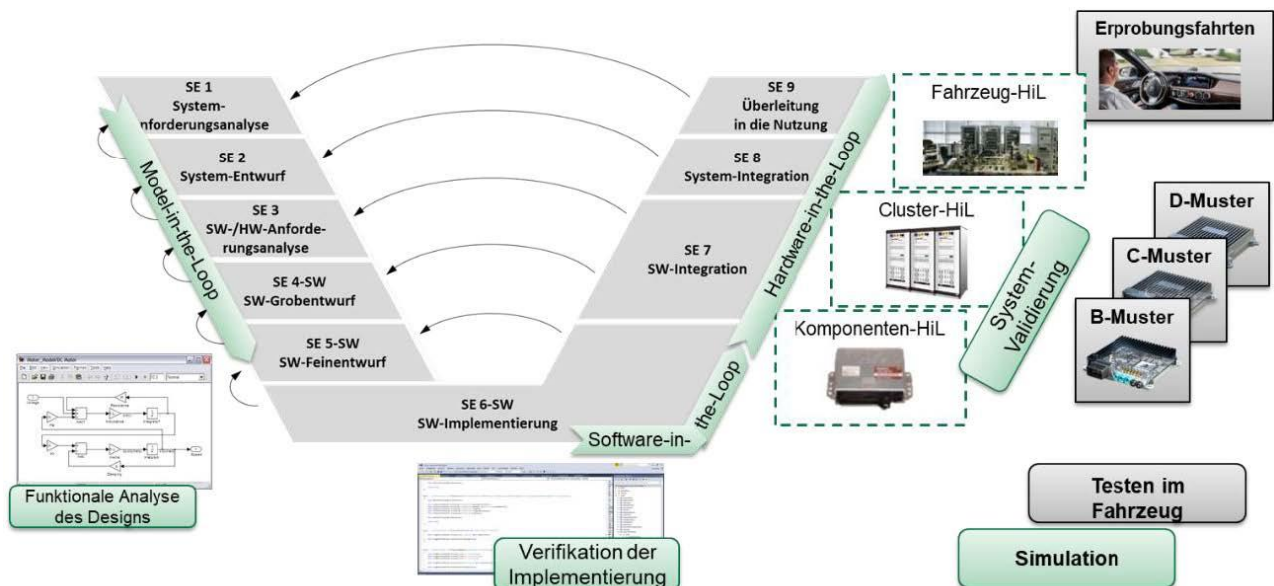


Abbildung 4: Verifikation und Validierung im Entwicklungsprozess

Der Test von interagierenden Regelungssystemen erfordert einen geschlossenen Wirkungsweg um die Interaktion des Systems mit seiner Umwelt zu ermöglichen. Dieser Ansatz einer simulativen Absicherung wird auch als X-in-the-Loop (XiL) Methode bezeichnet [4]. Der Begriff „in-the-Loop“

impliziert dabei einen bidirektionalen Informationsfluss zwischen Testobjekt und Simulationsumgebung. Für die simulationsgestützte Entwicklung von Closed-Loop Systemen existieren für alle Stufen entlang des Entwicklungsprozesses spezialisierte Methoden.

Die drei gängigen Abstraktionsebenen sind [5]:

- Model-in-the-Loop (MiL): Einsatz einer Simulationsumgebung zur Generierung der Stimuli für die Entwicklung und den Test spezifizierter Funktionsmodelle
- Software-in-the-Loop (SiL): Einsatz einer Simulationsumgebung für die Ausführung und den Test von Seriencode auf Ebene der Software Systemarchitektur
- Hardware-in-the-Loop (HiL): Einsatz einer Simulationsumgebung für die Ausführung und den Test der in die Zielhardware integrierten Softwarekomponenten unter Verwendung der technischen Systemschnittstellen. Hier wird zudem zwischen Komponenten-, Cluster und Integrationstests unterschieden [6]

Um das gesamte Fahrzeug auf Systemebene zu testen und abzusichern, werden in großem Umfang reale Versuchsfahrten angewendet. Diese können im ganzen Entwicklungsprozess im Rahmen von ersten Prototypen (Rapid Prototyping), Erprobungsträgern und Dauerläufern durchgeführt werden [7]. Zum Ende der Entwicklung hin sind diese insbesondere für eine Validierung der Systeme im Realbetrieb relevant. Für den Abnahmetest und die Zulassung des Fahrzeugs wird heute eine hohe Zahl an Testkilometern in Realfahrten benötigt. Diese werden beispielsweise bei Winter- und Sommererprobungen sowie bei länderspezifischen Erprobungen gesammelt.

Die durchzuführenden Testfälle in den unterschiedlichen Stufen werden im Rahmen von Testspezifikationen definiert und festgelegt. Hierbei sind die Tests systematisch aufgebaut und folgen einem stringenten Vorgehen. Basierend auf definierten Vorbedingungen werden einzelne Testschritte bspw. als Events festgelegt. Am Ende des Testschrittes werden die beschriebenen erwarteten Ergebnisse mit den Ergebnissen des Tests verglichen. Eine Aussage, ob der Test bestanden wurde erfolgt über entsprechende Testauswertungen, bspw. binäre Entscheidungen (PASS/FAIL). Die Definition der einzelnen Testschritte sowie die Überprüfung des Testfalls erfolgen zumeist über dezidierte Signale bzw. Signalverläufe, die Spezifikation der Testfälle anhand von Anforderungen an das System. Somit sind die Tests zumeist synthetisch und basieren stark auf der Expertise des Testingenieurs [8].

Die Absicherung und Homologation von autonomen Fahrzeugen kann nicht ausschließlich auf Basis von Realfahrten ermöglicht werden, da die notwendigen Kilometer [9] [10] von den benötigten Aufwänden nicht leistbar ist. Der Ansatz im Projekt VVMethoden ist, Testverfahren zu entwickeln bzw. zu optimieren, dass die Zulassung von automatisierten Fahrfunktionen mit möglichst wenigen aufwändigen Realtests möglich ist. Im Rahmen des Projektes werden Vorgehen, Methoden und Werkzeuge entwickelt, welche ein sinnvolles Zusammenspiel sowie Verzahnung von simulativer Absicherung und Realtests ermöglichen. Hierbei muss die Komplexität des erforderlichen Nachweises berücksichtigt werden. Es soll eine Hierarchie von ineinandergreifenden Teststufen und Testinfrastrukturen konzipiert werden, mit denen das Verhalten eines autonomen Fahrzeugs und seiner Systeme und Komponenten valide beurteilt werden kann. Simulationsmodelle müssen konsistent und systematisch für die unterschiedlichen Teststufen erarbeitet werden, um einen Austausch zu gewährleisten. Abhängig von den jeweiligen Testzielen bzw. Teststufen werden

unterschiedliche Granularitäten von Modellen für unterschiedliche Testziele und Systemreifegrade verwendet.

Aktuelle Fahrzeugtests ermöglichen durch die mitgeloggten Informationen eine weitergehende Analyse von kritischen Situationen am Schreibtisch. Es existieren jedoch keine durchgängigen und kontinuierlichen Ansätze, Erkenntnisse aus dem Fahrzeugtest für die Simulation zur Verfügung zu stellen bzw. Teile der Simulation aus den Realdaten zu verwenden [11]. Dadurch sind kritische Situationen im realen Fahrzeugtest nicht reproduzierbar. Für eine sinnvolle Wiederverwendung von aufgezeichneten Daten sind einheitliche Schnittstellen und Formate erforderlich. Dies geht einher mit einer gesamtheitlichen Teststrategie über Teststufen und Systemreifegrade hinweg.

Für die Evaluation des Funktionsverhaltens im realen Fahrzeugtest wird die subjektive Bewertung des Testfahrers herbeigezogen. Für eine Objektivierung sowie die Vergleichbarkeit mit Tests in der virtuellen Welt und damit übergreifende und durchgängige Bewertung zu ermöglichen, sind einheitliche funktionale Metriken und Bewertungskriterien notwendig. Diese sind über alle Teststufen und Testmittel hinweg anzuwenden und erlauben eine durchgängige Bewertung des Systemreifegrads.

Für Level4/5-Funktionen ist insbesondere die Reaktivität des Systems mit den umliegenden Verkehrsteilnehmern relevant. In diesem geschlossenen Regelkreis ist es für einen Test wichtig, dass je nach Funktionsreifegrad und Testziel auch die passenden Umgebungsmodelle in Bezug auf Abstraktionslevel und Güte bereitgestellt werden. Für eine solche sinnvolle Orchestrierung und optimale Ausnutzung aller Testinfrastrukturen sowie Testmitteln ist bisher keine Methodik existent. Das Projekt VVMethoden liefert ein Konzept und eine Methodik, um eine durchgängige Absicherung über alle Teststufen zu gewährleisten. Insbesondere die enge Verzahnung von Simulation und Realtests für einen durchgängigen Nachweis der Systemqualität bis hin zur Freigabe ist hierbei zu betrachten. Basierend auf einer Aufstellung aktueller Testmittel und -infrastrukturen ist ein Vorschlag für eine gesamtheitliche Teststrategie bzw. einen neuartigen Testprozess zu erarbeiten.

1.5 Zusammenarbeit mit anderen Stellen

Im Rahmen des Projektes bestand für das FZI die Möglichkeit, mit zahlreichen hochqualifizierten und technologisch führenden Partnern des Automobilsektors aus Wirtschaft und Wissenschaft zusammenzuarbeiten, siehe Abbildung 5. Als TP-Lead und mit einem Schwerpunkt auf der Erarbeitung von neuartigen Testmethoden bestand für das FZI eine intensive Zusammenarbeit insbesondere mit den Partnern in TP7.

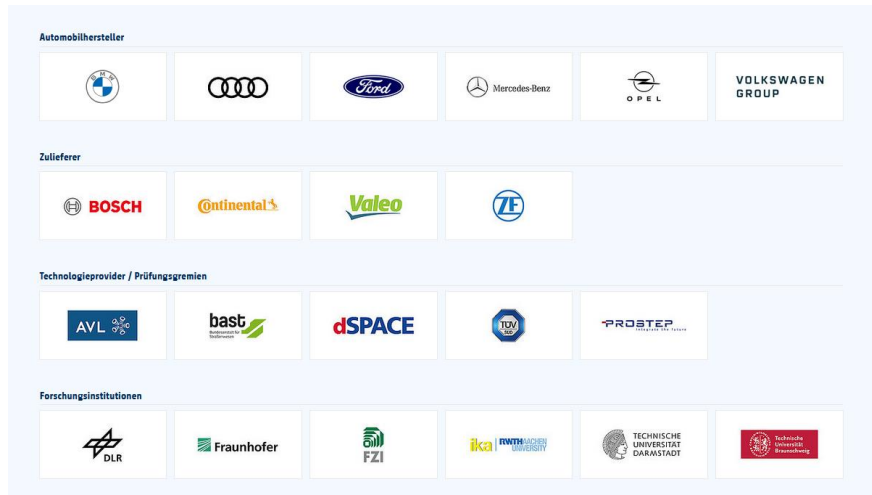


Abbildung 5: Zusammensetzung des VVM-Konsortiums

Das FZI hatte zudem spezifische Unteraufträge in der Verantwortung, welche eine Zusammenarbeit mit dem jeweiligen Unterauftragnehmer erforderten:

- Unterauftrag an das Testfeld Autonomes Fahren Baden-Württemberg (TAF BW): Bereitstellung von Kartendaten und Daten von Verkehrsteilnehmern aus Kreuzungen des Testfelds sowie Darstellung von Testszenarien im Testfeld mit dem Versuchsträger des FZI
- Unterauftrag an für die Aufrüstung und den Umbau des Versuchsträger an EDAG: Konzepterstellung für die Integration der Sensorik und Hardware in den Versuchsträger sowie anschließender Fahrzeugaufbau

Während der Projektlaufzeit erfolgte kontinuierlich der Austausch von Erkenntnissen und Ergebnissen mit Gremien und anderen Projekten, die sich mit dem Thema der Absicherung von automatisierten Fahrzeugfunktionen beschäftigen. Zudem wurde intensiv mit Kolleg*innen am FZI, die in verwandten Projekten forschen, im vertragsgemäßen Rahmen diskutiert und über fachliche Ideen beraten. Darüber hinaus gab es einen regen Austausch mit der breiteren wissenschaftlichen Community im Rahmen von Fachkonferenzen sowie nationalen Versammlungen, Projekttreffen, Tagungen und Workshops.

2 Eingehende Darstellung

2.1 Erzielte Ergebnisse

2.1.1 TP3: Sicherheitsanalyse und funktionale Architektur

Im Rahmen von TP3 war das FZI in Umfang an der Dekomposition und Modularisierung von Konzepten für den Sicherheitsnachweis beteiligt. Der Fokus der Arbeiten des FZI lag hierbei insbesondere darauf, die Durchgängigkeit und Harmonisierung des erarbeiteten Ansatzes zur Sicherheitsargumentation mit den in TP7 entwickelten Testmethodiken und -vorgehen sicherzustellen. Die Schritte und Schnittstellen von spezifischen Testaktivitäten wurden mit den erarbeiteten Hierarchieebenen und Prozessschritten im Assurance Framework von VVM harmonisiert bzw. integriert sowie in den Arbeitsgruppen diskutiert und Ergebnisse reviewed. Dies erfolgte vorrangig über Abstimmungen in TP-übergreifenden Runden, in diesem Fall die Arbeitsgruppe „Core04“ welche die Klammer über die zentralen Arbeitspakete 3,4 und 7 hergestellt hat.

2.1.2 TP4: Regelwerk System- und Testanforderungen

Im Rahmen regelmäßiger Diskussionsrunden der Arbeitsgruppe „Core04“ wurde die Schnittstelle zwischen den Arbeiten des TP4 und TP7 ausgearbeitet. Fokus liegt auf der Durchgängigkeit der Testinstanzen bzw. Ableitung von Anforderungen und durchgängigen Anwendungsbeispielen über TP3, TP4 und TP7. Für das FZI war hierbei insbesondere die Integration des HiL-Prüfstandes mit einem Schwingungsprüfstand relevant, da hieran die Abbildung und Durchgängigkeit von nicht-funktionalen Anforderungen in den Testinstanzen dargestellt werden konnte sowie der finale Aufbau des Versuchsträger zur Darstellung von durchgängigen Testinfrastrukturen. In gemeinsamen Arbeitsgruppen wurden die methodischen Ansätze zur Ableitung von System- und Testanforderungen abgestimmt und für die konkreten Beispiele und Anwendungsfälle des FZI bzw. aus TP7 harmonisiert.

2.1.3 TP6: Datenbereitstellung

Im Rahmen von TP6 lagen die Ziele, Schwerpunkte und Ergebnisse des FZIs auf den folgenden Aspekten:

- Datenlieferung von Kartendaten und Kreuzungsdaten von Verkehrsteilnehmern aus dem Testfeld Autonomes Fahren (TAF) Baden-Württemberg (erfolgreich realisiert über Unterauftrag)
- Datenaufbereitung von Kreuzungsdaten und Konzept zur Überführung in Szenarien (siehe Abschnitt 2.1.3.1)
- Erarbeitung von Methoden zur Szenariendefinition und -analyse mit semantischer Annotation (siehe Abschnitte 2.1.3.2 und 2.1.3.3)

2.1.3.1 Überführung von eingefahrenen Szenarien in OpenScenario

Die Straßengeometrie und -topologie der im Projekt verwendeten Verkehrsdatensätze, darunter inD [12], INTERACTION [13] und der vom Testfeld Autonomes Fahren (TAF) Baden-Württemberg bereitgestellte Datensatz [14], sowie die vom Forschungsfahrzeug erfassten Datensätze werden mit Hilfe von Lanelets beschrieben.

Aufgrund der häufigen Referenzierung des OpenScenario-Formats auf OpenDrive-Karten ist ein Konverter zwischen diesen beiden Formaten notwendig. OpenSCENARIO ist als Format zur Beschreibung dynamischer Verkehrsszenarien weit verbreitet und gilt in einigen Forschungsbereichen als Quasi-Standard. Da OpenSCENARIO häufig auf OpenDRIVE-Karten verweist, ist die Entwicklung eines Konverters notwendig. Ein existierender Konverter der TUM von Althoff et al. kann OpenDRIVE in Lanelets konvertieren, jedoch nicht umgekehrt. Um verschiedene Szenarien mit unterschiedlichen Datenbasen einheitlich zu konvertieren, wird daher an einem Konverter zwischen Lanelet und OpenDRIVE gearbeitet. Es wurde eine umfassende Formatanalyse durchgeführt und mögliche Konvertierungsmethoden diskutiert. Zusätzlich dazu wurden Konzepte entwickelt und Frameworks betrachtet sowie ausgewählt, um automatisch konkrete Szenarien, beispielsweise aus dem TAF-Datensatz, in das OpenSCENARIO-Format zu konvertieren.

Die zentrale Herausforderung bei der Konvertierung von Lanelet nach OpenDRIVE besteht in der Zusammenführung der Straßensegmente, um sie optimal an die komplexe Topologie des OpenDRIVE-Formats anzupassen. Im Lanelet-Format bestehen Straßen aus elementaren Straßensegmenten, die jeweils einen Teil einer Fahrspur beschreiben. Durch das Zusammenfügen mehrerer paralleler dieser elementaren Straßensegmente entsteht eine mehrspurige Fahrbahn. Im OpenDRIVE-Format hingegen werden alle Fahrstreifen, die parallel zu einer Referenzlinie verlaufen, zu einem Segment zusammengefasst, einschließlich der Fahrstreifen auf der Gegenfahrbahn. Diese Tatsache und die Konvention zur Definition von zusammenhängenden Straßen machen die Entwicklung eines zuverlässigen Konverters sehr anspruchsvoll.

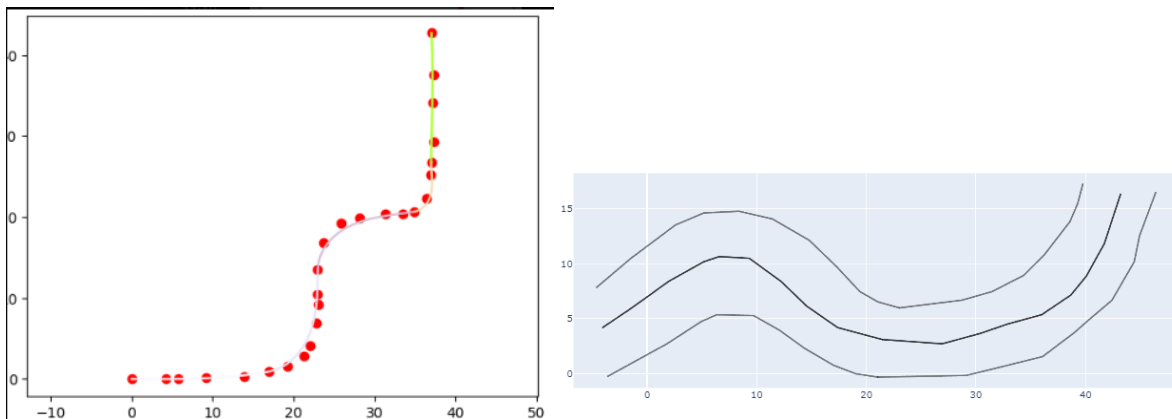


Abbildung 6: Nachahmung der Straßengeometrie mit Clothoiden (links), Straßengeometrie (rechts). Die Orientierung des Straßenabschnitts ist in der linken Abbildung gedreht.

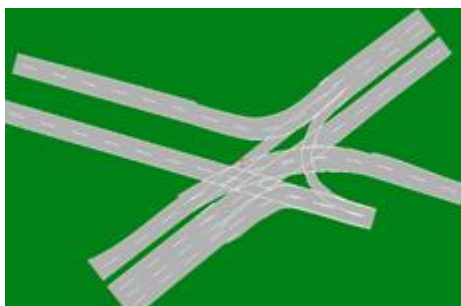


Abbildung 7: Konvertierte Opendrive Karte aus Lanelet

Die derzeit entwickelte Version des Konverters ermöglicht die Zusammenführung mehrerer Lanelets, um sie dem OpenDRIVE-Format anzupassen. Die resultierende Referenzlinie wird dabei durch eine Hermann-Clothoide beschrieben. Dadurch ist es möglich, einfache Straßenzüge erfolgreich zu interpretieren. Dennoch ist die Anwendung noch nicht so weit fortgeschritten, dass beliebige Szenarien, insbesondere Kreuzungssituationen, fehlerfrei abgebildet werden können.

2.1.3.2 Beschreibung und semantische Analyse von Verkehrsszenarien

In TP6 soll die grundlegende Frage, welche Verkehrsszenarien von Relevanz sind und welchen Beitrag sie leisten können, bearbeitet werden. Um dieser Herausforderung zu begegnen, ist es zunächst essenziell, Verkehrsszenen präzise als Datenmodell zu beschreiben. Ein Beschreibungsmodell sollte in der Lage sein, Verkehrsszenen aus verschiedenen Domänen zu vergleichen und eine Bewertung der Verkehrsszenen zu ermöglichen. Herkömmliche Beschreibungsmethoden stoßen hier teilweise an ihre Grenzen, insbesondere wenn es darum geht, Szenen mit unterschiedlichen Straßengeometrien, an verschiedenen Orten oder mit verschiedenen Verkehrsteilnehmern skalierbar zu vergleichen.

Vor diesem Hintergrund ist es von entscheidender Bedeutung, ein Modell zu entwickeln, das die Vielfalt der Verkehrsszenen so beschreiben kann, dass nicht nur ähnliche Szenen identifiziert, sondern auch klassifiziert werden können. Ein weiterer Aspekt ist die Reduzierung der Entitätstypen und Relationstypen auf ein Minimum. In diesem Zusammenhang wird das Konzept des „Semantic Scene Graph-Modells“ (SSG) vorgeschlagen und angewendet. Mithilfe dieses Modells können Verkehrsszenen in Bezug auf die Straßentopologie beschrieben und unabhängig vom Ort miteinander verglichen werden. Die Beziehungen zwischen den Verkehrsteilnehmern werden durch semantisch klassifizierte Kanten dargestellt. Diese abstrakte Beschreibung erleichtert nicht nur die maschinelle Lesbarkeit, sondern ermöglicht auch die Anwendung von Methoden des maschinellen Lernens auf Verkehrsszenen.

Des Weiteren können semantische Szenengraphen von neuronalen Graphennetzen analysiert und verarbeitet werden. Dies erlangt insbesondere im Kontext des autonomen Fahrens und dessen Validierung eine hohe Relevanz. Das Verständnis einer Verkehrsszene spielt eine zentrale Rolle, sowohl für die Suche nach ähnlichen Szenen für eine effiziente Validierung autonomer Fahrfunktionen als auch für die Optimierung der Entscheidungsfindung dieser Fahrfunktionen. Ergebnisse und eine detaillierte Umsetzung der Arbeit wurden in der FZI-Veröffentlichung mit dem Titel "Towards Traffic Scene Description: The Semantic Scene Graph" [15] beschrieben.

Inputdaten

Als Grundlage für die Analyse von Verkehrsszenen werden entsprechende Daten benötigt. Die Voraussetzung für die Verarbeitung der Daten durch das Semantic Scene Graph Framework ist, dass zu jedem gewünschten Zeitpunkt alle Verkehrsteilnehmer i einem diskreten Zustand zugeordnet werden können. Der Zustand $\mathcal{X}^i$ des Verkehrsteilnehmers i ($i \in I$) (Menge aller Verkehrsteilnehmer I) ist wie folgt definiert:

$$\mathcal{X}^i = \{x^i, y^i, \psi^i, \dot{x}^i, \dot{y}^i, \ddot{x}^i, \ddot{y}^i, w^i, l^i, \kappa_{obj}^i\}$$

Die Verkehrsteilnehmer werden ausschließlich in einem planaren $\mathbb{R}^2$ -Koordinatensystem betrachtet. Dabei beschreiben x^i und y^i den geometrischen Mittelpunkt des Verkehrsteilnehmers in einem

lokalen metrischen Koordinatensystem, dessen Ursprung üblicherweise durch einen geodätischen Bezugspunkt definiert ist. ψ^i gibt die Ausrichtung (Gierwinkel) des Fahrzeugs an. $\dot{x}^i, \dot{y}^i$ und $\ddot{x}^i, \ddot{y}^i$ geben die zweidimensionale Geschwindigkeit bzw. Beschleunigung des Verkehrsteilnehmers in Richtung der jeweiligen Achse an. Die Größe der jeweiligen Objektbox wird durch die Breite w^i und die Länge l^i beschrieben. Zusätzlich zu den geometrischen Informationen wird ein Objekt durch seine Klassifizierung κ_{obj}^i (PKW, Fußgänger, LKW, ...) beschrieben. Die Zustände $\mathcal{X}$ aller Verkehrsteilnehmer eines bestimmten Zeitschrittes werden dann zu einer Szene zusammengefasst und durch den Szenenzustand $S_{\mathbb{R}^2}(t)$ beschrieben:

$$S_{\mathbb{R}^2}(t) = \{\mathcal{X}^i(t) | i \in I\}.$$

Architektur:

Neben den Objektlisten werden als Eingangsdaten hochgenaue Straßenkarten benötigt. Dabei ist es besonders wichtig, dass die Straßentopologie, d.h. die Information, welche Fahrspuren wie miteinander verbunden sind, enthalten ist. Die Straßenkarte wird durch einen gerichteten Graphen G_{road} beschrieben. Jeder elementare Straßenabschnitt (Fahrspur) wird durch einen Knoten beschrieben. Verschiedene Straßensegmente sind durch Kanten miteinander verbunden. Die jeweiligen Kantenattribute geben an, wie zwei Straßensegmente miteinander verbunden sind. Zwei Straßenabschnitte können in Bezug auf die Fahrtrichtung nebeneinander und hintereinander liegen. Im letzteren Fall kann der gerichtete Graph den Vorgänger und den Nachfolger eindeutig identifizieren. Darüber hinaus können sich zwei Fahrspuren kreuzen oder überschneiden, beispielsweise an einer Straßenkreuzung. Abbildung 1 zeigt ein beispielhaftes Straßennetz und den daraus resultierenden Straßengraphen, in dem jeder elementare Straßenabschnitt durch einen Knoten dargestellt ist. Jeder Straßenabschnitt in Abbildung 8a ist in Abbildung 8b als Knoten dargestellt. Der Abschnitt A liegt topologisch gesehen neben dem Abschnitt B und verläuft in dieselbe Richtung, so dass die entsprechenden Knoten in einer *adjacent* Beziehung stehen. Die elementaren Straßenabschnitte A, B, G bilden bildlich eine Fahrbahn. Daher sind die entsprechenden Knoten (A, B, G) in Abbildung 1b durch gerichtete, *consecutive* Kanten verbunden.

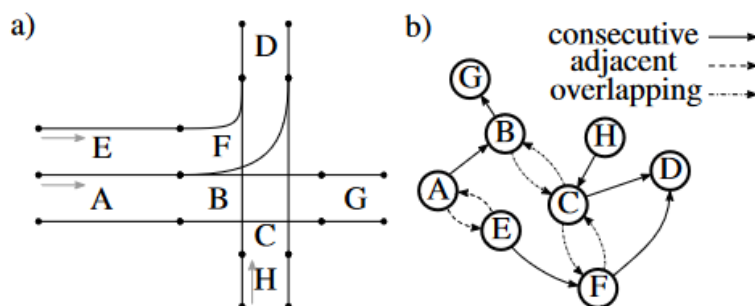


Abbildung 8 Geometrische Abbildung der elementaren Straßenabschnitte (A,B,C,D;E;F), die durch die Fahrspurränder definiert werden (a). Der resultierende Straßengraph von der Straßengeometrie aus a). Mögliche Beziehungen (consecutive, adjacent, overlapping) werden durch Kanten zwischen den Knoten definiert (b)

Für das Verständnis einer Verkehrsszene sind die relativen Abstände und Geschwindigkeiten der Verkehrsteilnehmer, die das Verhalten der anderen beeinflussen, wichtiger als die absoluten Posen im kartesischen Raum. In dem vorgeschlagenen Ansatz werden die Verkehrsteilnehmer, die durch

Objekte im kartesischen Raum beschrieben werden, auf den Frenet-Raum projiziert. Die Mittellinien der Straßenabschnitte stellen die Kurven dar. Aufeinanderfolgende Straßensegmente eines Pfades im Straßennetz werden zusammengefasst und durch eine Kurve dargestellt. Die Projektion ist die erste Phase im Prozess der Kombination der Straßentopologie und der Objektlisten. Jeder Verkehrsteilnehmer einer Verkehrsszene wird individuell einem elementaren Straßensegment zugeordnet, abhängig von seiner Lage im Raum. Eine solche Zuordnung wird als Projektionsidentität definiert. Da bei der Projektion einige räumliche Informationen verloren gehen, muss sichergestellt werden, dass bei mehrdeutigen Posen eines Verkehrsteilnehmers alle potentiell relevanten Projektionsidentitäten berücksichtigt werden. Abhängig von der Position und Orientierung in Bezug auf den jeweiligen Straßenabschnitt wird eine Wahrscheinlichkeit geschätzt, wie gut die tatsächliche Pose des Verkehrsteilnehmers mit der Fahrspur übereinstimmt. Unter der Annahme, dass sich ein Verkehrsteilnehmer in der seitlichen Mitte des Straßensegments befindet und perfekt mit seiner Orientierung übereinstimmt, wäre die geschätzte Wahrscheinlichkeit maximal.

Um aussagekräftige Beziehungen zwischen einzelnen Verkehrsteilnehmern herzustellen, wird deren Position im Verhältnis zu anderen Verkehrsteilnehmern in Bezug auf den Straßenverlauf betrachtet. Entscheidend sind semantische Beziehungen, wie z.B. auf der gleichen Spur, der Nachbarspur, der Parallelspur oder auf einer Kreuzungsspur. Diese Informationen sind im Straßengraphen in Kombination mit den zugeordneten Verkehrsteilnehmern bereits implizit enthalten. Im semantischen Szenengraphen (Semantic Scene Graph) wird jedoch die Beziehungen explizit spezifiziert, um die Darstellung auf die wesentliche Information zu beschränken. Abbildung 9a zeigt eine beispielhafte projizierte Verkehrsszene mit fünf Fahrzeugen. Jedes Fahrzeug, außer Fahrzeug 1, hat der Einfachheit halber nur eine Projektionsidentität. Eine Darstellung des resultierenden Semantischen Szenengraphen ist in Abbildung 9b zu sehen.

Für eine konkrete Beschreibung der Bestimmung der Kantenattribute wird auf die im Rahmen des vorliegenden Projekts entwickelte Veröffentlichung [15] verwiesen.

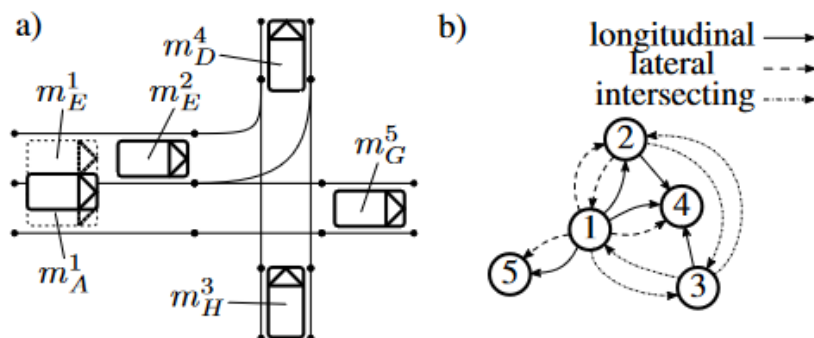


Abbildung 9: Das gleiche Straßennetz wie in Abbildung 8 mit sechs Projektionsidentitäten m von fünf Fahrzeugen (a). Der resultierende Szenengraph, in dem jeder Verkehrsteilnehmer durch einen Knoten und die Beziehungen zwischen seinen Projektionsidentitäten als Kanten dargestellt sind (b)

Analyse:

Im folgenden Abschnitt werden die generierten semantischen Szenengraphen anhand von zwei Anwendungsszenen detailliert diskutiert. In Abbildung 10 sind zwei exemplarische Verkehrsszenen sowohl in der Szenengraphen-Darstellung als auch in der Vogelperspektive dargestellt. Zur besseren Veranschaulichung wurden die beiden sich schneidenden Beziehungen zwischen zwei

Entitäten in dieser Abbildung als bidirektionale Kante repräsentiert. Die erste Verkehrsszene (Abbildung 10a) zeigt mehrere Fahrzeuge, die einen Kreisverkehr befahren. Auf der rechten Seite fahren fünf Fahrzeuge hintereinander, wobei das erste gerade den Kreisverkehr betreten hat. Diese Fahrzeuggruppe (13, 14, 15, 17, 18) ist im Graph durch eine dunklere blaue Farbe hervorgehoben. Diese Gruppe kann im Diagramm durch aufeinanderfolgende Längsbeziehungen separiert werden. Genauer gesagt sollte das Fahrzeug mit der ID 10 ebenfalls zu dieser Gruppe gehören, aber da dieses Fahrzeug keine weiteren Querbeziehungen zu einer anderen Gruppe (3, 16) hat, kann es als zusätzliche Gruppe betrachtet werden. Auf diese Weise können relevante Verkehrsteilnehmer nur nach ihrer Beziehung zueinander gruppiert werden, um verschiedene Verkehrssituationen voneinander zu unterscheiden.

Abbildung 10b zeigt eine zweite Verkehrsszene an einem anderen Ort mit einer sich zum ersten Beispiel unterscheidende Straßengeometrie. Obwohl die Szenen auf den ersten Blick unterschiedlich erscheinen, sowohl hinsichtlich der Straßengeometrie als auch der Position der Verkehrsteilnehmer, lassen sich beim Betrachten des Graphens Ähnlichkeiten in den gegenseitigen Beziehungen der Verkehrsteilnehmer erkennen. In beiden Graphen gibt es drei verschiedene Gruppen von Verkehrsteilnehmern: Eine Gruppe, die in den Kreuzungsbereich einfährt (dunkleres Blau), eine zweite Gruppe, die diesen Bereich überqueren kann/wird (grün), und schließlich eine dritte Gruppe, die den Kreuzungsbereich bereits verlassen hat (hellblau). Dieses Beispiel verdeutlicht, dass mithilfe der generischen Beschreibung des semantischen Szenengraphen zwei verschiedene Verkehrsszenen miteinander verglichen werden können. Es wurde gezeigt, dass die Topologie der Verkehrsszene in einer bestimmten Form in die Graphenstruktur übertragen wurde. Es ist zu beachten, dass die Verkehrsszene ausschließlich durch die Interaktion zwischen verschiedenen Verkehrsteilnehmern und auf Basis ihres aktuellen Bewegungszustands beschrieben wird. Für tiefgreifende Analyse der Graphen wird erneut auf das Paper [15] verwiesen.

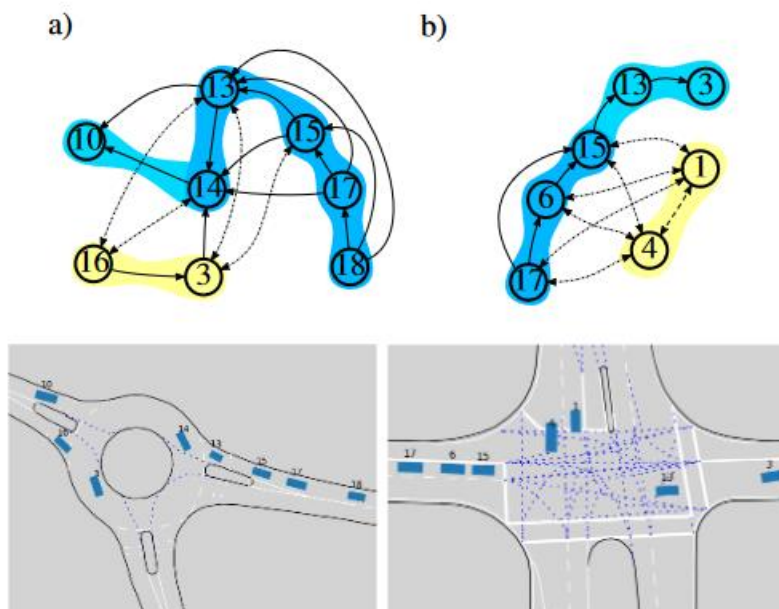


Abbildung 10: Zwei beispielhafte Verkehrsszenen bei unterschiedlichen Straßengeometrien (unten) und die entsprechenden Semantischen Szenengraphen (oben). Die Knoten wurden eingefärbt, um die Visualisierung von Gruppen zu verbessern

2.1.3.3 Bereitstellung von Szenenklassifikationen für die Kritikalitätsanalyse

Die Verifizierung und Validierung des hochautomatisierten Fahrens stellen eine erhebliche Herausforderung für die breite Einführung hochautomatisierter Fahrzeuge dar. Szenariobasierte Tests bieten eine mögliche Lösung, indem sie den Homologationsaufwand für die Systeme reduzieren. Eine entscheidende, bisher nicht komplett gelöste Voraussetzung ist jedoch die Definition und Reduktion des Testraums auf eine endliche Anzahl von Testszenarien. Um dieser Herausforderung zu begegnen, schlagen wir eine Erweiterung eines kontrastiven Lernansatzes vor, der Graphen zur Konstruktion eines sinnvollen Embeddingspace verwendet. Unser Ansatz demonstriert die kontinuierliche Abbildung von Szenen anhand von szenenspezifischen Merkmalen und die Bildung von thematisch ähnlichen Clustern auf der Grundlage der resultierenden Embedding. Basierend auf den gefundenen Clustern können im anschließenden Testprozess ähnliche Szenen identifiziert werden, was zu einer Reduktion von redundanten Testläufen führen kann.

Das Ziel unseres Ansatzes ist es, den Abstand zwischen dem Embedding s_0 , s_+ zweier ähnlicher Graphen, G_0 und G_+ , im Embeddingspace S zu minimieren. Zusätzlich ist es wichtig, dass die Embedding s_- eines unähnlichen Graphen G_- so weit wie möglich von s_0 entfernt ist. Wir erreichen dies durch die Implementierung eines kontrastiven Lernansatzes, um dieses Problem der Graphendarstellung zu lösen.

Graph Augmentierung

Die Objektlistendaten aus einem Bewegungsdatensatz dienen als Grundlage für unsere Methodik. Unser Ansatz beruht darauf die Verkehrsszenen direkt aus den Objektlistendaten (siehe Oben) zu modifizieren. Im Gegensatz zu einer direkten Veränderung der Graphenmerkmale hat dies den Vorteil, dass die Konsistenz und der Realismus der erzeugten Verkehrsszenen besser sind. Das zufällige Hinzufügungen, das Löschungen oder die Änderungen von Kanten und Knoten innerhalb der Graphendarstellung kann zu Verkehrsszenen führen können, die in der Realität unwahrscheinlich oder unmöglich sind.

Beim Augmentierungsprozesses werden Verkehrsteilnehmer zufällig ausgewählt und in ihrer Geschwindigkeit und Position, nach einer Standardverteilung entsprechen augmentiert (modifiziert). Anschließend werden sowohl die augmentierten als auch die ursprünglichen Szenen mit Hilfe des Szenengraphenframeworks (siehe oben) in eine Graphendarstellung umgewandelt.

Modell Architektur

Die Modellarchitektur, die für die Kodierung eines Verkehrsszenengraphen in ein Embedding verwendet wird, ist ein Graph Neural Network mit zwei Schichten, einer Auslesefunktion in Form einer Summierung über alle versteckten Knotenzustände und einem Projektionskopf, wie in Abbildung 3 zu sehen. Das Prinzip des von uns verwendeten Message-Passing-Ansatzes kann mit der folgenden Formel erklärt werden:

$$v_k^i = \gamma_k(v_{k-1}^i, \bigoplus_{j \in \mathcal{N}(i)} (m_k(v_{k-1}^i, v_{k-1}^j, e^{ji}))).$$

k steht für die Schicht des GNN, während i einen bestimmten Knoten angibt. Der erste Schritt bei der Berechnung des neuen Zustands v_k^i besteht darin, Nachrichten entlang aller eingehenden Kanten zu dem Knoten zu generieren. Diese Nachrichtenfunktion m_k kombiniert die Merkmale der ankommenden i_- und abgehenden j -Knoten sowie die Kantenmerkmale e^{ij} . Anschließend wird eine Aggregationsfunktion $\oplus$ angewendet, um die beliebige Menge von Nachrichten auf eine feste Größe des Embeddings des Knotens zu reduzieren. Der sich ergebende Vektor wird dann zusammen mit den Merkmalen der eingehenden Knoten durch die Aktualisierungsfunktion γ_k^i geleitet und ergibt den endgültigen neuen Knotenzustand v_k^i .

In Abbildung 11 ist eine schematische Darstellung der Pipeline für die Generierung des Embeddings aus Graphen abgebildet. Die Blöcke MP_1, MP_2 stellen das Messagepassing dar, welches in der oberen Gleichung genauer definiert worden ist. Der Readoutblock R aggregiert alle einzelnen Embeddings der Knoten welche dann anschließend über ein kleines vollvernetztes Netzwerk in die entsprechenden Graph-Embeddings s überführt wird.

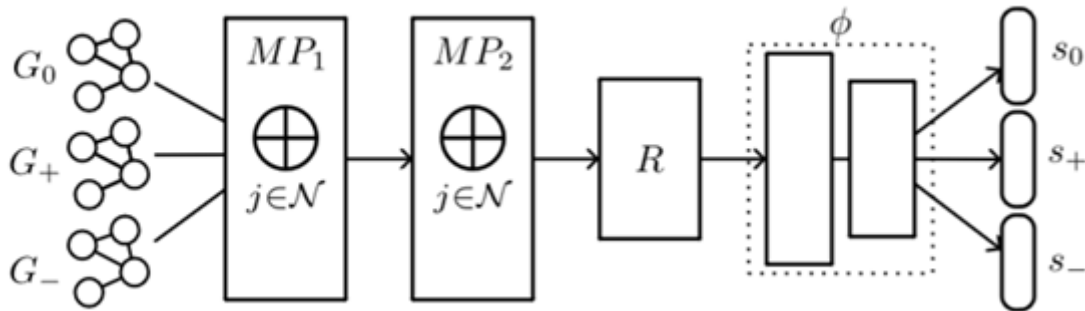


Abbildung 11: Embedding Pipeline für die Graphentriplets [16]

Analyse A. Embedding Space

Unser erstes Ziel ist es, festzustellen, ob das trainierte Modell ähnliche Verkehrsszenen effektiv in den Embedding Space S einbetten kann. Um dies zu beurteilen, führten wir Experimente mit dem Holdout-Set (Datensatz, welches weder zum Validieren, noch zum Training des Netzes verwendet wurde) durch. Um die Unterscheidungsfähigkeit des Netzwerks zu testen, sollten die positiven Samples s_+ einen geringeren Abstand zum Anker s_0 im Embedding Raum aufweisen als die negativen Samples s_- , damit sie korrekt klassifiziert wurden. Wir definieren diese Annahme durch nachfolgende Gleichung:

$$d(s_0, s_+) < d(s_0, s_-)$$

Die Gesamtgenauigkeit des Modells für den Testdatensatz beträgt etwa 99,3 %. In Verbindung mit den konsistenten Abständen zwischen Anker-, Positiv- und Negativproben stellen wir fest, dass unser Modell seine diskriminierenden Eigenschaften verallgemeinert hat.

Das Labeling (händische, auf Regeln basierte Klassifizieren) von Verkehrsszenen stellt ein anspruchsvolles Problem in der Forschung dar. Während es für Szenarien mit einer geringen Anzahl von Fahrzeugen möglich sein mag, eine prägnante Auswahl von regelbasierten Kennzeichnungen

wie "folgen" oder "überholen" zu erstellen, übersteigt die Komplexität unserer Verkehrsszenen dieses bei weitem, da sie oftmals bis zu 40 Fahrzeuge umfassen. Folglich würde sich der Versuch, solche komplexen Interaktionen manuell zu kennzeichnen oder sich auf regelbasierte Methoden zu verlassen, als unproduktiv erweisen. Aus diesem Grund ist ein alternativer Ansatz für die Validierung unerlässlich.

Unser Ansatz verwendet Merkmale, die aus der Struktur des Graphen selbst abgeleitet werden. Diese Merkmale sind zwar nicht so aussagekräftig wie kategorische Bezeichnungen, liefern aber wertvolle Informationen über die zu definierenden Merkmale einer bestimmten Verkehrsszene. Wir verwenden diese Merkmale, um zu bewerten, wie gut unser Embedding Space die Informationen der als Graph repräsentierten Verkehrsszenen erfasst. Wiederum unter Verwendung des Holdout-Sets erstellen wir mehrere Merkmale und verwenden einen kleinen vierschichtigen MLP (30 hidden Dimensions), um von S auf das entsprechende Merkmal zu regressieren. Das Modell wurde anhand des mittleren quadratischen Fehlers (MSE) trainiert. Um die sich daraus ergebenden Fehler in einen Kontext zu bringen, wird der mittlere absolute Fehler (MAE) der Vorhersage in Kombination mit dem Mittelwert, der Standardabweichung (Std), dem Minimal- und Maximalwert des regressierten Merkmals im Testdatensatz angegeben. Durch die Auswertung des absoluten Fehlers im Verhältnis zu den üblichen Bereichen der entsprechenden regressierten Variablen können wir die erhebliche Signifikanz unserer Einbettungen feststellen. Zum Beispiel kann die Anzahl der Autos $|V_{car}|$ mit einer Abweichung von durchschnittlich weniger als einem Auto geschätzt werden. Ähnliche Ergebnisse lassen sich für die Anzahl der Kanten, die Geschwindigkeiten der Autos und die Anzahl der Interaktionstypen ($|E_{lon}|$, $|E_{lat}|$, $|E_{int}|$) feststellen, die auf die Anzahl der Autos in der betrachteten Straße normiert wurden.

Sobald wir die Aussagekraft der im Embedding Space kodierten Informationen bestätigt haben, verlagert sich unser Schwerpunkt auf die Untersuchung der Verteilung dieser Informationen. Von besonderem Interesse ist dabei die Kontinuität innerhalb des Embedding Space. Um diesen hochdimensionalen Raum effektiv zu visualisieren, verwenden wir Techniken zur Reduktion der Dimensionalität, insbesondere die PCA und UMAP, um die Dimensionalität auf zwei Dimensionen zu reduzieren. Anschließend ordnen wir den Proben Farben zu, die auf bestimmten handgefertigten Merkmalen basieren, die zuvor in dem oben genannten Schritt verwendet wurden (siehe Tabelle 1).

Tabelle 1: Regressionsleistung des Embedding Space auf handgefertigte Merkmale [16]

Regr. Var.	MSE	MAE	Mean	Std	Min	Max
$ E_{lon} $	0.572	0.553	1.937	1.37	0.0	8.409
$ E_{lat} $	0.885	0.218	0.243	0.388	0.0	2.067
$ E_{int} $	0.367	0.588	3.195	2.097	0.0	12.85
$ E $	0.108	3.490	58.45	53.611	1.0	398.0
$ V_{car} $	0.289	0.772	8.160	3.75	1.0	22.0
$mean(\vec{v})$	0.434	0.635	4.349	1.863	0.0	10.894

In Abbildung 12 (links) ist das Embedding anhand der Durchschnittsgeschwindigkeit der Autos ($mean(\vec{v})$) eingefärbt. Es ist ein Farbverlauf von links oben nach rechts unten zu beobachten, von hoher Geschwindigkeit zu niedriger Geschwindigkeit. Dies deutet darauf hin, dass der Embedding Space eine gewisse szenenbeschreibende Kontinuität bewahrt. Das zweite wichtige Beispiel wird durch die Einfärbung der Anzahl der in der Szene vorhandenen Autos $|V_{car}|$ sichtbar. Dies ist in Abbildung 12 (rechts) zu sehen. Insbesondere auf der linken Seite der Abbildung sind etwa sechs deutlich voneinander getrennte linienförmige Cluster zu sehen. Wie zuvor ist auch hier ein Gradient

zu erkennen. Diesmal jedoch von links nach rechts. Interessant ist, dass diese Linien die Anzahl der Autos in der Szene in einer geordneten Weise kodieren. Die an der weitesten links liegenden Zeile enthält hauptsächlich Szenen mit einem Auto, und mit jeder Zeile kommt ein weiteres Auto hinzu. Nach sechs Zeilen scheint dieses Muster zu verschwinden. Der Grund dafür könnte die größere Anzahl von Autos in den Szenen sein, was zu komplexeren Interaktionen führen kann. Um zwischen diesen Szenen zu unterscheiden, würde das Modell einen differenzierteren Ansatz erfordern. Bei der Betrachtung weiterer Merkmale können ähnliche Beobachtungen gemacht werden. Zum Abschluss unserer Untersuchung des Embedding Space testen und beobachten wir drei wichtige Eigenschaften unseres Ansatzes. Erstens stellen wir fest, dass ähnliche Graphen im Vergleich zu unähnlichen Graphen eine größere Nähe zueinander aufweisen. Zweitens stellen wir fest, dass unser festdimensionaler Embedding Space wichtige Informationen enthält, die in nachfolgenden Aufgaben effektiv genutzt werden können. Schließlich bestätigen wir, dass der Embedding Space zumindest bei einigen der verkehrsszenendefinierenden Informationen Kontinuität aufweist.

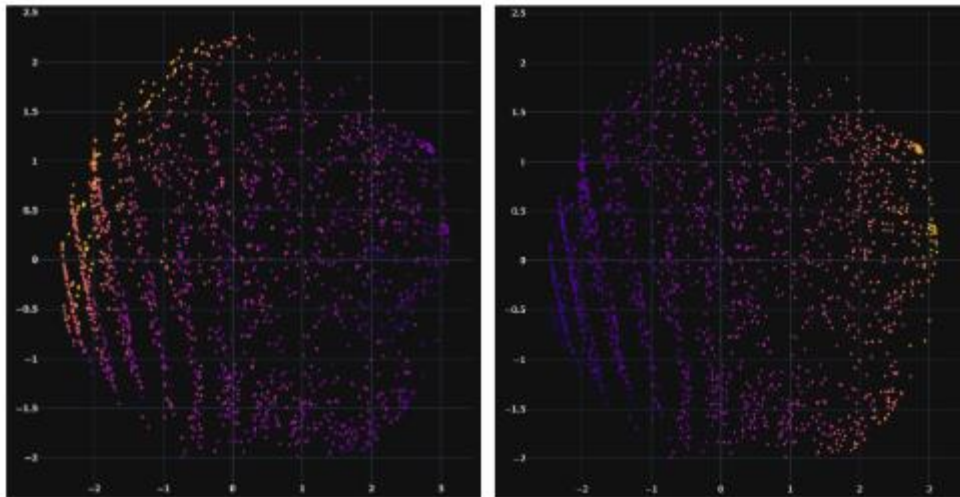


Abbildung 12: Der PCA-reduzierte Embedding Space, eingefärbt nach der Durchschnittsgeschwindigkeit der in der Szene vorhandenen Fahrzeuge (links) und nach der Anzahl der Fahrzeuge (rechts) [16]

Clustering

Um die Kodierungen auf einer tieferen semantischen Ebene qualitativ bewerten zu können, wird ein Clustering auf den Testdatensatz angewandt. Zunächst wird eine Dimensionalitätsreduktion mit dem UMAP-Algorithmus durchgeführt. Auf dem resultierenden zweidimensionalen Raum wird ein agglomeratives hierarchisches Clustering durchgeführt (siehe Abbildung 13).

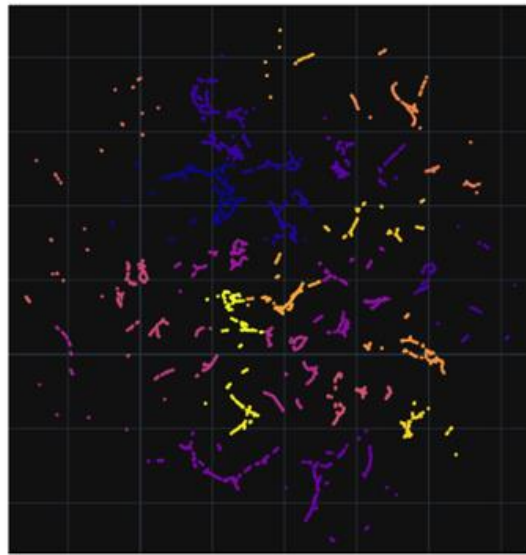


Abbildung 13: Der Embedding Space S wurde mit Hilfe des UMAP-Algorithmus auf einen 2-dimensionalen Raum reduziert. Hierarchisches Clustering wurde eingesetzt, um die Datenpunkte 25 verschiedenen Clustern zuzuordnen, die durch unterschiedliche Farben dargestellt werden [16]

Ergebnisse und eine detaillierte Umsetzung der Arbeiten sind in dem Paper "Traffic Scene Similarity: a Graph-based Contrastive Learning Approach" [17] und in der Arbeit „Self Supervised Clustering of Traffic Scenes using Graph Representations“ [16] beschrieben.

2.1.4 TP7: Testvorgehen

Als wissenschaftliche Teilprojektleitung war das FZI gemeinsam mit der industriellen Teilprojektleitung ZF für alle Aktivitäten in TP7 inhaltlich und organisatorisch federführend verantwortlich. Hierbei waren neben der inhaltlichen und strategischen Ausrichtung des TPs auch die Zusammenführung von Ergebnissen auch in Form von Dokumenten und Deliverables relevant. Die Planung, Durchführung und Moderation von sämtlichen Meetings (TP / AP Regelmeetings, Quartalstreffen, Workshops, Arbeitsgruppen) fiel ebenfalls unter die Aufgaben. Der enge Austausch unter den Teilprojektleitern eng verknüpfter Teilprojekte, insbesondere TP3, TP4 und TP7 als TPs für die Sicherheitsargumentation. Hier erfolgte auch die gemeinsame inhaltliche Ausgestaltung des Halbzeitevents und Abschlussevents sowie die erarbeiteten Storylines hierfür.

Die Ergebnisse des FZI finden sich im Detail in den aus TP7 erarbeiteten Deliverables D04, D07 und D12 wieder. Das D12 mit dem Titel „Test Orchestration and Test Execution – Process, Methods and Example Realizations“ ist wie auch die für das Abschlussevent erarbeiteten Präsentationen und Poster freigegeben und öffentlich verfügbar³.

2.1.5 AP7.1: Stand der Technik und Anforderungsanalyse

2.1.5.1 Erarbeitung des Stands der Technik für das Testen von automatisierten Fahrzeugen

Im Rahmen des AP7.1 wurde der aktuelle Stand der Technik zum Testen von automatisierten Fahrzeugen vom FZI in Form eines Dokuments erarbeitet, welches sich ausführlich mit verschiedenen Aspekten und Prozessen des Testmanagements und der Validierung automatisierter Fahrsysteme beschäftigt. Eine vollständige und umfangreiche Fassung und Dokumentation der

³ <https://www.vvm-projekt.de/veroeffentlichungen>

Inhalte findet sich im Ergebnis E7.1a/b. Es stellt wesentliche Standards und Modelle vor, die in der Automobilindustrie zur Anwendung kommen, darunter ISO 29119 [18] und Automotive SPICE [3]. Diese Standards bieten Rahmenwerke für die Testplanung, -durchführung und -dokumentation. Zudem werden zentrale Begrifflichkeiten im Bereich der Verifikation und Validierung definiert, welche in das VVM-übergreifende Glossar eingeflossen sind.

Die erarbeiteten Ergebnisse bietet einen umfassenden Überblick über die Methoden und Werkzeuge, die zur Validierung und Verifikation (V&V) in der Automobilentwicklung eingesetzt werden. Das Vorgehen stellte sicher, dass alle Testaktivitäten strukturiert und standardisiert durchgeführt werden, um die Zuverlässigkeit und Sicherheit automatisierter Fahrsysteme zu gewährleisten und dient als Basis für die erarbeiteten neuartigen Ansätze in AP7.2.

2.1.5.2 Testkonzept

Das Testkonzept gibt Aufschlüsse über die eingesetzten Mittel, den Ablaufplan der Testaktivitäten, Abgrenzungen und die genaue Vorgehensweise. Es bestimmt die Testobjekte und die Testaufgaben [19]. Das Testkonzept bestimmt, operativ in welcher Testumgebung (z.B. Simulation, Prüfgelände, Feld) die logischen Szenarien mit den zugehörigen Parameterausprägungen getestet werden und stellt die Anforderungen für diese Tests auf [20]. Nach IEEE 829 „Standard of Software Test Documentation“ [21] ist das Testkonzept ein Dokument von acht Dokumenten die für die Dokumentation von Tests standardisiert wurden [22].

Im Folgenden werden verschiedene Methoden zur Überprüfung eines System-under-Test (SuT) beschrieben:

- **Anforderungsbasiertes Testen:** Bei diesem klassischen Ansatz im V-Modell bilden Systemanforderungen die Basis für die Testfallgenerierung. Hauptprinzipien sind „Kein Test ohne Anforderung“ und „Zu jeder Anforderung mindestens ein Test“. Die Testabdeckung wird als Metrik genutzt, um Testlücken zu identifizieren und den Systemreifegrad zu bestimmen [23]. Testfälle bestehen aus Vorbedingungen und Testschritten inklusive Testbedingung.

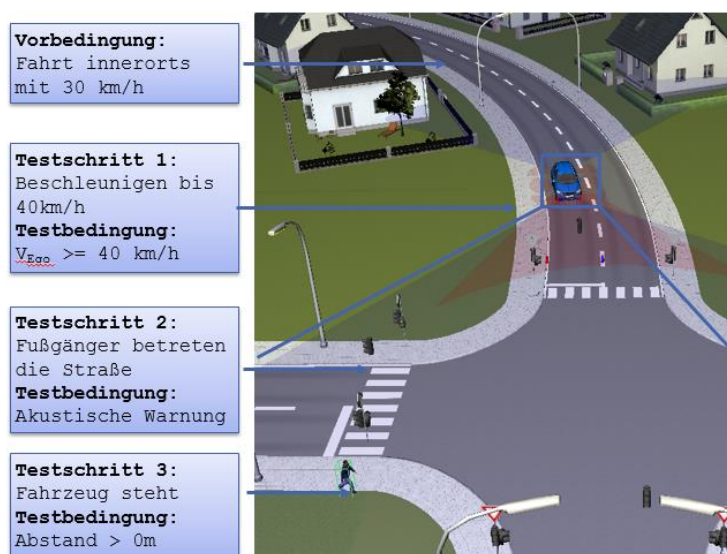


Abbildung 14: Beispiel eines Testfalls für ein Notbremssystem

- Für die Bewertung des Testablaufs ist a priori Wissen über den genauen Testablauf notwendig. In jedem Testschritt wird auf Signalbasis geprüft, ob eine Testbedingung erfüllt

wurde. Die Testfälle werden auf Basis von Systemanforderungen abgeleitet und in Testkatalogen zusammengeführt [24] [25]. Der ISO26262 Standard definiert eine Vielzahl von Möglichkeiten zur Ableitung von Testfällen. Testfälle können zum Beispiel durch die Anforderungsanalyse, durch Use Cases, durch die Erstellung von Äquivalenzklassen oder durch Fehlerabschätzung auf Basis von Expertenwissen generiert werden [26].

- **Statistische Validierung:** Hierbei werden Feldtests auf öffentlichen Straßen durchgeführt und relevante Daten aufgezeichnet [27] [28] [29] [30] [31] [9]. Der „gefahrte Kilometer“-Ansatz nutzt statistische Werte, um die Wahrscheinlichkeit von Fehlern zu schätzen. Da extrem viele Testkilometer nötig wären, wird die Kilometeranzahl durch „disengagement reports“⁴ reduziert, die Situationen berücksichtigen, in denen menschliche Fahrer eingreifen müssen.
- **Szenariobasierte Validierung:** Dieser Ansatz, Gegenstand vieler Forschungsprojekte wie [32] [33] [34] [35], reduziert den Testaufwand durch die Abdeckung mit Szenarien [36]. Es werden Gütekriterien für jedes Szenario definiert und verschiedene Abstraktionsebenen genutzt [37], um den Entwicklungsfortschritt zu messen. Logische Szenarien fassen ähnliche Szenarien zusammen und konkrete Szenarien werden durch Parametervariationen abgeleitet [38].

Funktionale Szenarien	Logische Szenarien	Konkrete Szenarien
Basisstrecke: 3-streifige Autobahn in Kurve Begrenzung auf 100 km/h durch Verkehrszeichen rechts und links	Basisstrecke: Breite Fahrstreifen [2,3..3,5] m Kurvenradius [0,6..0,9] km Pos_Verkehrszeichen[0..200] m	Basisstrecke: Breite Fahrstreifen [3,2] m Kurvenradius [0,7] km Pos_Verkehrszeichen [150] m
Stationäre Objekte: -	Stationäre Objekte: -	Stationäre Objekte: -
Bewegliche Objekte: Ego, Stau; Interaktion: Ego in Manöver „Annähern“ auf mittlerem Fahrstreifen, Stau zählfließend	Bewegliche Objekte: Stauende_Pos [10..200] m Stau_Geschw. [0..30] km/h Ego_Abstand [50..300] m Ego_Geschw. [80..130] km/h	Bewegliche Objekte: Stauende_Pos 40 m Stau_Geschw. 30 km/h Ego_Abstand 200 m Ego_Geschw. 100 km/h
Umwelt: Sommer, Regen	Umwelt: Temperatur [10..40] °C Tröpfchengröße [20..100] µm	Umwelt: Temperatur 20 °C Tröpfchengröße 30 µm



Abbildung 15: Abstraktionsebenen von Szenarien nach [39]

- **Modellbasiertes Testen:** Dieses Konzept vergleicht das Laufzeitverhalten des Systems mit einem prädiktiven Testmodell, das das Systemverhalten beschreibt. Modelle können verschiedene Darstellungsformen haben (natürliche Sprache, Grafiken, mathematische Gleichungen). Das Modell sollte dabei idealerweise nachvollziehbar, wiederverwendbar und eine genaue Beschreibung des SuT beinhalten. Grundsätzlich existiert eine Vielzahl von Modellen, welche immer einen Teilaspekt des Systems beschreiben [25]. Dieser Ansatz ermöglicht eine verbesserte Automatisierung im Testlebenszyklus und eine einfachere Generierung von Testinformationen. Der Ansatz nimmt insbesondere im Bereich der E/E-Entwicklung und Software im Automobil aufgrund der die Zunahme an Systemfunktionen und der steigenden Vernetzung eine zunehmend wichtigere Rolle ein [40].

⁴ <https://www.dmv.ca.gov/portal/vehicle-industry-services/autonomous-vehicles/disengagement-reports/>

- **Formale Verifikation:** Diese Methode versucht, die korrekte Funktionalität formal nachzuweisen [41]. Es wird eine formale Anforderungsspezifikation genutzt, die allgemeingültige Regeln wie „Verursache keinen Auffahrunfall“ enthält. Diese Regeln können zur analytischen Überprüfung des ADAS verwendet werden [42].
- **Shadow Mode:** Das SuT wird in einer Open-Loop-Schleife während realer Testfahrten ausgeführt [43], wobei es Eingabedaten von realen Sensoren erhält, aber keine Ausgabe an die nachfolgenden Elemente weitergibt. Dieser Ansatz testet das SuT unter realen Bedingungen ohne zusätzliche Risiken und vergleicht das Systemverhalten mit dem des menschlichen Fahrers.
- **Virtuelle Testfahrten:** Reale Testfahrten werden in Simulationen übertragen, um beliebige Szenarien zu erzeugen und zu testen, ohne teure reale Testfahrten durchführen zu müssen [44]. Modifizierte aufgezeichnete Daten oder statistische Verkehrsmodelle werden verwendet, um realistische Szenarien zu simulieren [45].
- **Data Replay:** Aufgezeichnete Testdaten aus realen Testfahrten werden wiedergegeben, um notwendige Informationen für die Eingabe des SuT zu liefern [46] [7]. Dieser Ansatz ermöglicht eine automatische Bewertung des Systems und die Wiederholung für nachfolgende Entwicklungsstufen. Virtuelle Objekte wie Fußgänger [47] oder Fahrzeuge [48] können während der Re-Simulation eingefügt werden, um das Szenario zu erweitern.

2.1.5.3 Testmanagement & Testprozess

Das Testmanagement behandelt die Überführung eines oder mehrerer Testkonzepte in einen formalen Testprozess und die Überwachung des Testprozesses. Die ISO 29119 [18] beschreibt ein dreischichtiges Prozessmodell für das Testen, bestehend aus:

- **Organisatorischer Testprozess:** Dieser entwickelt und überwacht Testspezifikationen, einschließlich Testrichtlinien, Teststrategien, Prozesse und Prozeduren. Die Testrichtlinie deckt die Erwartungen der Stakeholder und die bevorzugte Ausrichtung der Teststrategie ab. Sie schafft Rahmenbedingungen für eine kontinuierliche Verbesserung des Testprozesses. Die organisationsbezogene Teststrategie, abgeleitet von der Testrichtlinie, bestimmt die Anforderungen und Beschränkungen für den Testmanagement-Prozess und den dynamischen Testprozess, anwendbar auf alle Projekte innerhalb einer Organisation.
- **Testmanagement-Prozess:** Hier wird eine projektspezifische Teststrategie aus der organisatorischen Teststrategie abgeleitet. Risiken werden identifiziert und analysiert, um die Testplanung zu gestalten. Diese umfasst statische und dynamische Testmethoden, Personalbedarf und Ressourceneinsatz, die in einem Testplan dokumentiert werden. Der Testplan kann sich auf das gesamte Projekt oder spezifische Testphasen beziehen, wie Systemzulassung oder Performance-Tests. ISO 26262 beschreibt zusätzliche Anforderungen an den Testplaninhalt und die zu verwendenden Verifikationsmethoden. Überwachungsmaßnahmen stellen sicher, dass alle Aktivitäten wie geplant durchgeführt werden, und notwendige Änderungen werden durch Steuerungsanweisungen implementiert.
- **Dynamischer Testprozess:** Dieser beschreibt die konkrete Durchführung eines bestimmten Testtyps oder der Tests in einer Testphase. Er besteht aus vier Teilprozessen: Testentwurf und -implementierung, Bereitstellung und Wartung der Testumgebung, Testausführung und Berichterstattung der Testergebnisse.

Zusätzlich zu ISO 29119, gibt es das den Standard nach IEEE 829 [21] und Automotive SPICE [3]:

- IEEE 829: Fordert acht Dokumente in drei Kategorien: Testkonzept (Testplan), Testspezifikation (Testentwurf, Testfälle, Testablauf) und Testbericht (Testübergabe, Testprotokoll, Testabweichungen, Testabschlussbericht).
- Automotive SPICE: Definiert ein Prozessreferenz- und Bewertungsmodell für Entwicklungsprozesse in der Automobilindustrie, fokussiert auf System- und Softwareentwicklung. Es teilt Prozesse in primäre, unterstützende und organisatorische Prozesse und legt Wert auf Traceability und Konsistenz zwischen den Ergebnissen der Prozesse.

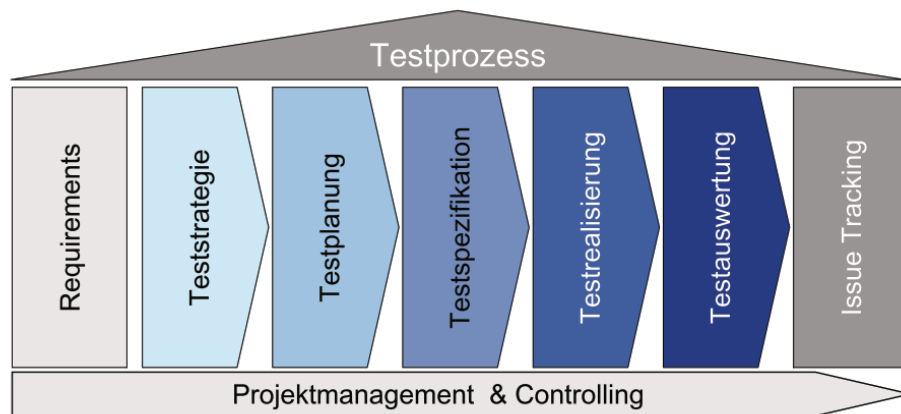


Abbildung 16: Testprozess [2]

Diese Standards und Modelle bieten umfassende Richtlinien und Verfahren, um Testprozesse in verschiedenen Kontexten zu planen, durchzuführen und zu überwachen. Ein idealtypischer Testprozess besteht aus mehreren Phasen und kann wie in Abbildung 16 dargestellt werden. Das Testmanagement erstreckt sich von der Testplanung über die Erstellung der Testspezifikation und Durchführung der Tests bis hin zur Protokollierung der Testergebnisse und der Auswertung der Tests. Die einzelnen Phasen umfassen zusammengefasst folgende Prozess und Ziele, detaillierte Beschreibungen finden sich in [2].

2.1.5.4 Testinstanzen & Testmittel

Die Klassifizierung von Validierungsansätzen, Testmethoden oder Testumgebungen ist Gegenstand mehrerer Studien [42]. Ein Whitepaper von elf führenden Unternehmen der Automobilbranche liefert eine Klassifizierung von Testumgebungen hinsichtlich der verwendeten Stimuli [49], siehe Abbildung 17. Eine Testumgebung besteht aus einem Prüfling und einer Testplattform. Je nach Kombination von Testplattform und Testobjekt werden entweder reale, virtuelle, keine oder Roboter-Stimuli eingesetzt. In der Studie werden verschiedene XIL-Technologien, Software- und Hardwareaufbereitung, Testgelände und offene Straße als Testplattformen betrachtet. Als Testobjekte werden die Zielsoftware und -hardware sowie das Fahrzeug, der Fahrer und die Fahrumgebung untersucht.

Test Platform and Test Item					
Test Item	Target SW (Code)	Target HW (ECU)	Vehicle	Driver	Driving Environment
SiL (Simulation in the Closed Loop)	Virtual	Virtual	Virtual	Virtual	Virtual
	Real			None	
SW Repro (Software Re-processing)	Virtual	Virtual	None	None	Virtual
	Real				
HiL (Hardware in the Closed Loop)	Real	Real	Virtual	Virtual	Virtual
				None	
HW Repro (Hardware Re-processing)	Real	Real	None	None	Virtual
DiL (Driver in the Loop)	Real	Virtual	Virtual	Real	Virtual
		None	None		Real
PG (Proving Ground)	Real	Real	Real	Real	Real
				Robot	
OR (Open Road)	Real	Real	Real	Real	Real

Abbildung 17: Testinstanzen und Testitems angelehnt an [49]

Das Projekt PEGASUS [39] schlägt eine Klassifizierung von Testplattformen nach der Anzahl der realen Komponenten, der Realitätsnähe und dem Aufwand vor. Im Gegensatz zur Klassifikation von Wood ist die Testplattform keine Klassifikationsdimension mehr, sondern ein zu klassifizierender Gegenstand. Außerdem werden alle Klassifikationsdimensionen als stetig angenommen. Obwohl die Eigenschaft der Kontinuität eine genauere Zuordnung der Testplattformen ermöglichen mag, erschwert sie aus unserer Sicht eine fundierte Zuordnung. Im Hinblick auf das Testen und die Identifikation von Emergenzen hängt der Integrationsgrad des Systems weniger von der reinen Anzahl der realen Komponenten ab als vom Vorhandensein von Ereignisketten oder Funktionsclustern.

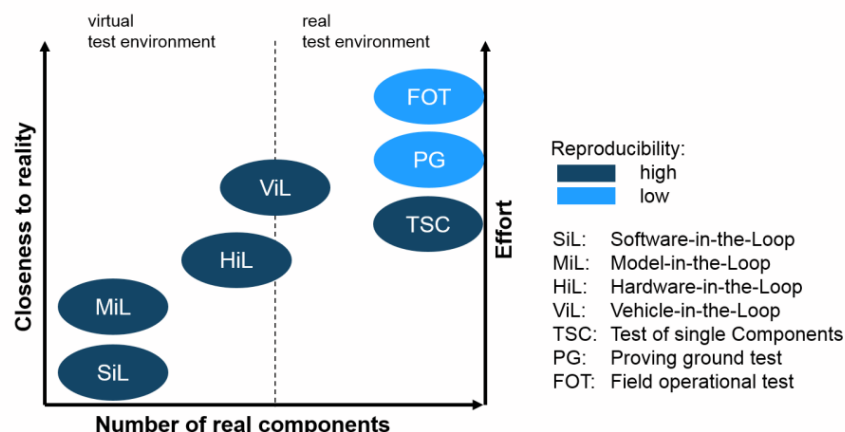


Abbildung 18: Klassifikation von Testumgebungen angelehnt an PEGASUS [39]

Steimle stellt eine Methode zur Klassifizierung von Testumgebungen in einem szenariobasierten Testansatz vor [50]. Basierend auf einer funktionalen Systemarchitektur eines automatisierten Fahrzeugs aus [51] wird eine Methode vorgeschlagen, die 10 verschiedene

Klassifizierungsdimensionen unterscheidet und Radardiagramme für die Darstellung verwendet. Die vorgeschlagenen Dimensionen können grob in zwei Gruppen unterteilt werden: Dimensionen, die sich auf das Ego-Fahrzeug beziehen (wie Testobjekt, Restfahrzeug, Fahrzeugdynamik, Umgebungssensorik, Lokalisierungssensorik) und Dimensionen, die sich auf das Testszenario beziehen (wie Fahrer, Landschaft, bewegliche Objekte, Umweltbedingungen, V2X-Kommunikation). Jede Dimension wird in Anlehnung an [52] [42] auf einer nominellen Skala von drei diskreten Stufen bewertet. Die vorgeschlagenen Stufen sind: simuliert, emuliert und real.

Eine Taxonomie der Testansätze für die Validierung von ADAS wird in [53] vorgeschlagen. Dafür werden drei Dimensionen definiert: Testkriterien und -metrik, Referenzierung und Testszenario. Hinsichtlich der Testkriterien und -metriken unterscheidet die Taxonomie zwischen nicht-technischen Testkriterien, technischen Kriterien auf Systemebene und Metriken für Teilsysteme. Der Begriff Referenzierung bezieht sich auf die Generierung der für den Test benötigten Ground Truth. Die Grundwahrheit kann durch Messung mit bordeigenen oder externen Referenzsensoren, durch Simulation oder durch eine Kombination aus Messung und Simulation erzeugt werden. Bei der Kombination von Messungen und Simulationen werden zwei Konzepte unterschieden: XIL und Augmentation von Messdaten. Nach [53] umfasst ein Testszenario alle relevanten Bedingungen der Testplattform, unter denen das SUT während des Tests arbeitet. Relevante Bedingungen sind zum Beispiel: Straßentyp, Wetterbedingungen, statische Umgebung und andere Verkehrsteilnehmer. Die Testszenarien werden im Hinblick auf ihren Umfang und ihre Definition weiter analysiert. Es werden zwei mögliche Ansätze für die Szenariodefinition vorgestellt. Der erste Ansatz ist wissensbasiert, wobei bekannte Use-Case-Kataloge oder Metamodelle zum Aufbau von Testszenarien verwendet werden. Zweitens wird ein datengetriebener Ansatz vorgestellt, der Datenbanken mit Fahrszenarien verwendet.

2.1.5.5 Ontologien für Szenarien im automatisierten Fahren

Um den derzeitigen Stand der Forschung in Bezug auf Ontologien für das szenariobasierte Testen von hochautomatisierten Fahrzeugen abzugleichen und auf dessen Basis die Testszenarien zu definieren, wurde eine Survey über bestehende Ontologien verfasst. Diese umfasst die wichtigsten Vertreter in diesem Gebiet, aber auch Ontologien die im Rahmen des Projekts VVM selbst entwickelt wurden.

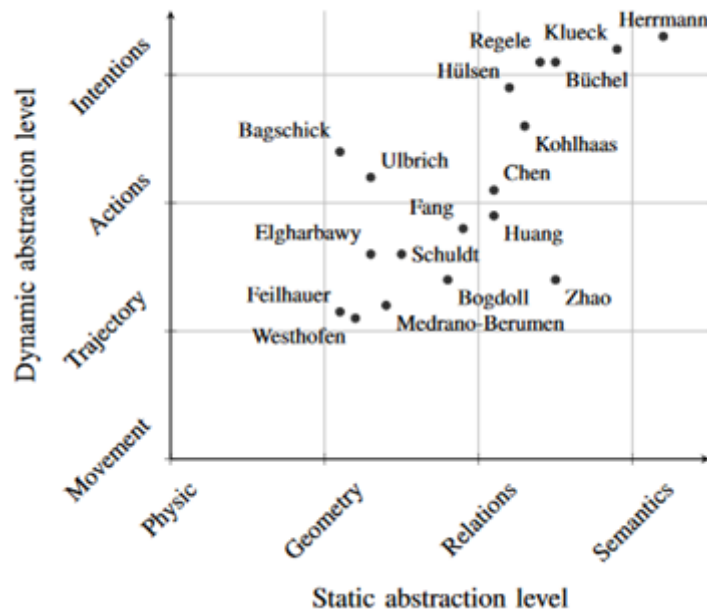


Abbildung 19: Raum für die Kategorisierung von Szenario-Ontologien auf der Grundlage des statischen und dynamischen Abstraktionsebene. Die Realität befindet sich im Ursprung.

Die Ontologien wurden anhand der Abbildbarkeit von wichtigen Entitätsklassen und Relationen bewertet und anhand ihres Detaillevels der dynamischen und statischen Aspekte kategorisiert. In Abbildung 19 ist diese Kategorisierung abgebildet. Die Untersuchung des Stands der Technik bzgl. Szenarien-Ontologien des FZI ist in Form eines Survey-Papers veröffentlicht worden [54], weitere Details sind dort dargestellt.

2.1.5.6 Priorisierung und Ableitung von Anforderungen

In gemeinsamen Abstimmungsrunden wurden die aufbereiteten Inhalte, Erkenntnisse und Methoden diskutiert und priorisiert auf die dringendsten Bedarfe, um eine gemeinsame Grundlage für die zu erarbeiteten notwendigen Methoden für den Test in AP7.2 zu erzielen. Zudem sind aus der Aufbereitung Anforderungen ausgearbeitet worden, insbesondere im Hinblick auf die Verwendung verschiedener Testinstanzen, von der Simulation bis zum Realtest. Hierbei wurde zwischen verschiedenen Arten von Anforderungen an Tests unterschieden und eine Klassifizierung vorgenommen, ob die Anforderung die Anwendung eines Standards bzw. die Durchgängigkeit im Test betrifft, ob sie die Verknüpfung von Szenarien und Testfällen adressiert oder ob sie die konkrete Ausführung eines Testfalls adressiert. Eine vollständige Auflistung der Anforderungen findet sich in Deliverable D04, ein exemplarischer Auszug ist in Tabelle 2 dargestellt:

Tabelle 2: Auszug von abgeleiteten Anforderungen an Tests und Testinstanzen

Category	Hierarchization	Requirement	Motivation, background
Verwendung von Standards	High Level	Die Beschreibung von Szenarien soll offen zugängliche Standards verwenden.	Einheitliches partnerübergreifendes Verständnis von Szenario Inhalten. Kompatibilität von Szenarien zu unterschiedlichen Testinfrastrukturen (Simulationsumgebungen, Prüfgelände, ...)

Verwendung von Standards	High Level	Funktionale Szenarien müssen mit einem eindeutigen und abgestimmten Vokabular beschrieben werden.	Einheitliches Verständnis muss sichergestellt werden
Verwendung von Standards, Schnittstellen	High Level	Logische Szenarien müssen auf Parameterebene beschrieben werden. Zusätzlich sollen Parameterbereiche, statistische Verteilungen sowie Parameterabhängigkeiten darstellbar sein.	Logische Szenarien sollen als Grundlage für eine Konkretisierung genutzt werden können. Sie spannen den abzutestenden Raum auf.
Verwendung von Standards	High Level	Bewusste Unterscheidung von Szenarien und Testfällen. Szenarien sind einer von mehreren Bestandteilen von Testfällen. Testfälle haben u.a. Bestehenskriterien und können "bestanden" werden.	Einheitliches partnerübergreifendes Verständnis / Glossar
Argumentation	High Level	Im szenariobasierten Testen sollen Szenarien zu Testfällen erweitert werden. Nur Testfälle ermöglichen eine Argumentation zu Bestehen / nicht Bestehen, Szenarien beschreiben lediglich Abläufe von Verkehrsgeschehen. Dabei soll ein Testfall neben einem Szenario auch Informationen • zum System Under Test (SUT), • einem Testziel, • SUT Performance Metriken und • Anforderungen an eine Testinfrastruktur enthalten.	Herstellen einer argumentativen Verbindung zwischen Szenarien und einer Testargumentation durch zusätzliche Testinformationen. Erst durch diese Zusatzinformationen können Testfälle und damit die darin enthaltenen Szenarien "bestanden" werden, um auf eine übergeordnete (Sicherheits-) Argumentation einzahlen zu können.
Argumentation, Schnittstellen	High Level	Konkrete Szenarien müssen als Grundlage für (konkrete) Testfälle verwendbar sein	Durchgängigkeit der Artefakte
Schnittstellen, Argumentation, Nachverfolgbarkeit	Technical Implementaiton	Die Testplattform muss Rückgabe von Messergebnissen (Signale, Testergebnisse, eindeutige Identifizierung, ...) in Referenz zum ursprünglichen Testfall bzw. Szenario liefern.	Für weitergehende Analysen. Traceability

2.1.6 AP7.2: Entwicklung Testprozesse und -methoden

Im Rahmen des AP7.2 wurden zentrale Konzepte und Methoden für den Test und die Absicherung von automatisierten Fahrfunktionen im urbanen Raum entwickelt. Das FZI verfolgte hier folgende Ziele und Schwerpunkte:

- Mitarbeit am Gesamtprozess in der Domäne V&V, Einbettung in die VVM Gesamtmethodik sowie Definition der Teilschritte, insbesondere funktionale und technische Testspezifikation (siehe Abschnitte 2.1.6.1 und 2.1.6.2)
- Durchgängige Verknüpfungen von Testinfrastrukturen von der Simulation bis hin zum Realtest für ein „Seamless Testing“, Konzeption und Methodik für eine Auswahl von Testkonfigurationen und Testinfrastrukturen (siehe Abschnitt 2.1.6.4)

- Entwicklung von Metriken und Bewertungskonzepten für die Analyse des Testraums, Abdeckung und Relevanz von Fahrsszenarien sowie der Bewertung von Testinstanzen und Testorchestrierungen für den durchgängigen Test (siehe Abschnitte 2.1.6.3, 2.1.6.8 und 2.1.6.10)
- Durchgängige Verknüpfung und Abbildung von Szenarienbeschreibungen in virtuelle Testumgebungen, Abbildung von statischen und dynamischen Elementen in eine reaktive Umgebung sowie der Variation und Generierung von Fahrsszenarien (siehe Abschnitte 2.1.6.5, 2.1.6.6 und 2.1.6.7)
- Ganzheitliche System- und Funktionsbewertungen für ein „Seamless Testing“ als Durchgängigkeit zwischen verschiedenen Testinstanzen, Bestimmung eines Reifegrads der Funktion (siehe Abschnitte 2.1.6.3, 2.1.6.9 und 2.1.6.12)
- Ontologie zur hierarchischen Bewertung von Erprobungsfahrten (siehe Abschnitt 2.1.6.11)

2.1.6.1 Mitarbeit und Konzeptionierung Gesamtprozess Domäne V&V

Im Rahmen des Projektes VVM wurde eine Gesamtmethodik entwickelt, welche eine zusammenfassende Darstellung der Prozesse bietet, die von der Konzeption bis zur Freigabe eines technischen Systems notwendig sind. Als Teil des VVM beantwortet es die Frage, was getan werden muss, um ein marktreifes ADS Level 4/5 zu entwickeln. Das „Warum“ ist implizit enthalten. Über die einzelnen Prozessschritte hinaus enthält die Übersicht aber auch deren wesentliche Verknüpfungen, die die logische inhaltliche Abhängigkeit der einzelnen Schritte widerspiegeln. Aus Gründen der Übersichtlichkeit werden diese Abhängigkeiten im Verlauf nur vorwärts dargestellt, d.h. Iterationsschleifen und Rückkopplungen an den Schnittstellen sind natürlich vorhanden, aber nicht gesondert abgebildet.

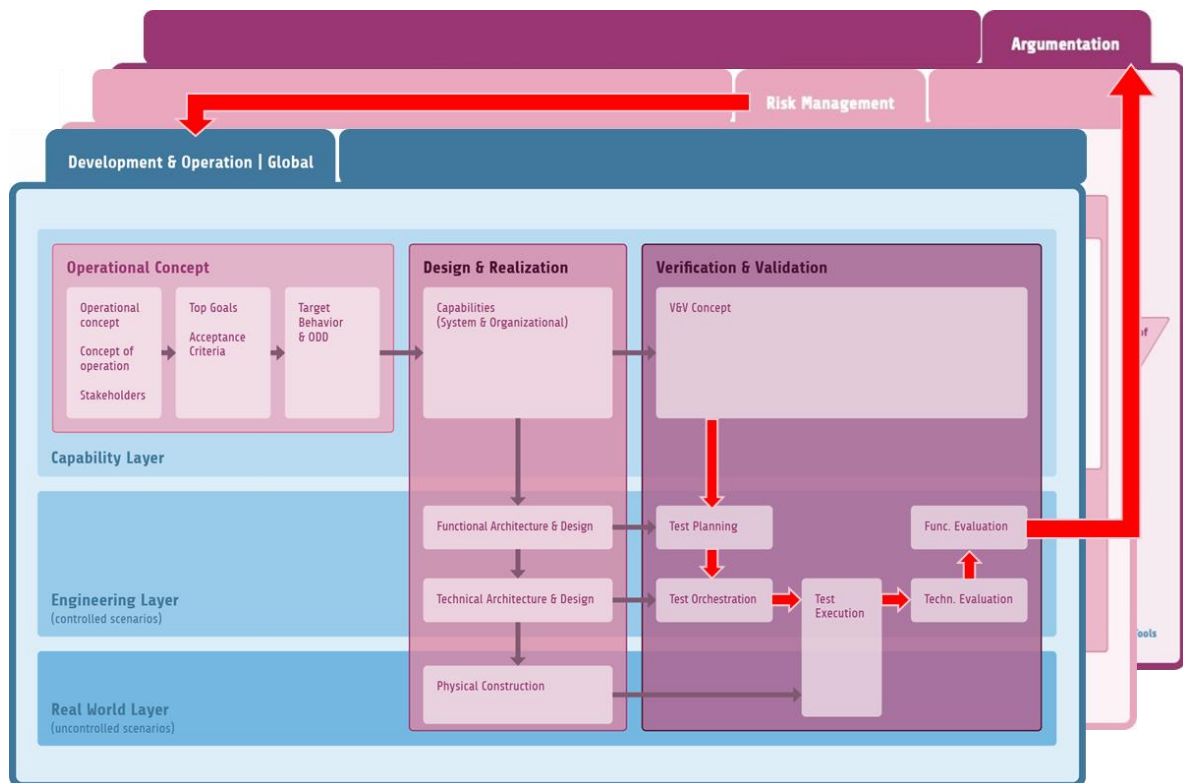


Abbildung 20: Einordnung der V&V Anteile in der Gesamtmethodik

Das FZI war hierbei maßgeblich an den Diskussionen zur Integration und Harmonisierung der V&V-Anteile beteiligt. Für die spezifischen Inhalte wurde das V&V-Konzept in drei große Teilprozesse heruntergebrochen:

- Testplanung
- Testorchestrierung
- Testdurchführung

Die Ausarbeitung der Prozessstruktur sowie Ausgestaltung der einzelnen Teilaspekte und Harmonisierung in der Gesamtmethodik war Kernaufgabe des TP7, hier war das FZI stark aktiv. Die detaillierten Prozess- und Arbeitsschritte für die spezifischen Prozesse im Block V&V sowie jeweilige In- und Outputs sind in Deliverable D12 dargestellt. Hier sind auch alle vollständigen Subprozesse der drei Blöcke detailliert dargestellt.

Zusätzlich zu den Prozessen erfolgte die Aufteilung der Testumfänge in Hierarchieebenen. Das erarbeitete Konzept erlaubt es, die Abstraktionsebenen in der Entwicklung im Sinne einer funktionalen und technischen Ebene auch auf Tests und Testumfänge zu spiegeln. Hierfür erfolgte eine Unterteilung in eine funktionale und eine technische Ebene auch für Tests und Testspezifikationen. Die Testorchestrierung ist hierbei der entscheidende Prozessschritt, da sie aus den funktionalen Testspezifikationen mit Berücksichtigung des technischen Designs, des physikalischen Designs sowie der verfügbaren Testplattformen eine Verteilung von Tests auf unterschiedliche Instanzen vornimmt, welche im Anschluss in technischen Testspezifikation für eine Ausführung beschrieben werden.

2.1.6.2 Funktionale und Technische Testspezifikation

Im Gesamtprozess von VVM stellt die technische Testspezifikation (TT) das Austauschdokument zwischen Testorchestrierung und Testdurchführung dar. Die TT fungiert somit als Kommunikationslinie zwischen Testmanagern und Testingenieuren. Mit dem Format und der Umsetzung der TT werden hohe Ziele verfolgt. Zum einen soll es die Konsistenz der Sicherheitsargumentation über die Testplattformen SiL, HiL, Testgelände und Realverkehr hinweg ermöglichen. Im Hinblick auf die Testautomatisierung muss der TT daher zumindest teilweise sowohl von Menschen als auch von Maschinen lesbar und interpretierbar sein. Diese Durchdringung öffnet die Tore für eine Neuausrichtung der notwendigen Tests für V&V. Die Tests verlagern sich von kostspieligen und zeitintensiven Tests, wie z. B. im realen Straßenverkehr, hin zum massiven Einsatz von SiL, insbesondere von ganzheitlichen Simulationen der Wahrnehmung, Bewegungsplanung und Betätigung in realen Umgebungen. Auf der anderen Seite muss sich das TT in eine lange Geschichte verschiedener Formate und Konzepte zur Testbeschreibung einreihen, die von verschiedenen Projektpartnern, aus früheren Forschungsprojekten wie dem PEGASUS-Projekt und aus bestehenden Standards und Normen wie ISTQB®, OSI, FMU, OpenX stammen.

Der TT muss für verschiedene Arten von Prüfungen anwendbar sein:

- Komponenten- und Systemtests
- Szenariobasierte und nicht-szenariobasierte Tests
- Open-Loop- und Closed-Loop-Tests

Vor diesem Hintergrund wurde ein Vorschlag für das Format des TT mit erarbeitet und exemplarisch auf den vier Testplattformen erfolgreich angewendet.

In verschiedenen multilateralen Diskussionsgruppen wurden für die VV-Methoden folgende Paradigmen für die Erstellung des TT entwickelt:

- Die Testingenieure müssen allein durch den TT bereits über ausreichende Informationen verfügen, um einen Test mit einem vordefinierten Testziel durchzuführen.
- Der TT beschreibt nur eine Testinstanz, da ein Ingenieur typischerweise nur in einer Testinstanz über Expertise verfügt. Für direkte Vergleichsmessungen verschiedener Testinstanzen, wie z. B. eine Simulation in SiL und ein Test im realen Verkehr, sind zwei analoge TTs zu erstellen.
- Der TT enthält mindestens einen Testfall, der durch festgelegte Variationsregeln aus der funktionalen Testspezifikation auf bis zu N Testfälle erweitert werden kann. Jeder einzelne Testfall liefert dann Messdaten, die in der Testauswertung gemeinsam ausgewertet werden können.
- Die Testauswertung ist grundsätzlich Teil der TT und sollte die Testziele widerspiegeln.
- Prinzipiell können einzelne Felder in bestimmten TT leer bleiben. Dies ermöglicht diverse Ableitungen des TTs in einem Format.

Um die Anforderung der Maschinenlesbarkeit zu erfüllen, wurde ein JSON-Format (JavaScript Object Notation) vorgeschlagen, das bereits von einzelnen Projektpartnern zur Beschreibung von Tests verwendet wird. JSON ist ein offenes Standard-Dateiformat und Datenaustauschformat, das menschenlesbaren Text zur Speicherung und Übertragung von Datenobjekten verwendet. Es ist ein weit verbreitetes Datenformat mit vielfältigen Anwendungen im elektronischen Datenaustausch, einschließlich Webanwendungen mit Servern. Daher ist es ein vielversprechender Kandidat für die zukünftige Anbindung an die verschiedenen Schnittstellen aktueller und zukünftiger Testinstanzen.

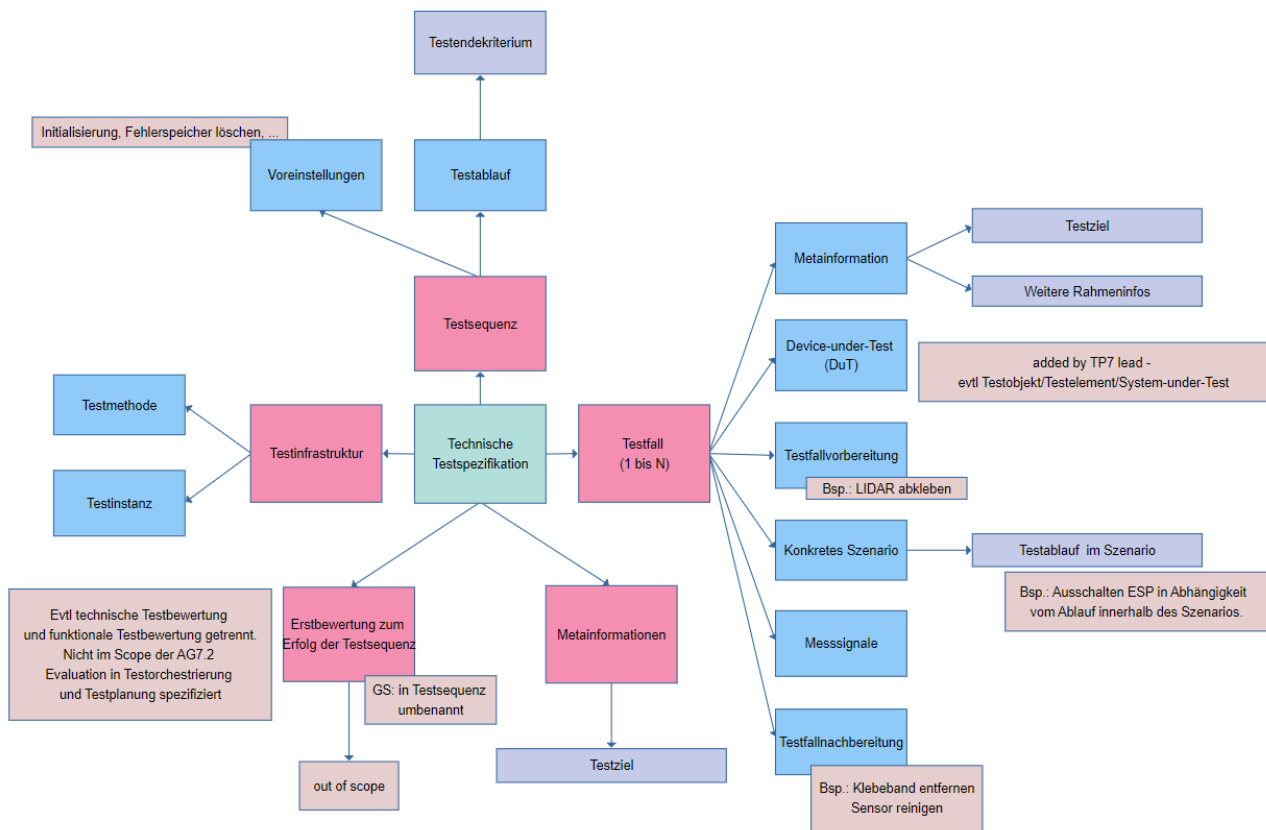


Abbildung 21: Hierarchische Repräsentation der Kapitel der technischen Testspezifikation

Die Kapitelstruktur des TT ist in Abbildung 4 entsprechend dem Stand der Arbeiten dargestellt. Sie gliedert sich in fünf Hauptkapitel: Metainformation, Testfall, Testinfrastruktur, Testablauf und Erste Bewertung des Erfolgs des Testablaufs. Den Hauptkapiteln (rote Kästen) sind bis zu zwei weitere Ebenen von Unterkapiteln zugeordnet. Die Auswertung von Testdaten und die Erstellung von gesammelten Ergebnissen eines Tests wird im TT explizit nicht beschrieben. Alle weiteren Details und spezifische Inhalte der Testspezifikationen sowie dessen Strukturierung finden sich in den Deliverables D07 und D12. Ein Auszug einer beispielhaften Testspezifikation für eine SiL-Umgebung am FZI im Functional Use Case 2.3 und dem Test einer Notbremsfunktion ist in Tabelle 3 dargestellt.

Tabelle 3: Auszug einer exemplarischen technischen Testspezifikation für die Umfänge des FUC2.3 anhand eines AEB

ID	Metainformation	Concrete scenario
FUC_23_TC_1	Test goal: Check if the AEB requests the front collision warning (FCW) and the emergency braking at the correct TTC Additional Information: TTC for FCW and emergency braking depend on the velocity: $v_{Ego} = 30 \text{ km/h}$ $TTC_{FCW} = 1.x$, $TTC_{AEB} = 1.y$	Logical Scenario: FUC2-3 Parameters: $v_{Ego} = 30 \text{ km/h}$ $v_{Ped} = 2 \text{ m/s}$ $d_{VehPed} = 15 \text{ m}$ Test sequence:

	- Signals: - Sw_BrakeMode: 2 = AEB only, 3 = AEB off - AEB: 0 = Off, 1 = On - FCW: 0 = Off, 1 = On	- Accelerate to velocity set point (v_Ego) - Turn on AEB with ECU commands: 1010 1031 1020
FUC_23_TC_2	Test goal: Check if the AEB requests the front collision warning (FCW) and the emergency braking at the correct TTC Additional Information: - TTC for FCW and emergency braking depend on the velocity: v_Ego = 40 km/h TTC_FCW = 1.x, TTC_AEB = 1.y - Signals: - Sw_BrakeMode: 2 = AEB only, 3 = AEB off - AEB: 0 = Off, 1 = On - FCW: 0 = Off, 1 = On	Logical Scenario: FUC2-3 Parameters: - v_Ego = 40 km/h - v_Ped = 2 m/s - d_VehPed = 15 m Test sequence: - Accelerate to velocity set point (v_Ego) - Turn on AEB with ECU commands: 1010 1031 1020
FUC_23_TC_3	Test goal: Check if the AEB requests the front collision warning (FCW) and the emergency braking at the correct TTC Additional Information: - TTC for FCW and emergency braking depend on the velocity: v_Ego = 30 km/h TTC_FCW = 1.x, TTC_AEB = 1.y - Signals: - Sw_BrakeMode: 2 = AEB only, 3 = AEB off - AEB: 0 = Off, 1 = On - FCW: 0 = Off, 1 = On	Logical Scenario: FUC2-3 Parameters: - v_Ego = 30 km/h - v_Ped = 4 m/s - d_VehPed = 15 m Test sequence: - Accelerate to velocity set point (v_Ego) - Turn on AEB with ECU commands: 1010 1031 1020

2.1.6.3 Durchgängiges Testen & Evaluation und Bewertung von Testinstanzen

Der Begriff des durchgängigen Testens umfasst die ergänzende Argumentation auf der Grundlage des in VVM erarbeiteten Testprozesses. Dies soll auch durch die nahtlose Nutzung verschiedener Testplattformen unterstützt werden. Die folgenden Schlüsselthemen in Bezug auf nahtloses Testen werden adressiert:

- **Vertikale Aspekte:** Nahtloses Testen soll eine komplementäre Argumentation ermöglichen, die von der Testorchestrierung über die Testdurchführung bis zur Testauswertung reicht, wie in Abbildung 22 dargestellt. Dies beinhaltet eine Aufschlüsselung von funktionalem Design und funktionalem Testen zu technischem Design und technischer Testspezifikation. Die Testdurchführung erfolgt parallel auf verschiedenen Testplattformen und nach verschiedenen technischen Testspezifikationen. Anschließend muss eine Bewertung jedes Tests auf der Grundlage verschiedener Metriken erfolgen, die von den entsprechenden System-KPIs abgeleitet werden. Zusätzlich müssen alle einzelnen Testergebnisse im Hinblick auf die gesamte Sicherheitsargumentation harmonisiert werden. Der nahtlose Ansatz sollte eine harmonisierte und nachvollziehbare
- **Horizontale Aspekte:** Für die Durchführung von Tests auf technischer Ebene stehen unterschiedliche Testplattformen zur Verfügung. Dies reicht von der Simulation in einer virtuellen Umgebung über X-in-the-Loop-Ansätze auf einer halbvirtuellen Ebene bis hin zu Fahrzeugtests in der realen Welt, die auf Testgeländen oder öffentlichen Versuchsflächen durchgeführt werden können.



Abbildung 22: Seamless Testing über verschiedene Testinstanzen

Die Testplattformen sollten komplementär und durchgängig genutzt werden. Ein Beispiel ist die nahtlose Testdurchführung bestimmter Tests oder Szenarien auf verschiedenen Testplattformen, indem dieselbe Szenariobeschreibung, dieselben Modelle oder dieselben Metriken / KPIs verwendet werden. Ein weiterer Aspekt betrifft die Vergleichbarkeit von Testergebnissen aus verschiedenen Testplattformen, um eine Gesamtargumentation zu ermöglichen. Die verschiedenen verfügbaren Testplattformen müssen für die Argumentation komplementär genutzt werden. Die Testorchestrierung sorgt für eine optimierte Nutzung der Plattformen basierend auf den relevanten Inputs:

- Testobjekte, Ziele und Szenarien
- Ziel-KPIs und Metriken

- Verfügbare Testplattform

Im Rahmen des Projektes VVM wird basierend auf der State-of-the-art und Anforderungsanalyse (siehe Abschnitt 2.1.5.4) für Testinstanzen eine Taxonomie vorgeschlagen, in der Testansätze anhand von drei Dimensionen unterschieden werden. Details und Inhalte finden sich in der erarbeiteten FZI-Veröffentlichung [55]. Die vorgestellten Testplattformen verwenden einen unterschiedlichen Grad an Virtualisierung und Realismus, was zu einer unterschiedlichen Validität der Ergebnisse führt. Daher ist der Realismus der Testplattform die erste Dimension der vorgestellten Taxonomie. Auf der Grundlage der vorgestellten Untersuchung verwendet die Taxonomie die vier diskreten Klassen virtuell, halb-virtuell, halb-real und real. Die Kriterien für diese Klassen sind von der Klassifizierung von Steimle abgeleitet.

- Die virtuelle Klasse enthält alle Testplattformen, die vollständig simulierte Komponenten und Umgebungen wie SIL oder PIL verwenden.
- Alle Varianten von Hardware-in-the-Loop-Plattformen gehören in die Klasse der semi-virtuellen Testplattformen. Nach Steimle ist für die halbvirtuelle Klasse ein reales Testobjekt erforderlich. Je nach Integrationsgrad der betrachteten HIL-Plattform kann die Lokalisierungssensorik oder das Restfahrzeug emuliert oder real sein.
- Die semi-reale Klasse umfasst alle Plattformen, die ein reales Fahrzeug verwenden, jedoch mit Einschränkungen hinsichtlich der Testumgebung oder der verwendeten Hardware. Das Testen eines realen Fahrzeugs auf einem Testgelände mit künstlich erzeugten Wetterbedingungen und Ballonfahrzeugen ist ein Beispiel für eine halb reale Testplattform.
- Die reale Klasse umfasst den Einsatz eines serienmäßigen oder serienähnlichen realen Fahrzeugs auf öffentlichen Straßen und im realen Verkehr.

Da Emergenzen nur durch die Interaktion verschiedener (Teil-)Systeme entstehen, ist der Integrationsgrad des SUT ein Schlüsselfaktor für die Identifikation von Emergenzen. Darüber hinaus beeinflusst der Integrationsgrad maßgeblich den erforderlichen Aufwand und Umfang des Testens.

- Einzelne Komponente: Detaillierte Betrachtung einer einzelnen Komponente (Sensor, Steuergerät oder Aktor).
- Kompositum von Komponenten: Das SUT besteht aus einem Verbund von verschiedenen Komponenten, die keine vollständige Wirkungskette bilden.
- Wirkungskette: Integration einer kompletten Wirkungskette (z.B. Sensor, ADAS-Funktion und Aktor).
- Vollständiges Fahrzeug: Vollständige Integration aller elektronischen und mechanischen Komponenten.

Die Taxonomie verwendet die Informationsquelle als dritte Klassifizierungsdimension. Der Grundgedanke dahinter ist, dass Tests auf jeder Testplattform und jeder Integrationsstufe auf der Grundlage unterschiedlicher Informationen durchgeführt werden können. Ein reales Fahrzeug kann zum Beispiel nach einem Manöverkatalog auf dem Prüfgelände oder im realen Verkehr getestet werden. Während der Inhalt der Tests auf dem Prüfgelände durch Fahrmanöver vordefiniert ist, ist der Inhalt der Testfahrt zufällig. Dennoch können in beiden Tests die gleichen Szenarien, wie z.B. sicherheitskritische Wendemanöver, auftreten, obwohl der Testinhalt in einem Fall gezielt und im

anderen zufällig generiert wurde. Für die Zwecke seines Beitrags bedeutet der Begriff Informationsquelle die Quelle oder die Methode, wie der Testinhalt erstellt wird. Um diese Definition zu verdeutlichen, werden die folgenden Informationsquellen für die Taxonomie vorgeschlagen:

- Manuell erstellt: Testinhalte (z.B. Testfälle), die manuell aus den Anforderungen abgeleitet werden, Fehlervermutung auf Basis von System-/Erfahrungswissen oder mit Methoden wie Äquivalenzklassen oder Analyse von Grenzwerten.
- Modelle: Durch modellbasierte Ansätze generierte Testinhalte.
- Abgeleitete Szenarien Testinhalte (z.B. logische Szenarien), die von realen Daten abstrahiert wurden.
- Realistisches Fahren: Bei den Testinhalten handelt es sich um uninterpretierte, realistische Fahrinformationen auf öffentlichen Straßen, die entweder aufgezeichnete Daten sind oder bei Testfahrten live erlebt werden können.
- Modifiziertes realistisches Fahren: Bei den Testinhalten handelt es sich um modifizierte realistische Fahrinformationen, die entweder während der Re-Simulation oder der Live-Ergänzung während der Testfahrten aufgezeichnet werden.

Eine beispielhafte Klassifizierung mit den erarbeiteten Methoden der etablierten Ansätze in der Verifikation & Validierung ist in der folgenden Abbildung dargestellt.

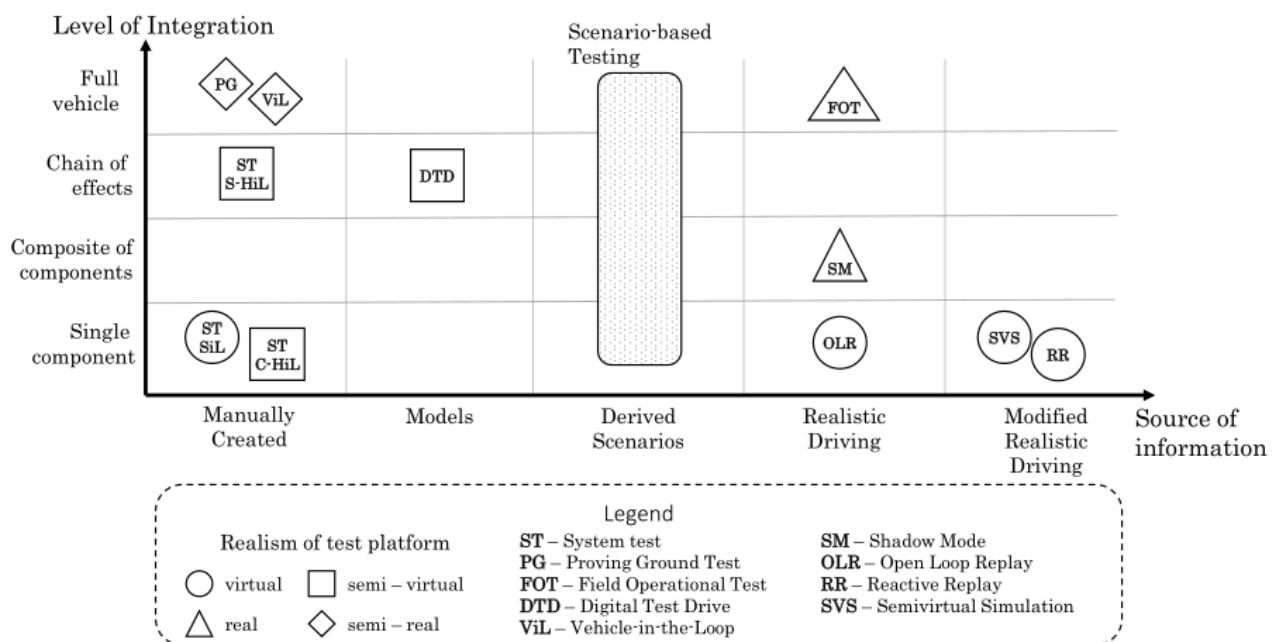


Abbildung 23: Beispielhafte Klassifizierung von Ansätzen zur Verifikation & Validierung [55]

Die vorgestellten Konzepte konzentrieren sich in ihrer Klassifizierung auf bestimmte Aspekte, wie z.B. ausschließlich auf die Testumgebungen oder Methoden zum Testen von ADAS. Für die Verteilung von Tests ist jedoch eine vertiefte Betrachtung der Eigenschaften der Testinstanzen notwendig. Um eine optimale Nutzung zu gewährleisten, ist ein tiefes Wissen über die verfügbaren Testplattformen erforderlich. Dazu gehören verschiedene Eigenschaften und Verhaltensweisen der

Testplattformen, um daraus Stärken und Schwächen abzuleiten. Ein Auszug der Eigenschaften ist in Abbildung 9 dargestellt.



Abbildung 24: Unterschiedliche Eigenschaften von Testplattformen

Um diese Eigenschaften zu analysieren, wurde ein systematischer Vergleich von Testplattformen durchgeführt. Als Input für die Testorchestrierung soll sie eine individuelle Auswahl und Kombination innerhalb der Testorchestrierung ermöglichen. Je nach Grad der Integration in das Gesamtsystem können Testplattformen in reale, semi-reale und simulierte unterteilt werden. Bei virtuellen Testplattformen werden die fehlenden Teile des Gesamtsystems durch simulative Teile abgedeckt (z.B. Restbussimulation des fehlenden Fahrzeugbusses). Die Testplattform beschreibt den grundsätzlichen Aufbau bzw. Ansatz, ein Vergleich verschiedener Testplattformen auf der Grundlage des Integrationsgrades ist in der folgenden Abbildung basierend auf der vorgestellten Taxonomie vorgenommen.

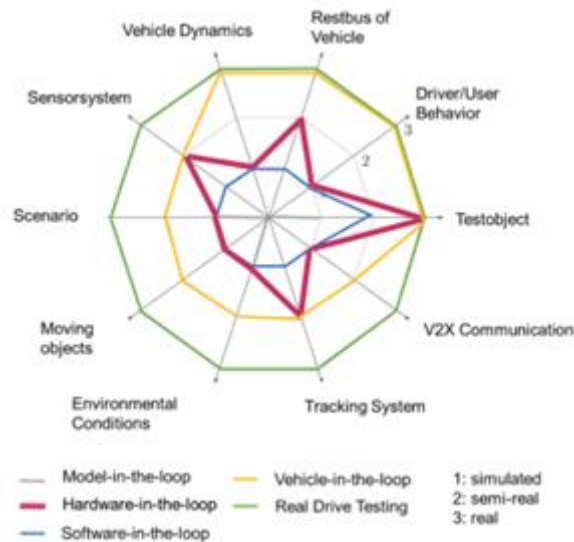


Abbildung 25: Vergleich verschiedener Testplattformen auf der Grundlage des Integrationsgrads

Es ist zu beachten, dass nicht jede Funktionalität des Fahrzeugs auf jeder Testplattform getestet werden kann. Wird eine bestimmte Plattform angesprochen, wie z.B. der Hardware-in-the-Loop-Prüfstand für die Fahrassistenz, so wird diese spezifische Version als Testinstanz bezeichnet. Konkrete Szenarien können auf Testinstanzen ausgeführt werden.

Zudem wurden im Rahmen des Projektes Kriterien für die Evaluation von Testinstanzen erarbeitet, ein Auszug findet sich in der folgenden Tabelle 4. Eine vollständige und detaillierte Beschreibung finden sich in den Deliverables D07 bzw. D12.

Tabelle 4: Auszug an Kriterien für die Evaluation von Testinstanzen

Category	Hierarchy	Description	Example
Description of tests	High Level	Format of test description	OpenScenario
Description of tests	Technical Implementation	Accuracy of measureands	Accuracy of GPS-position, time resolution,...
Parameter Variation	High Level	Adjustable scenario parameters	Speed, acceleration,...
Parameter Variation	Technical Implementation	Parameter range	Speed: 10-50 km/h Acceleration: 1-5 m/s ²
Available ODD Elements	High Level	Available static ODD elements	Level 1-3 from the 5-Level- Model
Available ODD Elements	Technical Implementation	Class of available static ODD elements	Cars, traffic signs, buildings, traffic lights, obstacles,...
Available ODD Elements	Technical Implementation	Road Layout	Plain area, road markings, intersections, roundabouts...
Available ODD Elements	High Level	Classes of dynamic ODD elements	
Available ODD Elements	Technical Implementation	Property level of detail	Reflectivity of clothes, color, specific cars
Available ODD Elements	Technical Implementation	Real behavior of dynamic ODD elements	Bicyclist with leg movement, reflections from target, UFO or cable pull
Available ODD Elements	High Level	Executable maneuvers from test vehicle/steering robots	turn, lane change
Available ODD Elements	High Level	Existing disturbance factors from the test instance	Reflections from rails of the pedestrians
Available ODD Elements	High Level	Testable phenomena	Occlusion of objects, low sun
Scope and Accuracy	Technical Implementation	Accuracy of test execution	Deviation from specified trajectory - Deviation of mean value $\frac{1}{(N)} * \sum (x_i)$ to reference value (if reference value is available) - Std. deviation of the measured values: $\frac{1}{\sqrt{N}} * \sqrt{\sum (x_i - x_{ref})^2}$
Scope and Accuracy	Technical Implementation	Repetition accuracy of test execution	Deviation from specified trajectory for multiple runs - Std. deviation of the measured values from one run to the mean value of all $\frac{1}{\sqrt{N-1}} * \sqrt{\sum (x_i - \langle x \rangle)^2}$
Effort for test execution	Technical Implementation	Financial effort for generation of test cases	

Effort for test execution	Technical Implementation	Time effort for execution of test cases	$T_{\text{Test}} = T_{\text{preparation}} + T_{\text{Execution}} + T_{\text{Postprocessing}}$
Effort for test execution	Technical Implementation	Financial effort for preparation of the test execution	Costs for the setup of the scenery (personal, buildings, other vehicles)
Effort for test execution	Technical Implementation	Time effort for preparation and post processing of the test execution	$T_{\text{Test}} = T_{\text{preparation}} + T_{\text{Execution}} + T_{\text{Postprocessing}}$ Time effort for preparation: bring vehicles in initial position, clear error memory of ECUs, load scenarios into simulation Time effort for post processing: Measurement data storage, restore initial conditions on test instance, ...
Evaluation	High Level	Available measuring equipment	External measurement, reference sensors, in vehicle bus
Evaluation	High Level	Available measureands	Position, speed, acceleration,...
Evaluation	Technical Implementation	Kind of measureands	Vehicle signal, GPS-Position,
Evaluation	Technical Implementation	Accuracy of measureands	Accuracy of GPS position, time resolution,...
Evaluation	High Level	Format of measurements	CAN-Traces, csv, ...
Evaluation	High Level	Available criticality and evaluation metrics	TTC, PET, ...

2.1.6.4 Entwicklung von detaillierten Konzepten für eine durchgängige Testinfrastruktur

Testfelder und Testgebiete mit intelligenter Infrastruktur können eine innovative Lösung bieten, um die Lücke zwischen Laborprüfständen und realen Tests zu schließen. Dies geschieht, indem autonome Fahrzeuge gefahrlos mit kritischen Szenarien konfrontiert werden. Während sich aktuelle Forschungsbemühungen auf die infrastrukturelle Unterstützung des automatisierten und assistierten Fahrens konzentrieren, liegt unser Fokus auf der Integration von realen Testfahrzeugen mit simulierten, virtuellen Hindernissen und Verkehrsteilnehmern mittels intelligenter Infrastruktur auf Testgeländen und in Testgebieten.

Unser vorgeschlagenes innovatives Testsystemprotokoll dient als entscheidendes Bindeglied für die Integration physischer Fahrzeuge mit virtuellen Hindernissen. Wir leiten allgemeine Anforderungen und ein universelles Testsystemprotokoll ab und evaluieren insbesondere die Eignung des drahtlosen Kommunikationsprotokollstapels IEEE 802.11p/ETSI ITS-G5 für diesen Zweck aufgrund der technologischen Entwicklung und Verbreitung.

Das methodische Konzept der Mixed-Reality-Versuchsfelder umfasst experimentelle autonome Fahrzeuge mit dem HAF an Bord (SuT) und die intelligente Infrastruktur als Testsystem. Das Testsystem im Backend oder Kontrollzentrum führt die Testdurchführung durch, verwaltet verschiedene Testphasen und empfängt die Zustände der Versuchsfahrzeuge im Feld. Diese Zustände fließen in eine globale Verhaltenssimulation des Verkehrsszenarios ein, und die Zustände

der virtuellen Verkehrsteilnehmer werden an die SuTs zurückgesendet. Abschließend bewertet das Testsystem das Szenario.

Unser Ansatz sieht ein Testsystem vor, bestehend aus dem straßenseitigen und dem Backend-Testsystem sowie einem fahrzeugseitigen Testsystem-ECU (OTS), siehe Abbildung 26. Das Testsystem umfasst zwei drahtlos verbundene Steuereinheiten: die Backend-Roadside-Unit (RSU) und eine spezielle Onboard-Unit (OBU). Die OBU stellt eine direkte Schnittstelle zum OTS dar, das die Sensorsimulation und Lokalisierung übernimmt. Das OTS wiederum bildet eine direkte Schnittstelle zur automatisierten Fahreinheit (ADU) mit dem HAF. Die Simulation der virtuellen Verkehrsteilnehmer ist in Verhaltensmodell und Sensorsimulation aufgeteilt.

Die methodische Konzeption des Mixed-Reality-Testgeländes stellt spezifische Anforderungen an Fahrzeuglokalisierung, verkehrsbeeinflussende Elemente, Kommunikation und Backend-Infrastruktur. Beispielsweise kann die Lokalisierung durch externe Sensoren der intelligenten Infrastruktur oder internes GPS/INS-Sensorsystem erfolgen, wobei hohe Genauigkeit für die Verwendung als Ground Truth erforderlich ist.

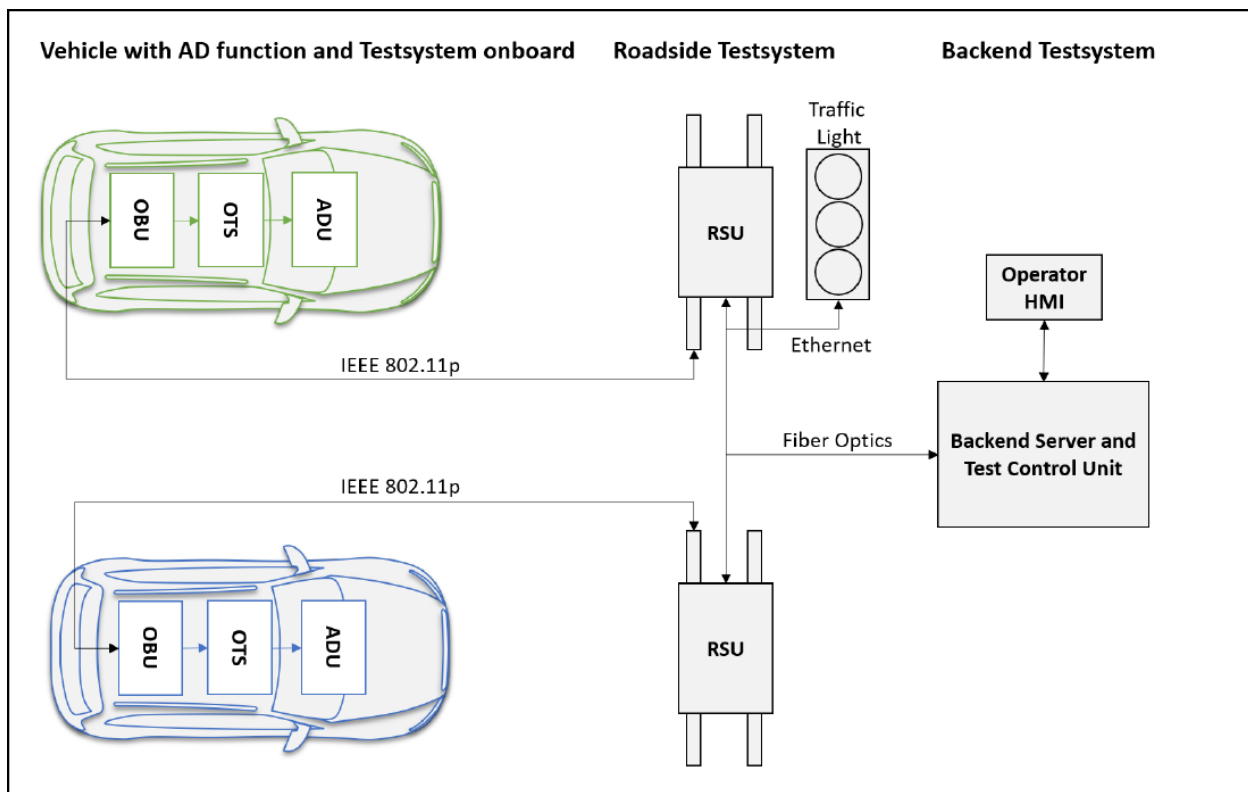


Abbildung 26: Gesamtarchitektur des Testsystems für die Integration mehrerer Fahrzeuge.

Um den Kommunikationsaufwand zwischen dem fahrzeugseitigen Testsystem und dem Testsystem während der Testdurchführung zu minimieren, schlagen wir vor, Informationen a priori zu laden. Dabei gehen wir davon aus, dass sowohl das Testsystem als auch die OTU Zugriff zu einer gemeinsamen Szenario-Datenbank haben, die Szenarien, hochauflösende Karten und Szenario-Parametersätze enthält.

Die Ergebnisse und die detaillierte Umsetzung wurden im Paper mit dem Titel "Paving the Ways for Vehicles-in-the-loop: Wireless Integration of Autonomous Vehicles into Mixed Reality Proving Grounds and Test Areas" [56] veröffentlicht.

2.1.6.5 Entwicklung eines durchgängigen Verfahrens zur Abbildung von Szenarien

Im Rahmen dieser Arbeit wurde ein Framework entwickelt, das die Darstellung von Szenen ermöglicht, die als Initialisierung für ein Testszenario für eine hochautomatisierte Fahrfunktion (HAF) verwendet werden können. Jede Aktion und jedes Verhalten eines Verkehrsteilnehmers innerhalb dieser Szene können den Ausgang des Szenarios verändern. Daher wurden verschiedene Zukünfte (Kind-Szenarien) simuliert, um ein breiteres Verständnis der Szene zu erhalten. In den Experimenten wurden Verhaltensmodelle verwendet, die so gewählt wurden, dass sie möglichst viele Interaktionen realitätsgetreu wiedergeben. Die Qualität der gewählten Verhaltensmodelle wirkt sich jedoch direkt auf die Ergebnisse aus. In einigen Experimenten verließ der vom hybriden Imitationsverhaltensmodell gesteuerte Verkehrsteilnehmer die Fahrspur und fand nicht mehr zurück auf die Straße. Dieses Phänomen ist ein Beispiel dafür, dass die durchgeführten Simulationen keine endgültige Beurteilung einer Verkehrsszene liefern können. Wir glauben jedoch, dass wir durch die Darstellung verschiedener Eventualitäten, wenn auch nur einer Teilmenge aller möglichen Zukünfte, unser Verständnis der Situation deutlich verbessern können. Darüber hinaus haben wir in unseren Experimenten das Ergebnis aller Kind-Szenarien aus einer Ausgangsszene hinsichtlich ihrer Kritikalität mit verschiedenen Kritikalitätsmetriken bewertet. Darüber hinaus haben wir gezeigt, dass unterschiedliche Seed-Szenen zu unterschiedlichen Kritikalitätsmerkmalen führen können (z. B. hat das Zusammenführen eine höhere Kritikalität in Bezug auf die Entfernung).

Es wurde ein Paper mit dem Titel "From Model-Based to Data-Driven Simulation: Challenges and Trends in Autonomous Driving" [57] veröffentlicht, welches die Konzepte detailliert aufzeigt. Das Paper präsentiert zum einen die Entwicklung eines Klassifikationsschemas für verschiedene Simulationsansätze. Zum anderen werden die Herausforderungen im Zusammenhang mit den verschiedenen Komponenten von Simulationsumgebungen vorgestellt und die in der Industrie und im akademischen Bereich beobachteten Trends zu ihrer Überwindung beschrieben. Dabei geht es vor allem um Methoden, Simulationen möglichst realitätsnah zu gestalten. Dies umfasst unter anderem die Modellierung der Umgebung, die Verhaltensmodelle, welche zur Simulation von Szenarien benötigt werden, sowie die Simulation von Sensordaten.

Darüber hinaus wurde eine weitere Veröffentlichung mit dem Titel "Cyclist behavior: a survey on characteristics and trajectory modeling" [58] erarbeitet. Das Paper befasst sich mit Methoden zur Modellierung von Fahrradfahrern in Simulation und typischem Verhalten von Fahrradfahrern im Straßenverkehr. Daraus können wichtige Szenarien, die Radfahrende beinhalten, abgeleitet werden.

2.1.6.6 Szenen Extrapolation: Generierung interaktiver Verkehrsszenarien

Ein Problem bei wissensbasierten Szenariogenerierungsansätzen besteht in der Szenarioexploration. Hierbei wird der Szenarienraum systematisch durchsucht, indem man einzelne, wenige Parameter verändert, um relevante Szenarien zu finden. Die datengestützte, alternative Methode stützt sich auf bekannte Szenarien aus Unfalldatenbanken oder dem realen Straßenverkehr. Im Gegensatz zu wissensbasierten Methoden bietet der datengesteuerte Ansatz mehr Realitätsnähe, da er die mit Expertenmodellen verbundenen Einschränkungen und Verzerrungen vermeidet.

In diesem Beitrag, der im Rahmen des Projekts bearbeitet wurde, wird ein Ansatz zur Generierung von Testszenarien vorgeschlagen, der sich von herkömmlichen Methoden unterscheidet und die Unabhängigkeit von der zu testenden Fahrfunktion beibehält. Anstatt sich auf definierte Szenarien

zu verlassen, verwenden wir Seed-Szenarien. Die Seed-Szenarien geben nur den Startpunkt für jedes Szenario vor. Der tatsächliche Verlauf des Szenarios wird durch jede einzelne zu testende Fahrfunktion und die vom Versuchsingenieur ausgewählten Verhaltensprofile der anderen Verkehrsteilnehmer bestimmt. In einer Closed-Loop-Simulation, die auf den Verhaltensmodellen der beteiligten Akteure basiert, extrapolieren wir die Szenarien in verschiedene Zukünfte, um ein allgemeines Verständnis der Ausgangssituation zu erhalten. Eine umfassende Bewertung aller simulierten Zukünfte sollte dann ein Maß für die Relevanz einer Szene für das Testen von Hochautomatisierten Fahrzeugen liefern.

Die Ergebnisse und die detaillierte Umsetzung der Arbeit wurden in der FZI-Veröffentlichung mit dem Titel "Scene-Extrapolation: Generating Interactive Traffic Scenarios" [59] aufgezeigt.

Simulationsumgebung

Der Simulationsrahmen dient der Bewertung von Szenen für hochautomatisiertes Fahren. Eine Seed-Szene (Ausgangsbedingungen einer Simulation, Position, Geschwindigkeit, Klassen von Verkehrsteilnehmern), basierend auf realen oder synthetischen Daten, bildet den Ausgangspunkt. Sie enthält eine Straßenkarte sowie Informationen zu Position, Orientierung und Geschwindigkeit der Verkehrsteilnehmer. In der Szenenextrapolation wird diese Seed-Szene verwendet, um mögliche zukünftige Szenarien (Kind-Szenarien) zu simulieren. Dabei werden den Akteuren, die in einem geschlossenen Regelkreis mit ihrer Umgebung interagieren, nach dem Zufallsprinzip realistische Verhaltensmuster zugewiesen. Jedes Zukunftsszenario wird mit Kritikalitätsmetriken bewertet, um die Relevanz der Seed-Szene für die Prüfung hochautomatisierter Fahrzeuge zu bestimmen.

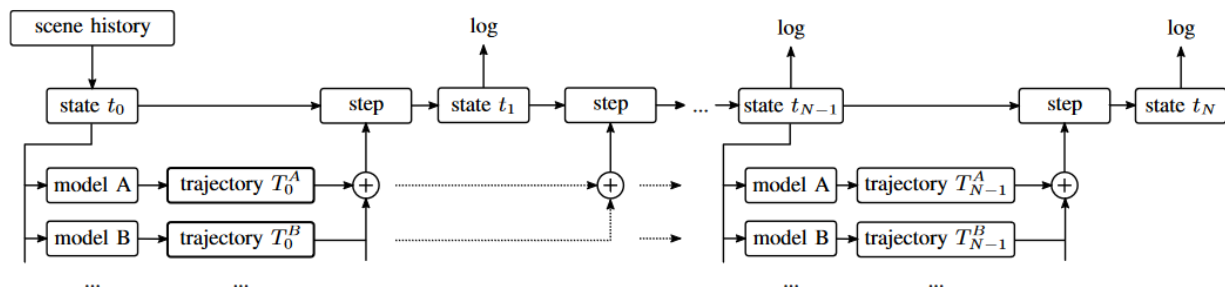


Abbildung 27: Blockdiagramm zur Veranschaulichung des Simulationsprozesses. Die Vorschläge für die Trajektorien werden aus einer gesehenen Szene berechnet, wobei verschiedene Verhaltensmodelle auf der Grundlage der aktuellen Situation berücksichtigt werden.

Der Simulator dient als Ausführungseinheit, die eine Eingabeszene an verschiedene Verhaltensmodelle weitergibt und die berechneten Trajektorien nach jedem Schritt in den nächsten Zustand aktualisiert. Die Schnittstellen zwischen den Modulen sind standardisiert und ermöglichen den beliebigen Austausch von Modulen. Andere Simulatoren, wie die in Abschnitt II vorgestellten, können ebenfalls verwendet werden, wenn zusätzliche Aspekte der Umgebung simuliert werden müssen. Wenn physikalische Berechnungen erforderlich sind, kann CARLA anstelle des für diese Arbeit implementierten Simulationsrahmens verwendet werden (siehe unten unter Aufbau einer Referenzimplementierung der Testinfrastruktur). Dieser Teil der Arbeit konzentriert sich ausschließlich auf das Verhalten und die daraus resultierenden Trajektorien, die je nach Modell nur mit Hilfe von Objektlisten und Straßenkarten berechnet werden können. Daher kann dieser Simulator leicht implementiert werden. Abbildung 27 zeigt die Abfolge der Simulationsschritte. Zu Beginn wird eine Verkehrsszene ausgewählt, die extrapoliert werden soll. Anschließend werden jedem

Verkehrsteilnehmer in der Seed-Szene Verhaltensmuster zugewiesen. Dies kann manuell oder zufällig über eine Verteilungsfunktion erfolgen. Der Simulator versorgt die einzelnen Modelle mit Umgebungs- und Kontextinformationen. Entsprechend der aktuellen Forschung auf dem Gebiet der Trajektorienvorhersage enthält er auch eine Historie der letzten Sekunde. Dies ermöglicht eine einfache Integration zusätzlicher Modelle für das Imitationslernen.

Der Simulator stellt für jeden Schritt der Trajektorienberechnung die gleiche Umgebung zur Verfügung, sodass die Modelle parallel berechnet werden können. Diese Umgebung enthält Position, Abmessungen, Geschwindigkeit und Klassifizierung (Typ) aller Verkehrsteilnehmer für den aktuellen und die 10 vorhergehenden Zeitschritte. Zusätzlich wird eine hochauflösende Straßenkarte mit Straßengeometrie und -topologie bereitgestellt. Mit diesen Kontextinformationen berechnen die Modelle eine Trajektorie für die zugeordneten Verkehrsteilnehmer. Die Modelle berechnen nicht nur den nächsten Zustand des eigenen Fahrzeugs, sondern eine geplante Trajektorie für mindestens die nächsten 30 Schritte. Die Simulationsfrequenz basiert auf dem INTERACTION-Datensatz und beträgt 10 Hz. Um die Simulationsleistung zu verbessern, kann die Trajektorienberechnung für eine parametrierbare Anzahl von Schritten übersprungen werden, sodass die Akteure für einen kurzen Zeitraum den zuvor berechneten Trajektorien folgen können. Die Simulation liefert die besten Ergebnisse, wenn die Trajektorien alle fünf bis zehn Schritte berechnet werden. Die Simulationsdauer für ein Szenario beträgt im Mittel 0,5 s und hängt stark von der Wahl der Verhaltensmodelle ab, wobei die Inferenz der KI-Modelle der begrenzende Faktor ist. Nach jedem Schritt werden die Zustände aller Akteure konsolidiert und im Logger gespeichert. Das Datenformat für die Speicherung der Szenarien folgt den Spezifikationen des INTERACTION-Datensatzes, was eine nachhaltige Wiederverwendbarkeit und die Nutzung anderer Toolkits gewährleistet.

Verhaltensmodelle

Verhaltensmodelle werden verwendet, um den einzelnen Akteuren unterschiedliche Fahrerprofile zuzuordnen. Jedes Modell nutzt die vom Simulator bereitgestellten Umgebungs- und Kontextinformationen, um eine zukünftige Trajektorie zu berechnen. Die Wahrnehmungsfunktion modifiziert die globalen Umgebungsdaten, indem sie die Perspektive des Modells auf einen bestimmten Wahrnehmungsbereich einschränkt und die Verkehrsszene in ein lokales Koordinatensystem transformiert. Zusätzlich wandelt eine nachgelagerte Funktion die Liste der Objekte und die HD-Karte aus dem Simulator in die für das Modell erforderliche Darstellung um. Folgende Modelle (siehe Tabelle 5) wurden für die Simulation verwendet:

Tabelle 5: Modelle die in der Simulation genutzt werden

Name	Parametrization
Standard Driver (III-D)	$b = 3.0, T = 3.1, s_0 = 9, \delta = 4$
Risky Driver (III-D)	$b = 8.0, T = 2.1, s_0 = 5, \delta = 4$
Constant Velocity (III-D)	Follows path with initial speed
Emergency Brake (III-D)	Decelerates abruptly at $-5.0 \frac{m}{s^2}$
Graph Predictor (III-D)	Acceleration imitation model
Graph Image Predictor (III-I)	Trajectory imitation model

Details zur Implementierung und der Ergebnisse des Hybriden Modells sind im Paper „Utilizing Hybrid Trajectory Prediction Models to Recognize Highly Interactive Traffic Scenarios“ [60] vorgestellt, welches ebenfalls im Rahmen des Projektes entwickelt wurde.

Experimente

Abbildung 28 zeigt die Dichteverteilungen der analysierten Metriken für zwei Ausgangsszenarien in blau und rot. Das blaue Szenario wurde einem Kreisverkehr entnommen, während das rote Szenario von einer Autobahnauffahrt mit einer Einmündungssituation stammt. Die Ground Truth, die den tatsächlichen Verlauf des Szenarios, wie er im Datensatz aufgezeichnet wurde, darstellt, ist durch eine gestrichelte Linie gekennzeichnet. Die GapTime ist im Einmündungsszenario (rot) nicht anwendbar, da sie eine Metrik für Kreuzungstrajektorien ist. In diesem Szenario gibt es nur die folgenden Fahrzeugkonstellationen.

Es ist erwähnenswert, dass die Dichteverteilungen der Kritikalitätsverteilungen selten einer Normalverteilung entsprechen. Stattdessen weisen sie in Abhängigkeit von der Seed-Szene unterschiedliche Merkmale auf. Außerdem ist es offensichtlich, dass die Kritikalitätswerte der Bodenwahrheit nicht immer repräsentativ für die Seed-Szene sind. So können beispielsweise für das hier dargestellte Ground-Truth-Szenario keine Kritikalitätsmetriken berechnet werden, wohl aber für einige Kind-Szenarien. Außerdem fällt der Wert der Ground-Truth nicht unbedingt auf einen bestimmten Punkt in der Verteilung, woraus wir schließen könnten, dass die Simulation dazu neigt, mehr oder weniger kritische Szenarien zu erzeugen, als in der Realität auftreten würden.

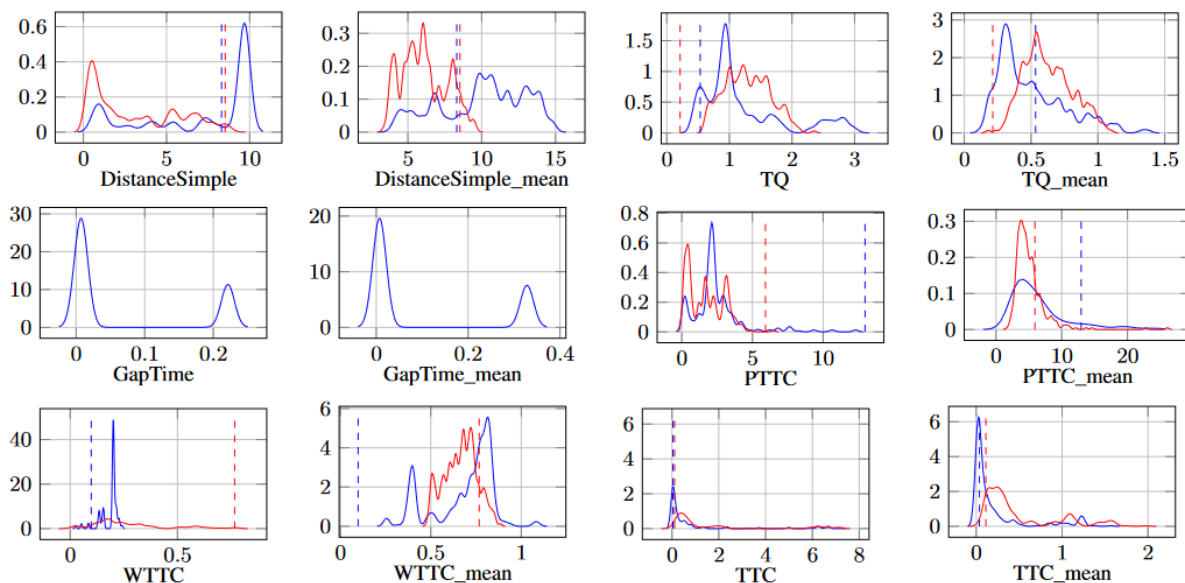


Abbildung 28: Die Abszisse ist die Kritikalität und die Ordinate die Dichte. Die blauen und roten Farben zeigen die Verteilung der Kritikalität für alle Child-simulationen an.

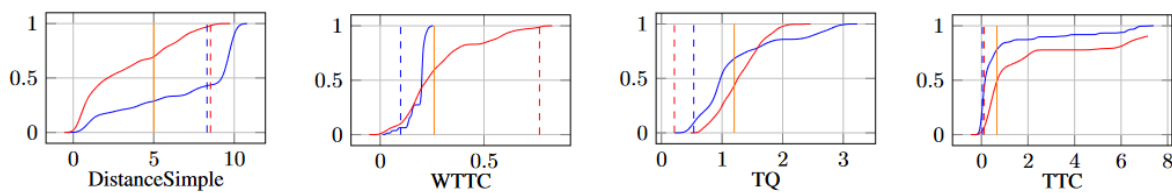


Abbildung 29: Das Diagramm zeigt die kumulative Dichte für Kritikalitätsmetriken für 2 Seed-Szenen (rot und blau)

Abbildung 29 beschreibt die kumulative Dichte für ausgewählte Metriken der beiden Seed-Szenen aus Abbildung 28. Die orangefarbene Linie zeigt die in der Literatur vorgeschlagene Kritikalitätsschwelle für alle Metriken an. Im Falle von Distance und WTTC sind alle Simulationen auf der linken Seite der Linie kritische Simulationen. Für die Verkehrsqualität und die invertierte TTC sind alle Simulationen auf der rechten Seite kritisch. Wir haben folgende Schwellenwerte verwendet: Abstand 5 m, WTTC 0,26 s (physikalische Grenze zum Bremsen), Verkehrsqualität 1,2, TTC 1,5 s. In Bezug auf den Abstand ist das rote Szenario die kritischere Ausgangsszene, da die Kurve schneller wächst als die blaue. Dies kann jedoch durch Unzulänglichkeiten des Abstandsmaßes erklärt werden. Autofahrer fahren beim Zusammenführen oft nebeneinander auf verschiedenen Fahrspuren. Dies führt zu kleinen Abständen und damit zu einer hohen Kritikalität. Die TTC-Metrik für die rote Kurve zeigt ebenfalls eine kritischere Szene als die blaue Kurve. Dies liegt wiederum daran, dass TTC nur für Auto-Folgekonstellationen berechnet werden kann und die rote Seed-Szene nur aus Auto-Folgekombinationen besteht.

Für WTTC zeigt die blaue Kurve in Abbildung 5 und Abbildung 6 eine etwas kritischere Seed-Szene. Die blaue Kurve erreicht den Wert 1 vor dem Schwellenwert, was bedeutet, dass alle simulierten Kind-szenarien einen niedrigeren WTTC-Wert als den betrachteten Schwellenwert haben. Anders als die klassische TTC kann die WTTC in jeder beliebigen Konstellation berechnet werden und kann daher kritische Situationen finden, die die TTC übersehen würde. Darüber hinaus neigt der WTTC dazu, mehr Situationen als kritisch einzustufen, als tatsächlich vorhanden sind. Im Allgemeinen sind Seed-Szenen mit kritischen Kind-Szenarien auf der linken Seite des Schwellenwerts von größerer Bedeutung, da die Kurve steiler wird. Umgekehrt sind Seed-Szenen mit kritischen Kind-Szenarien auf der rechten Seite bedeutsamer, wenn die Kurve auf der rechten Seite zu wachsen beginnt.

2.1.6.7 Umgebungsgenerierung und Variation statischer Elemente

Zur Erstellung von Variationen für statische Umgebungsobjekte wurde eine Methode für die Extraktion und Generierung von Elementen auf Basis einer OpenSceneGraph Definition erarbeitet umgesetzt. Dazu werden Elemente aus dem OpenSceneGraph importiert und abhängig von ihren Objektdimensionen gefiltert. Dieser Schritt ist nötig, da im Standard zu OpenSceneGraph keine Klassen oder Typen beschrieben sind. Eine händische Auswahl und Zuordnung ist ebenfalls möglich. Anhand eines in einem Zwischenschritt generierten Objektgrundrisses, wird prozedural für jedes Element ein äquivalentes Objekt in der Simulation erzeugt. Die so entstandenen Elemente beinhalten verschiedene Parameter und können danach weiter variiert werden, z.B. Änderung der Größe, Position oder Oberfläche. Für die Erprobung der verschiedenen Variationsansätze für virtuelle Umgebungen wurde nach realen Straßenzügen in Karlsruhe gesucht, die einen Bezug zum FUC 2.3 aufweisen. Dabei wurde die Kreuzung Ostendstraße / Frühlingstraße identifiziert, da hier drei Straßen ohne zusätzliche Vorfahrtregelung sowie ein Zebrastreifen vorhanden sind.

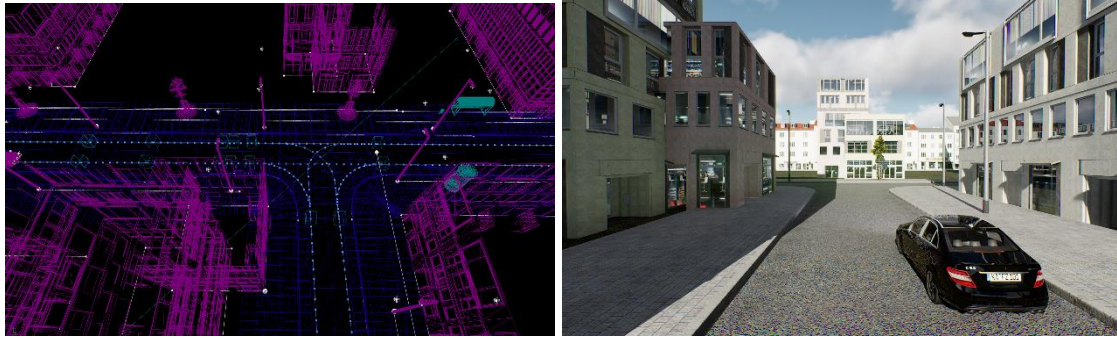


Abbildung 30: Extrahierte und ersetzte Objekte in FUC 2.3

Des Weiteren wurde ein parametrisierbares Oberflächenmodell für Asphaltstraßen implementiert. Das Straßenoberflächenmaterial ist somit konfigurierbar, insbesondere in den Eigenschaften zu Asphalzzusammensetzung bzw. Mischverhältnis, Alter und Verkehrsaufkommen. Durch eigenständiges UV-Mapping während der Generation der 3D Oberfläche aus OpenDRIVE Daten, können Abnutzungsspuren wie Spurrillen oder Fahrbahnverschmutzungen dem Straßenverlauf angepasst werden. Zukünftig soll noch die Kompatibilität mit OpenMATERIAL fertig gestellt werden.



Abbildung 31: Digitaler Zwilling in Karlsruhe für FUC 2.3

Im Rahmen der Vorbereitungsarbeiten für das Abschlussevent wurde die Simulationsumgebung am FZI um zusätzliche Schnittstellen erweitert, die eine Co-Simulation mit dem Schwingungsprüfstand des Fraunhofer Institut ermöglichen. Für die Co-Simulation wurde als Szenario der FUC 2.3 gewählt, allerdings in Form eines digitalen Zwillings einer Kreuzung in Karlsruhe. Die Arbeiten an der Umgebungsvariation wurden abgeschlossen und die entwickelten Werkzeuge zur Variation an verschiedenen Szenarien getestet. Für das Abschlussevent wurden unterschiedliche Umgebungsvariationen am FUC 2.3 exemplarisch gezeigt und deren Einfluss auf verschiedene Sensoren demonstriert werden. Für die Co-Simulation mit dem Schwingungsprüfstand (siehe Abschnitt 2.1.7.4) sind die Variationen auf die Beschaffenheit der Straße beschränkt (Fokus Unebenheiten). Die Ergebnisse der Arbeiten zur automatischen Umgebungsgenerierung wurden in der FZI-Veröffentlichung „Fast Environment Generation Methods for Virtual Testing“ [61] präsentiert.

2.1.6.8 Selektionsprozess zur Ermittlung der Aussagekraft eines logischen Szenarios im Absicherungs- und Freigabeprozess einer hochautomatisierten Fahrfunktion

Die rein kombinatorische Erkundung des Parameterraums führt zu einer nicht ausführbaren Menge von Testfällen. Um die Sicherheitsprüfung effizient durchzuführen, wird ein szenario-basierter Testansatz verwendet, der eine solide Begründung für den Auswahlprozess der Testfälle benötigt.

Ziel dieses Ansatzes ist es, relevante Szenarien auszuwählen, die repräsentativ für den gesamten Parameterraum sind und somit die Testaufwände zu reduzieren, während gleichzeitig aussagekräftige Ergebnisse erzielt werden. Es wird davon ausgegangen, dass Szenarien gefunden werden können, die repräsentativ für andere Szenarien sind.



Ziel: Selektion von relevanten Szenarien, welche den gesamten Parameterraum hinreichend repräsentativ abbilden

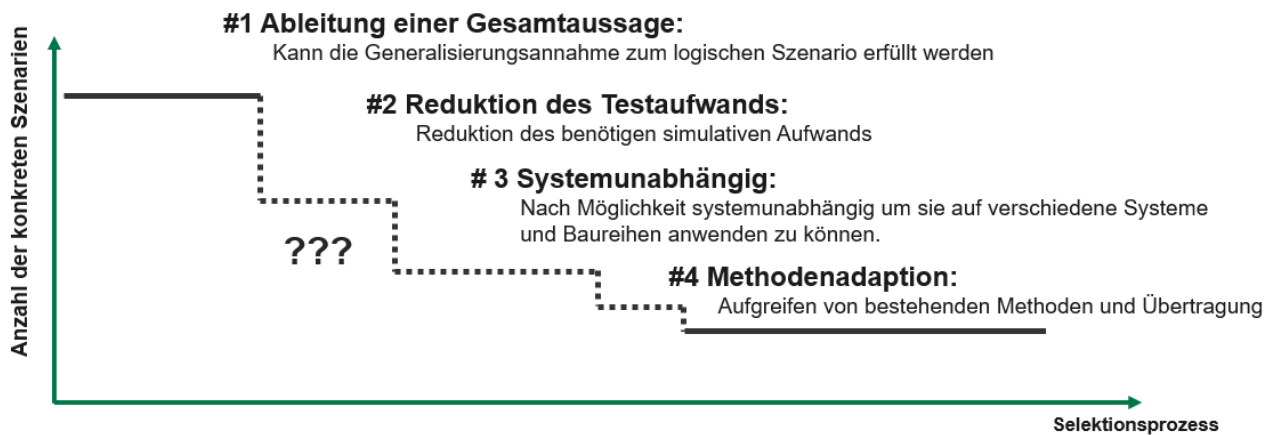


Abbildung 32: Anforderungen und Konzeptbild für die hinreichende Reduktion von Szenarien

Der Selektionsprozess beginnt mit der Erstellung einer vollständigen Liste der möglichen Parameterkombinationen, ohne diese auszuführen. Hierbei kommen Methoden wie die lineare Kombinatorik und die Zufalls-Kombinatorik zum Einsatz. Im nächsten Schritt werden Szenarien mit unlogischen oder physikalisch unmöglichen Kombinationen reduziert, um die Menge der zu simulierenden Testfälle zu verringern. Dies erfolgt durch die Kombination von Parametern basierend auf physikalischen und mathematischen Beziehungen sowie durch eine kombinatorische Reduzierung und Kritikalitätsanalyse basierend auf Exposition und Schweregrad.

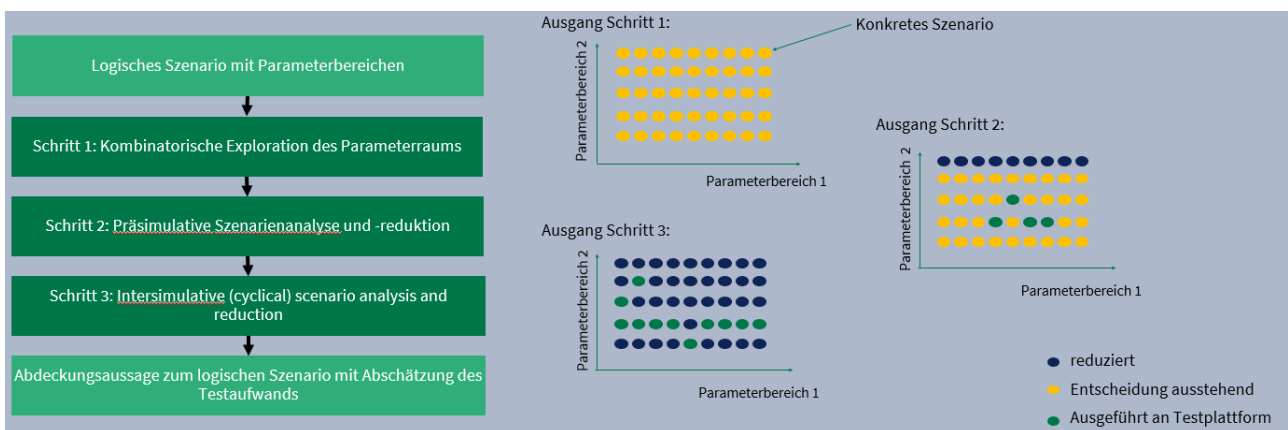


Abbildung 33: Schritte des Selektionsprozess zur Auswahl von repräsentativen konkreten Szenarien für ein logisches Szenario

Im dritten Schritt werden die verbleibenden Szenarien im Systemkontext durchgespielt, und die dabei gewonnenen Messergebnisse werden zur weiteren Reduktion der Szenarien verwendet. Methoden wie die Ähnlichkeitsanalyse basierend auf Key Performance Indicators (KPIs), Zeitreihen oder Parametern helfen dabei, die Szenarien weiter einzugrenzen.

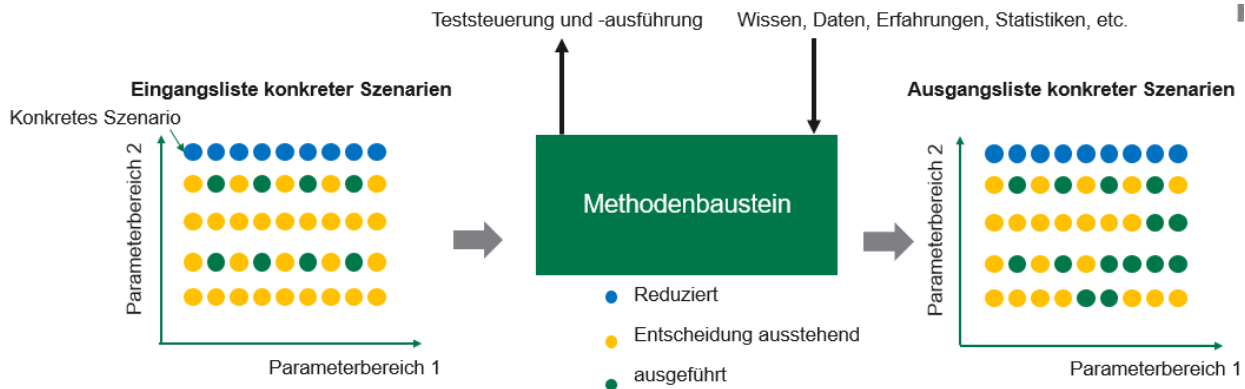


Abbildung 34: Aufbau eines Methodenbausteins im Selektionsprozess

Ein wesentlicher Bestandteil des Selektionsprozesses sind die Methodenbausteine. Ein Methodenbaustein ist eine in sich geschlossene Vorgehensweise, die eine Liste konkreter Szenarien innerhalb eines logischen Szenarios als Eingangsgröße verwendet. Diese Liste kann bereits teilweise ausgeführt und reduziert sein. Methodenbausteine sind kaskadierbar und austauschbar, wodurch der Selektionsprozess flexibel und anpassbar bleibt.

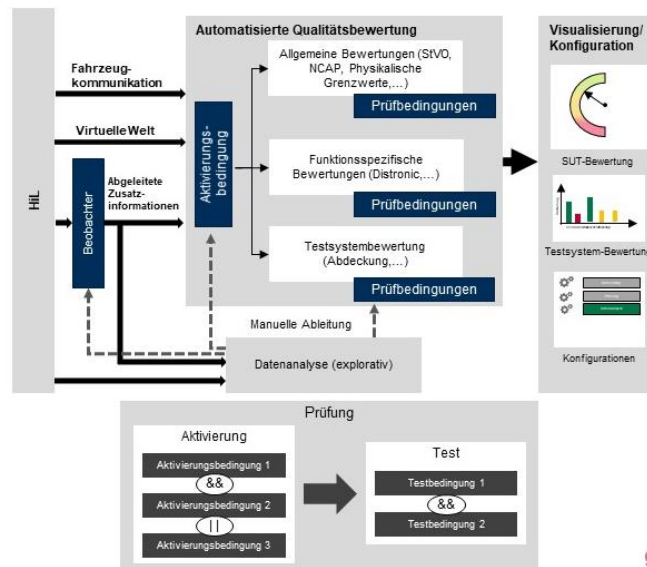
Der beschriebene Ansatz hat mehrere Vorteile. Durch die gezielte Auswahl und Reduzierung der Testfälle wird der erforderliche Simulationsaufwand signifikant verringert. Der Ansatz ist zudem systemunabhängig und kann auf verschiedene Systeme und Serien angewendet werden. Trotz der Reduktion der Testfälle bleibt die Gesamtaussage über die Sicherheit und Leistungsfähigkeit des Systems erhalten. Insgesamt ermöglicht dieser strukturierte und begründete Selektionsprozess eine effiziente und effektive Sicherheitsprüfung automatisierter Fahrsysteme.

2.1.6.9 Testkonzept zur kontinuierlichen Bewertung von Fahrsystemen

Die Einführung eines systematischen und realistischen Erprobungssystems in der Fahrzeugentwicklung ist essenziell, um die Komplexität moderner Fahrsysteme und deren Interaktion mit der realen Welt adäquat zu testen. Traditionelle, fest definierte Testfälle stoßen an ihre Grenzen, da sie nicht die Vielfalt und Unvorhersehbarkeit realer Verkehrssituationen abbilden können. Durch die Vorgabe einer Route anstelle starrer Testfälle und die Erzeugung realistischer Erprobungsfahrten innerhalb einer Simulation kann eine umfassendere und realitätsnähere Prüfung der Systeme gewährleistet werden.

Realistische Erprobungsfahrten und Simulationsmodelle

Die Verwendung von realen Fahrer- und Verkehrsmodellen in der Simulation erlaubt es, unbekannte und potenziell kritische Situationen künstlich zu erzeugen. Dies ist besonders wichtig, um sicherzustellen, dass die entwickelten Systeme nicht nur unter idealen Bedingungen funktionieren, sondern auch in komplexen, unvorhergesehenen Verkehrsszenarien.



9

Abbildung 35: Konzept der digitalen Erprobung mit kontinuierlichem Bewertungskonzept

Kontinuierliche Prüfungen

Ein Übergang von spezifikationsbasierten Tests hin zu kontinuierlichen Prüfungen ermöglicht eine dynamischere und flexiblere Erprobung. Mehrere Prüfungen können gleichzeitig ausgeführt werden, was die Effizienz des Testprozesses erheblich steigert. Unterschiedliche Prüfkriterien und -typen, wie physikalische, strukturelle, regulatorische, normative sowie funktions- und applikationsbezogene Kriterien, können parallel in einem Szenario getestet werden.

Physikalische und strukturelle Kriterien

- Freiraum, Fahrschlauch: Überprüfung der Einhaltung von Sicherheitsabständen und korrekter Spurführung.
- Beschleunigung und Lenkmomente: Analyse der Fahrzeugdynamik und der maximalen Leistungsfähigkeit.

Regulatorische und normative Kriterien

- Standards (StVO, EURO-NCAP, TÜV, DEKRA): Sicherstellung der Einhaltung gesetzlicher Vorschriften und Industriestandards.
- Funktions- und applikationsbezogene, weiche Kriterien
- Komfort, Effizienz, Sicherheit: Beurteilung der Fahrerezufriedenheit und Betriebseffizienz.
- Längs- und Querschleunigung, Ruck: Analyse des Fahrkomforts und der Fahrzeugsicherheit.
- Systematische und reproduzierbare Ausführung und Dokumentation

Die systematische und reproduzierbare Ausführung von Prüfungen sowie deren Dokumentation sind entscheidend für die Nachvollziehbarkeit und Validierung der Testergebnisse. Event-basierte

Prüfungen ermöglichen detaillierte Systembewertungen auf Verhaltensebene durch einen Beobachteransatz. Dieser erlaubt eine reine Systembewertung und ist besonders wertvoll für a-priori unbekannte Szenarien. Zudem können mehrere Systeme parallel in einem Szenario bewertet werden, was die Effizienz und Aussagekraft der Tests weiter erhöht.

Kontinuierliche Anwendung im Entwicklungsprozess

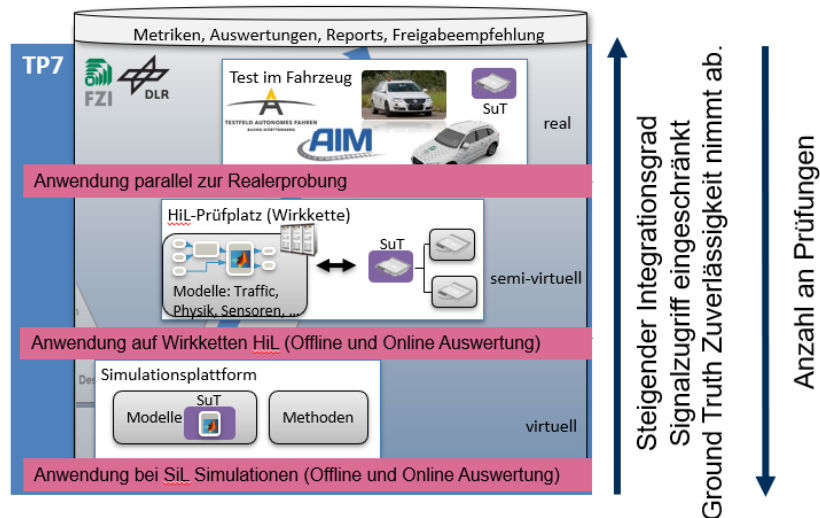


Abbildung 36: Anwendung im Entwicklungsprozess

Die entwickelten Prüfungen und Bewertungen sind während des gesamten Entwicklungs- und Absicherungsprozesses einsetzbar. Messsignale und Ground-Truth-Informationen stehen dabei auf den jeweiligen Testinstanzen zur Verfügung und ermöglichen eine präzise und detaillierte Analyse. Die Bewertung kann sowohl online während der Testfahrten als auch im Nachgang auf Basis der aufgezeichneten Messungen erfolgen. Dies gewährleistet eine kontinuierliche Überwachung und Verbesserung der Systeme, von den frühen Entwicklungsstadien bis hin zur finalen Absicherung.

Durch diese ganzheitliche Herangehensweise wird sichergestellt, dass die entwickelten Fahrsysteme nicht nur funktional und effizient, sondern auch sicher und regelkonform sind, was letztlich zu einer höheren Zufriedenheit der Endnutzer und einem höheren Maß an Verkehrssicherheit führt.

2.1.6.10 Methode zur Bewertung von Testinstanzen & Testorchestrierung (Seamless Testing)

Im Rahmen des Projekts wurde ein nahtloser Testprozess für urbane Szenarien entwickelt und implementiert, der sowohl Simulationen als auch Testgelände umfasst. Dabei kamen standardisierte Formate und Schnittstellen wie OSI, FMU, OpenScenario und OpenDrive zum Einsatz, um die Austauschbarkeit und Wiederverwendbarkeit von Simulationsmodellen sicherzustellen. Dies ermöglichte vergleichbare und aggregierbare Testergebnisse sowie die Schaffung von Testinstanzen, die sich gegenseitig ergänzen.

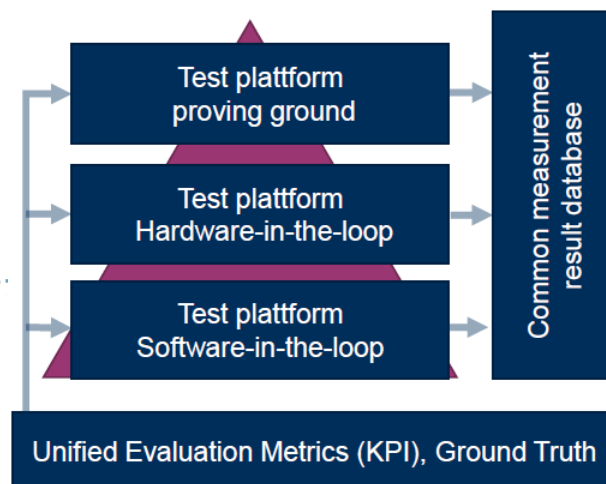


Abbildung 37: Gemeinsame Datenbasis und Evaluationsmetriken über verschiedene Testinstanzen

Es wurde ein durchgängiger Testprozess („Seamless“) entwickelt, der die Konsistenz von Testressourcen entlang der gesamten Validierungskette gewährleistet und die Verteilung von Tests auf verschiedene Testinstanzen wie Simulationen und Testgelände ermöglicht. Durch die Entwicklung standardisierter Testbeschreibungen wurde die Übertragbarkeit der Tests unterstützt.

Ein prototypischer Implementierungsansatz wurde erfolgreich demonstriert, bei dem ein logisches Szenario sowohl in der Simulation als auch auf einem Testgelände ausgeführt wurde. Dabei wurde dieselbe Parametrisierung und Testbeschreibung für beide Testinstanzen verwendet. Die Ergebnisse dieser Tests wurden erfolgreich analysiert und verglichen. Das betrachtete logische Szenario beinhaltete das Geradeausfahren über eine T-Kreuzung mit einem Fußgängerüberweg und geparkten Fahrzeugen am Straßenrand, wobei parkende Fahrzeuge für Sichtbehinderungen sorgten (FUC23).

Die Herausforderungen bei der gemeinsamen Bewertung der Prototyp-Umsetzung, wie die Zeitsynchronisation der Messdaten und die Transformation der Koordinatensysteme der Messdaten, wurden erfolgreich bewältigt. Die Anforderungen an die Austauschbarkeit der Testplattform wurden erfüllt, einschließlich der Validierung der Testumgebung für Vergleichbarkeit, der Standardisierung der Dokumentation und der Strukturierung der Szenarien im 6-Ebenen-Pegasus-Modell.

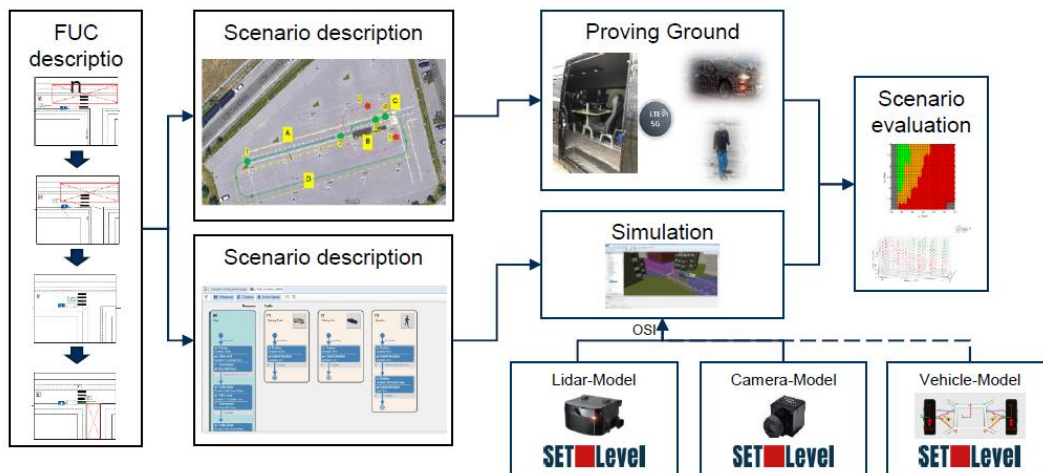


Abbildung 38: Konzeptbild zur Austauschbarkeit von Artefakten im Testprozess

Zudem wurden allgemeine Anforderungen an die Bewertung, wie die Spezifikation von GPS und V2X-Kommunikation, sowie die Standardisierung über OpenScenario und OpenDrive umgesetzt. Einheitliche Bewertungsmetriken (KPI) und der Zugriff auf vergleichbare Messwerte prototypische ermöglicht.

Technologische Ansätze wie die Verwendung von ASAM Open X-Standards für die Szenariobeschreibung sowohl für Simulationen als auch für reale Tests, sowie Datenverteilungsdienste im ROS2-Framework zur Sicherstellung der Synchronizität und Datenkonsistenz zwischen Objekten innerhalb der Testszenarien wurden implementiert. Die Kommunikation zur Übertragung von Daten zu Objekten innerhalb der Testszenarien erfolgte über WLAN und wurde später auf mobile Technologien erweitert.

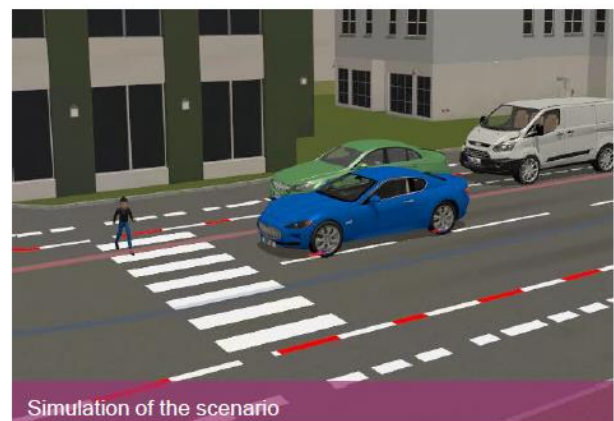


Abbildung 39: Vergleich zwischen Testgelände und Simulation

Insgesamt hat die Arbeit einen ersten Ansatz zur Seamless Integration und Vergleichbarkeit von Tests in urbanen Szenarien durch die Verwendung standardisierter und interoperabler Methoden und Technologien aufgezeigt.

2.1.6.11 Begriffsdefinitionen und Strukturierung mit Hilfe von Ontologien

Im Rahmen dieser Arbeit wurden verschiedene Begriffstrukturierungs- und Definitionsmöglichkeiten verglichen und evaluiert. Es wurden Vokabulare, Semantische Netze, Glossare hinsichtlich ihrer Anwendbarkeit untersucht. Final wurde eine Ontologie zur Strukturierung von Begrifflichkeiten aufgebaut.

Definition Ontologie: Ontologien in der Informatik sind meist sprachlich gefasste und formal geordnete Darstellungen einer Menge von Begriffen und der zwischen ihnen bestehenden Beziehungen in einem bestimmten Gegenstandsbereich. Sie werden dazu genutzt, „Wissen“ in digitalisierter und formaler Form zwischen Anwendungsprogrammen und Diensten auszutauschen.

Hierfür wurde das OpenSource verfügbare Tool Protege verwendet in welches 120 Begriffe, welche durch Literaturrecherche und Fachdiskussionen in verschiedenen Arbeitsgruppen ermittelt wurden.

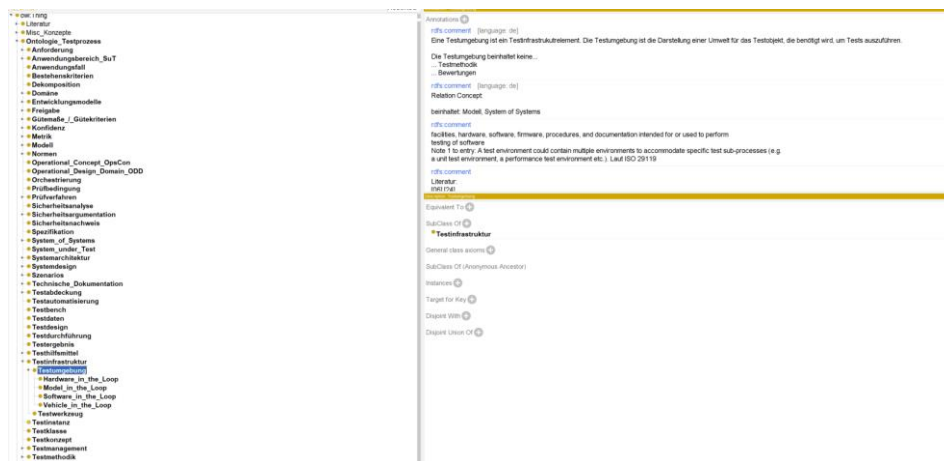


Abbildung 40: Auszug aus der aufgebauten Testing Ontologie in Protege

In der Ontologie wurden verschiedene Begrifflichkeiten im Projektkontext des TP7 in Relation gesetzt. Die Ontologie fokussiert sich dabei auf Begrifflichkeiten im Testing-Umfeld. Ziel war es hierbei ein einheitliches Verständnis zu prägen und Definitionen von verschiedenen Quellen zu sammeln.

2.1.6.12 Bestimmung des Reifegrads einer Fahrfunktion und Umsetzung in einer Datenauswertungspipeline

Für aktuelle Fahrerassistenzsysteme (FAS) bis zur Automatisierungsstufe 2 muss der Fahrer im Fehlerfall jederzeit als Rückfallebene zur Verfügung stehen. Während der Entwicklung wird daher die grundlegende Funktionalität des Systems stichprobenhaft in ausgewählten Szenarien getestet. Der Reifegrad des FAS wird beim anforderungsbasierten Testen durch das erfolgreiche Absolvieren von Testfällen bestimmt, mit denen die Erfüllung von Systemanforderungen exemplarisch nachgewiesen wird. Ob alle Systemanforderungen durch Tests überprüft werden, hängt von der gewählten Teststrategie ab.

Im Gegensatz dazu soll beim szenariobasierten Testen das FAS zukünftig in allen logischen Szenarien der Operational Design Domain (ODD) getestet werden, wobei in jedem Szenario überprüft wird, ob die Systemanforderungen erfüllt sind. Die Begriffe „Testbreite“ und „Testtiefe“ werden eingeführt, um die Testabdeckung zu beschreiben: Testbreite gibt an, in welchem Umfang

die ODD betrachtet wird, während Testtiefe angibt, bis zu welcher Systemebene Anforderungen in einem logischen Szenario betrachtet werden.

Beim anforderungsbasierten Testen wird das FAS in wenigen Szenarien detailliert betrachtet, was zu hoher Testtiefe und niedriger Testbreite führt. Das szenariobasierte Testen aus dem Forschungsprojekt PEGASUS hingegen bewertet das FAS in vielen logischen Szenarien anhand von Kritikalitätsmetriken [62] [63] [64], was zu hoher Testbreite und niedriger Testtiefe führt. Um fundierte Aussagen über den Reifegrad eines FAS treffen zu können, muss sowohl eine hohe Testbreite als auch Testtiefe erreicht werden.

Das Konzept der Reifegradbestimmung einer Fahrfunktion lässt sich in drei Schritte unterteilen: Bestimmung der Generalisierungskonfidenz, Bestimmung der szenariospezifischen Reifegrade und Aggregation aller logischer Szenarien zur Bestimmung des Reifegrads in der ODD

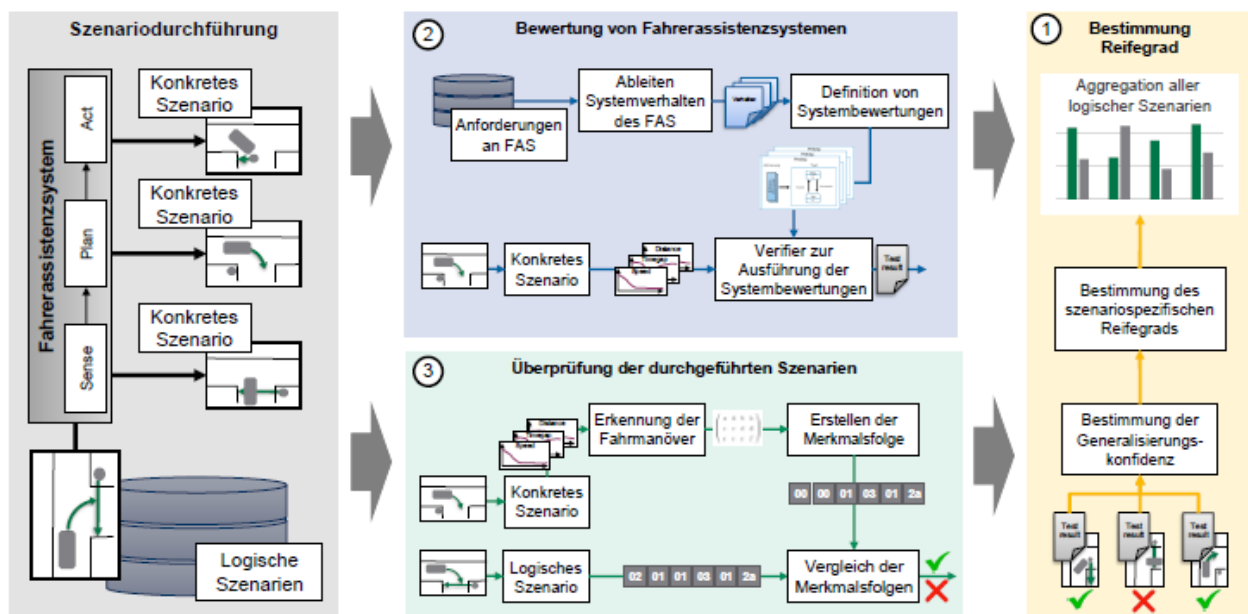


Abbildung 41: Konzept zum Testen von Fahrerassistenzsystemen in Fahrszenarien bestehend aus einer Bewertung der FAS, Validierung der Szenarien und der Bestimmung des Reifegrads

1. Reifegradbestimmung

Der Reifegrad eines FAS in der gesamten ODD wird durch die Aggregation von szenariospezifischen Reifegraden der einzelnen logischen Szenarien bestimmt. Während der Entwicklung muss nachgewiesen werden, dass das FAS in allen konkreten Szenarien korrekt funktioniert. Die Bestimmung der logischen Szenarien und die Ableitung konkreter Szenarien für das Testen sind bereits Bestandteil wissenschaftlicher Arbeiten. Für die Bestimmung des Reifegrads werden folgende Annahmen getroffen: alle zu testenden logischen Szenarien sind bekannt, ebenso die Systemanforderungen und die Anzahl der zu testenden konkreten Szenarien.

Der Reifegrad in der gesamten ODD wird durch die Aggregation der Reifegrade der logischen Szenarien berechnet. Der szenariospezifische Reifegrad hängt davon ab, in welchem Umfang die Anforderungen in den konkreten Szenarien erfüllt wurden. Da die szenariobasierte Absicherung nicht alle theoretisch möglichen Parameterkombinationen testet, sind Aussagen über den Reifegrad

mit Unsicherheit behaftet. Diese Unsicherheit muss auf ein vertretbares Minimum reduziert werden, um generalisierte Aussagen über den Reifegrad treffen zu können.

Die Generalisierungskonfidenz repräsentiert das Vertrauen in den ermittelten Reifegrad für ein logisches Szenario. Sie steigt mit jedem getesteten konkreten Szenario und ist von der Abdeckung des Parameterraums abhängig. Szenarien, die nicht den spezifizierten Inhalt eines logischen Szenarios besitzen, werden nicht zur Generalisierung herangezogen. Neue Entwicklungsstände des FAS führen zu einem Zurücksetzen der Konfidenz. Der Reifegrad in einem logischen Szenario wird durch die Aggregation der Erfüllungsgrade der Systemanforderungen und die Generalisierungskonfidenz bestimmt.

Die Bestimmung des Reifegrads lässt sich in drei Schritte unterteilen: Bestimmung der Generalisierungskonfidenz, Bestimmung der szenariospezifischen Reifegrade und Aggregation aller logischen Szenarien zur Bestimmung des Reifegrads in der gesamten ODD.

2. Bewertung

Im Rahmen des Projekts haben wir Methoden zur Quantifizierung des Erfüllungsgrades von Anforderungen für Fahrerassistenzsysteme (FAS) entwickelt. Dabei wurde für jedes konkrete Szenario ermittelt, in welchem Umfang die Anforderungen erfüllt wurden. Zur Bewertung eines FAS in zeitlich variablen Szenarien wurden zwei Herangehensweisen untersucht.

Der erste Ansatz nutzt ein Testorakel, um vor der Durchführung eines Szenarios den möglichen Verlauf anhand der Parameter zu prädictieren und daraus spezifische Testfälle abzuleiten. Diese Methode basiert auf der Vorhersage des Szenarioverlaufs, um den Zeitpunkt und die Art der Bewertung festzulegen.

Der zweite Ansatz bewertet das Systemverhalten des FAS dynamisch, sobald das relevante Verhalten auftritt. Dieser Ansatz erfordert eine Echtzeitanalyse während der Szenariodurchführung, um flexibel und genau auf das Verhalten des Systems reagieren zu können.

3. Überprüfung der durchgeführten Szenarien

Parallel dazu werden die konkreten Szenarien validiert. Dabei wird untersucht, ob alle Merkmale aufgetreten sind, die für das logische Szenario ursprüngliche spezifiziert wurden. Somit wird sichergestellt, dass die Ergebnisse der Systembewertung für die konkreten Szenarien auch für die Bestimmung des Reifegrads verwendet werden können. Hierzu wurden Szenarienbeschreibungssprachen und Merkmalbasierte Beschreibungen verwendet.

Die vorgestellte Methode wurde in einer Django Datenbank an die bestehende HiL/ SiL Testinfrastruktur angebunden und in eine kontinuierliche Bewertungspipeline integriert, welche über eine Webanwendung ansteuerbar ist und aggregierte Ergebnisse zu Szenariendurchführung liefert. Die integrierte Bewertung der konkreten Szenarien erfolgte anhand der Kritikalitätsmetriken für das automatisierte Fahren [65].

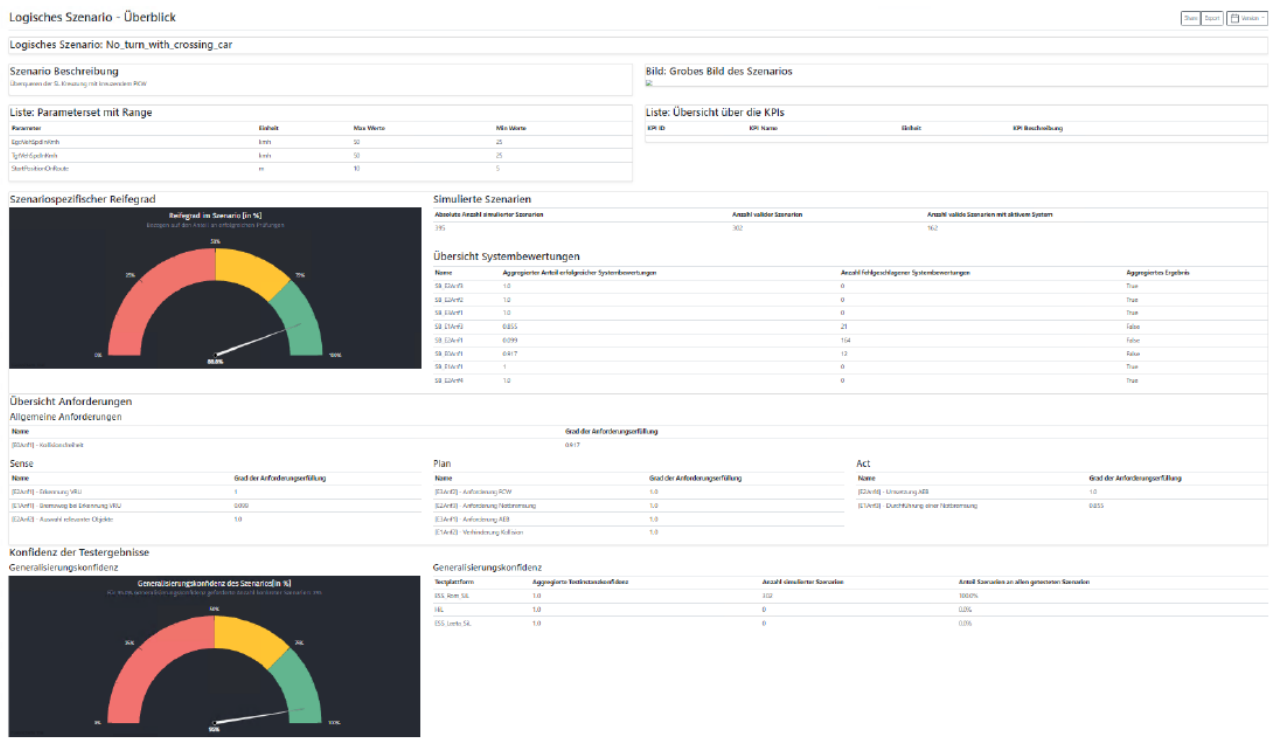


Abbildung 42: Screenshot der Detailansicht für ein logisches Szenario im FZI-Dashboard

2.1.6.13 Analyse von Umgebungseinflüssen unter Laborbedingungen als Referenzumsetzung am HiL-Prüfstand

Im aktuellen Zustand können Umwelteinflüsse auf einzelne Komponenten der hochautomatisierten Wirkungskette nur auf einem sehr hohen Reifegrad getestet werden. Mit vollständiger Hardwareintegration werden Verifizierungen in realen Fahrtests durchgeführt, um eventuelle umgebungsbedingte Einschränkungen zu ermitteln. Dies bedeutet, dass Optimierungen von Designentscheidungen nicht während des laufenden Entwicklungsprozesses umgesetzt werden können. Darüber hinaus steigt das Risiko von Unfällen während eines realen Tests aufgrund eines unerwarteten Leistungsdefizits. Mittels einer Verlagerung dieser Umfänge in semi-virtuelle Umgebungen am HiL-Prüfstand soll eine frühzeitige Erkennung von Abhängigkeiten zwischen dekomponierten Zuständen und Umweltphänomenen in Bezug auf die Leistung der Hardware abbildbar sein. Dies erlaubt eine hohe Reproduzierbarkeit bei gleichzeitiger Evaluation der realen Umgebungseinflüsse und stellt einen Durchstich zur methodischen Absicherung von nicht-funktionalen Anforderungen in virtuellen Umgebungen dar.

Dabei möchten wir Lösungsansätze aufzeigen:

- Testen der Kamerahardware mithilfe einer Kamerabox
- Beaufschlagung der Kamera mit Vibrationsbelastungen mittels Mehrachskinematik
- Die prozedurale Variation der Umgebung auf allen Straßenschichten

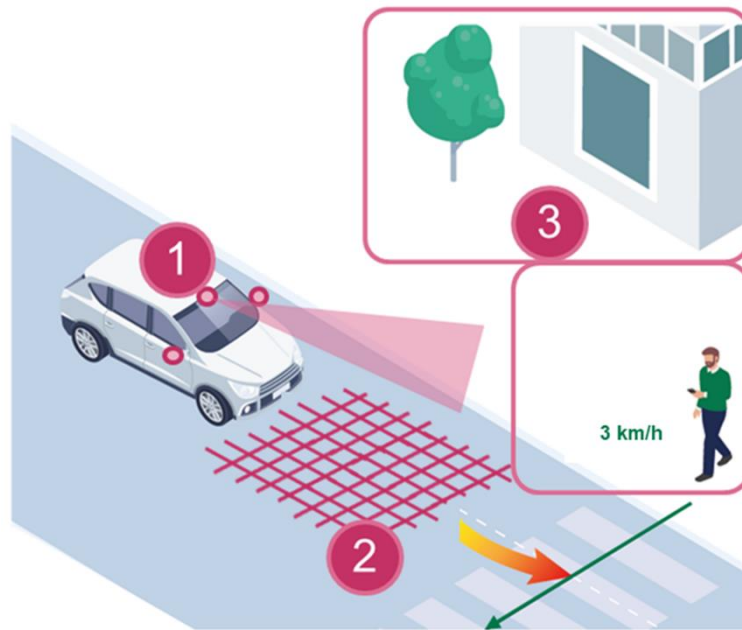


Abbildung 43: Abgezielte Realfahrten-Konditionen die unter Labor-Bedingungen verlagert werden sollen

Als Aufgabe zur Realisierung einer exemplarischen Umsetzung einer Testinstanz im Szenario-basierten Testen wird der Vibrationsprüfstand als Hardware-in-the-Loop-Testplattform für die Ursache-Wirkungsketten spezifiziert. Der Prüfstand liefert Informationen über die kinematische Kopplung zu einem konkreten Wahrnehmungsmodus (Kamera) durch eine systematische Analyse. Die Eingabe erfolgt durch Vorgabe eines Szenarios in einer virtuellen Fahrzeugsimulation, die die Beschleunigung des Kameraanbindungspunktes für Geschwindigkeiten zwischen 10 und 30 km/h auf einem Straßenprofil (Kopfsteinpflaster mit 25 cm breiten Steinen; z.B. als OpenCRG-Datei).

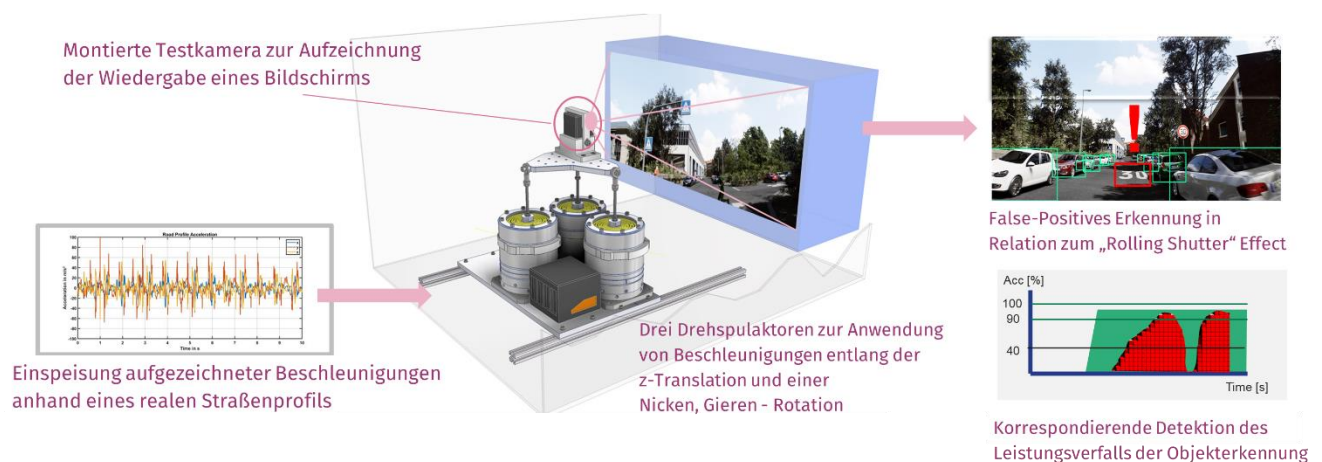


Abbildung 44: Konzeptioneller Aufbau und Methodik

Der Prüfstand ermöglicht die Variation des gesamten kinematischen Fahrzeugeigenschaften im Frequenzbereich bis etwa 200 Hz, der vom Fahrzeug zur Kamera gekoppelt ist. Es ist möglich, durch das Erfassen des Leistungsdichtespektrums der Beschleunigungen am Kameraanbindungspunkt zunächst unabhängig von den restlichen Umgebungen lokale Leistungsminima durch das Frequenzspektrum zu bestimmen. Durch eine frühzeitige systematische Erfassung einer generischen Aussage, nämlich Leistungsver schlechterung in Abhängigkeit vom Frequenzspektrum,

kann der Prüfraum erheblich reduziert werden. Unser Beobachtungsziel ist die Genauigkeit und Robustheit des bildbasierten Objektdetektors unter Test. Zusätzlich kann die visuelle Umgebung durch prozedurale Generierung variiert werden, um die Umgebung als Einflussfaktor auszuschließen und somit eine Kreuzvalidierung für die Frequenzaussage oder das Auffinden auslösender Objekte zu ermöglichen.

Die konkrete technische Referenzumsetzung dieses exemplarischen Anwendungsfalls erfolgte im Rahmen von AP7.3 und ist in Abschnitt 2.1.7.4 beschrieben.

2.1.7 AP7.3: Exemplarische Referenzumsetzung

Als AP-Leiter war das FZI für alle Aktivitäten im AP7.3 organisatorisch federführend verantwortlich. Es wurde ein regelmäßiger Austausch im AP7.3 durchgeführt, indem alle Partner ihre geplanten Beiträge vorstellen und sichergestellt werden soll, dass die einzelnen Partnerbeiträge sich zu einem noch wertvolleren Gesamtbild ergänzen.

Die Ziele, Themenschwerpunkte und Kernergebnisse des FZIs in diesem AP bezogen sich auf die durchgängige Darstellung von durchgängigem Testen eines eingebrachten System-under-Test (SuT) an einer Kreuzungssituation über alle Testinstanzen:

- Aufbau und Umsetzung verschiedener Testinstanzen und Darstellung am FZI
 - o Simulations-Testinfrastruktur (siehe Abschnitt 2.1.7.1)
 - o Wirkketten-HiL-Prüfstand für den Gesamtfunktionstest inkl. Sensormodelle, Umwelt, Bussysteme (siehe Abschnitt 2.1.7.2)
 - o Einbringung eines Versuchsträgers, Aktualisierung von Sensorik (siehe Abschnitte 2.1.7.3)
- Aufbau von durchgängigen Referenzumsetzungen für ein „Seamless Testing“ (siehe Abschnitte 2.1.7.4 und 2.1.7.5)
- Entwicklung und Erweiterung einer hochautomatisierten Fahrfunktion als System-under-Test (SuT) zur Integration in die verschiedenen Testinfrastrukturen des FZI, Schaffung von Schnittstellen, Anpassung und Optimierung von Teilkomponenten (siehe Abschnitt 2.1.7.6)

2.1.7.1 Konzeption, Aufbau und Integration einer Simulations-Testinfrastruktur

Im Rahmen des VVM wurde eine Testinfrastruktur für die Durchführung von simulativen Testumfängen in der Simulationsumgebung CARLA abgebildet. Dazu wurde insbesondere eine der Testfeldkreuzungen aus dem Testfeld Autonomes Fahren (TAF) Baden-Württemberg sowie der bereitgestellte bzw. in VVM aufgebaute Versuchsträger des FZI modelliert. Dies ermöglichte Testfahrten in der digitalen Welt und diente zur Unterstützung bei der Entwicklung des echten Versuchsträgers.

Für die Nachbildung des Versuchsträgers wurde ein CAD-Modell erstellt, welches für die Nutzung in der Simulationsumgebung CARLA angepasst und optimiert wurde. Abbildung 45 zeigt das Modell. Mithilfe der 3D-Grafiksoftware Blender wurden die Räder und das restliche Fahrzeugmodell über eine Skelettstruktur verbunden. Dies sorgt dafür, dass sich die Räder entsprechend dem Fahrverhalten drehen und die Bewegungen des Fahrzeugs realistisch animiert werden. Des Weiteren war für die Gewährleistung der Detektion durch Lidar und andere Sensorik ein weiteres hinreichendes 3D-Modell des Fahrzeugs erforderlich.

Die Kreuzung wurde zuvor in der Unreal Engine nachmodelliert und musste für die Nutzung in CARLA angepasst werden. Diesbezüglich wurden zahlreiche Objekte, wie Bäume, Zäune, Säulen und Ampeln durch spezielle CARLA-Objekte ersetzt, da sie ansonsten nicht von der Sensorik erfasst werden konnten. Um eine bessere Testeinbindung im Bereich der Kreuzung zu ermöglichen, wurde zudem die Karte erweitert, indem die Straßenarme miteinander verbunden wurden. Dies ermöglichte das durchgehende Befahren der Karte und das Zurückkehren zur Kreuzung. Die Nachbildung der Testkreuzung selbst blieb dabei realitätsgetreu. CARLA nutzt für die Simulation von Fahrzeugen und Zu Fuß Gehenden eine Karte im OpenDrive Format, die ebenfalls passend zur Karte erstellt wurde. Dabei wurden die Routen und Bereiche definiert, die die jeweiligen Verkehrsbeteiligten nutzen können.



Abbildung 45: Digitaler Zwilling der Testfeldkreuzung und des Versuchsträgers basierend auf CARLA.

Die digitalen Zwillinge von Testfahrzeug und Testkreuzung ermöglichen das Nachstellen von beobachteten Szenarien und das Testen spezifischer, beispielsweise kritischer Szenarien. Da es sich bei der simulierten Kreuzung um die Nachbildung einer Testfeldkreuzung handelt, besteht die Möglichkeit Verkehrsszenarien in der Simulation nachzubilden. Dazu wurden zunächst über die angebrachten Sensoren an der Kreuzung echte Szenarien aufgezeichnet und automatisiert die Trajektorien der einzelnen Verkehrsteilnehmenden mit einer eindeutigen ID, Position, Geschwindigkeit und Klasse (Fußgänger, Fahrrad, Auto, ...) in eine csv-Datei geschrieben. Mittels eines Skripts können die Daten aus der csv-Datei ausgelesen und jeweils entsprechende Verkehrsbeteiligte in CARLA bewegt werden, sodass das Szenario in der Simulation wieder abgespielt werden kann. Für die Wiedergabe des Szenarios in der Simulation wurden die Bewegungen der Fußgänger und Fußgängerinnen realistischer gestaltet. Im Hinblick auf das Abschlussevent wurde ein adäquates Kreuzungsszenario ausgewählt, um die Relevanz eines digitalen Zwillings für das Absichern von autonomen Fahrzeugen zu veranschaulichen (siehe Abbildung 46).



Abbildung 46: Resimuliertes Szenario (links) und reale Aufnahme (rechts).

Um sicherzustellen, dass das Testfahrzeug in der Simulationsumgebung das gleiche Verhalten wie in der Realität aufweist, wird das virtuelle Fahrzeug mit der gleichen Fahrfunktion gesteuert. Allerdings müssen die Schnittstellen für sämtliche Daten, die normalerweise aus der Umgebung kommen, angepasst werden, sodass auf die Simulationsdaten zugegriffen werden kann. Dazu zählen insbesondere die Sensordaten. Außerdem unterscheidet sich die Regelung vom Fahrzeug, die dafür sorgt, dass sich das Fahrzeug bewegt. Die Verarbeitung der Sensordaten bis hin zur geplanten Trajektorie erfolgt wie auf dem realen Versuchsträger.

Da die Karte der Testkreuzung in der Simulation angepasst wurde, mussten die Karten, die die Fahrfunktion nutzt, darauf abgestimmt werden. Die hochautomatisierte Fahrfunktion benötigt Lanelet Karten. Der darauf definierte Straßenverlauf gibt der Fahrfunktion die Fahrspuren an und muss möglichst nah an der zugrundeliegenden Karte liegen. Eine Abweichung führt zu einer Inkompatibilität zwischen den Positionen der Fahrzeuge auf der Lanelet-Karte und der Position in der Simulation.

Die Simulation kann durch ein Skript gestartet werden, welches die Kreuzung lädt und das Fahrzeug mit dem gewünschten Sensorsetup darauf platziert. Das Sensorsetup kann dabei leicht über eine Datei verändert werden, um verschiedene Positionen der Kameras und Lidarsensoren zu validieren. Dies ermöglichte vor allem vor dem Aufbau des echten Versuchsträgers die Validierung des Sensorsetups. Über das Startskript wird außerdem RViz geöffnet, ein 3D Visualisierungstool von ROS, zur Veranschaulichung der Lanelets und der Fahrzeugdaten, wie die Position des Fahrzeugs, der Sensordaten oder der Trajektorienplanung. Außerdem kann damit die Position des Fahrzeugs verändert oder ein Routen-Ziel gesetzt werden. Um verschiedene Komponenten, zum Beispiel die Perzeption, zu testen, wurden in der Simulation regelmäßig Rosbags (Aufnahmen im systemspezifischen Format) aufgenommen. In einer Rosbag werden bestimmte Informationen aufgezeichnet und können später zur genaueren Analyse wieder abgespielt werden. Die stetige Weiterentwicklung der hochautomatisierten Fahrfunktion machte außerdem eine ständige Harmonisierung dieser mit der Simulation nötig.

Für das Abschlussevent wurden verschiedene Szenarien zur Demonstration erstellt. Die ausgewählten Szenarien zeigen das Überqueren der Kreuzung einer Fußgängerin, das Abstandhalten zu einem Radfahrer und Szenarien mit entgegenkommenden Fahrzeugen beim Linksabbiegen. Abbildung 47 zeigt eine Szene aus dem Szenario, in dem das Testfahrzeug Abstand vom Radfahrer halten muss.



Abbildung 47: Links: Szene aus einem Szenario in CARLA. Rechts: RViz Ansicht mit Lidar Punktwolken, Lanelets und Planer.

2.1.7.2 Konzeption, Aufbau und Integration der Hardware-in-the-Loop Testinstanz am FZI

Ein Schwerpunkt in den Testinstanzen lag auf dem Aufbau einer Hardware-in-the-Loop-Testinstanz für die hardware-nahe Gesamtfahrzeugsimulation. Im Rahmen der Testinstanz sind Kameraboxen konzipiert, konfiguriert und abgenommen, die es ermöglicht, Wirkkettenanalysen an einer exemplarischen hochautomatisierten Fahrfunktion durchzuführen. Die Schwerpunkte seitens des FZIs lagen in der Entwicklung eines Systems-under-Test und der Erstellung von Schnittstellen zwischen dem kommerziellen dSPACE SCALEXIO Systemverbund mit Simulationsrechnern und der eigens entwickelten Teststeuerung mit Dashboard sowie dem System-under-Test. Um diese Module zu verbinden, wurden Schnittstellen und Konvertierungen entwickelt, die eine bidirektionale Kommunikation zwischen dem dSPACE VEOS Kommunikationsprotokoll und ROS1 Neotic auf Linux Distributionen ermöglichen. Diese Umrechnungen der Sensordaten erlauben die Konfiguration und Steuerung der Testszenarien, die Einspeisung synthetischer Sensordaten in das exemplarische System-under-Test und die Übergabe der Ergebnisse des SuT in Form von Fahrzeugsteuerungssignalen an das virtuelle Fahrzeug.

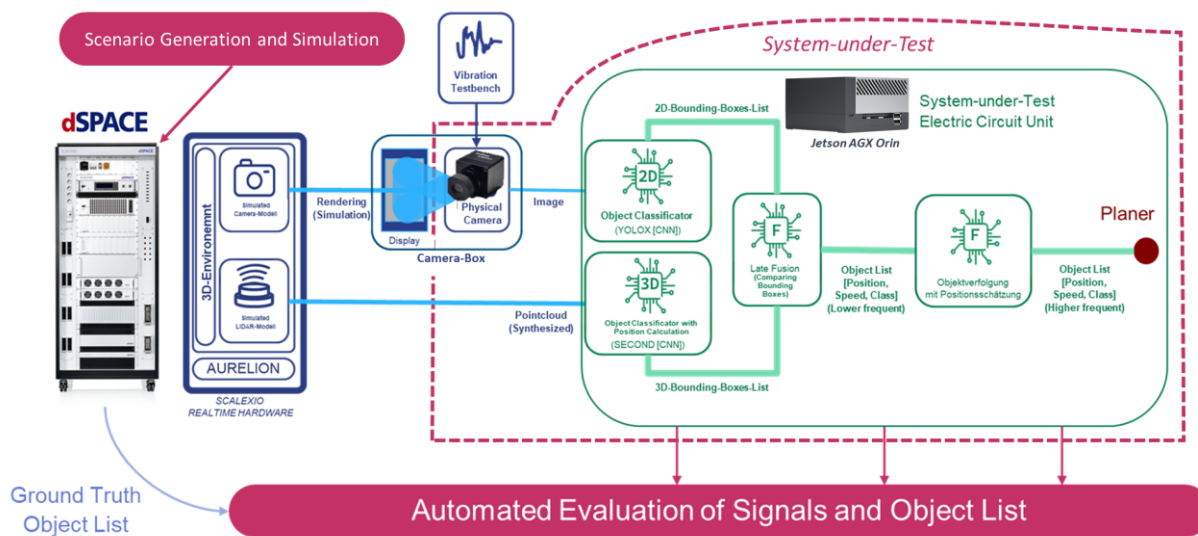


Abbildung 48: Architektur der Hardware-in-the-Loop Testinstanz am FZI

Die am FZI erstandene Testinstanz ergab folgende Wirkkette aus Testumgebung und System-under-Test: Links befindet sich die dSPACE Hardware, die SCALEXIO Echtzeithardware nutzt, um Testumgebungen zu erzeugen und zu simulieren. Die 3D-Umgebung von AURELION enthält Modelle einer simulierten Kamera und eines simulierten LiDAR-Sensors, die gerenderte Bilder und synthetisierte Punktwolken erzeugen. Diese Bilder werden auf einem Display angezeigt und von einer physischen Kamera erfasst, die auch auf einer Vibrationstestbank getestet wird, um die Robustheit der Systeme unter verschiedenen Vibrationsbedingungen zu überprüfen. Die Daten werden von einem Jetson AGX Orin verarbeitet, der das Elektronikmodul des zu testenden Systems steuert. Der 2D-Objektklassifikator YOLOX und der 3D-Objektklassifikator SECOND, beide auf CNNs basierend, verarbeiten die Bild- und LiDAR-Daten und liefern entsprechende Bounding-Box-Listen. Eine späte Fusion vergleicht und kombiniert die Bounding-Boxen beider Klassifikatoren, um eine umfassende Objektliste mit Positionen, Geschwindigkeiten und Klassen der erkannten Objekte zu erstellen. Diese Objektliste wird dann in das System zur Objektnachverfolgung und Positionsschätzung eingespeist, welches höherfrequente Objektinformationen für die Planung bereitstellt.

Die Testinstanz erlaubt eine automatisierte Auswertung der Signale und der Objektliste, um die Leistung und Genauigkeit des Systems unter verschiedenen Fußgänger kreuzenden Szenarien variierender Umstände zu beurteilen und Durchführbarkeit von Testszenarien auf KI-Testobjekte zu untersuchen. Um den Einfluss der Assemblierung der Szenen auf die Leistung des Testobjektes zu ermitteln wurden mittels einer automatisierten Platzierung von fotorealistischen 3D Objekten und eine manuelle Kartengenerierung unterschiedlicher virtueller Straßenszenen, unter anderem der Nachbau der Testfeldkreuzung k729 erarbeitet. Die Durchführung zeigte die Relevanz eines systematischen Umgangs mit der statischen Umgebung die aus Komponentensicht eine potentielle Störgröße sein kann. Zudem wurden Analysen durchgeführt bei der inkrementell die dSpace Regen- und Nebelmengen-Augmentation auf Lidar-Punktwolken das Rauschen und Verzerrungen unterzogen um die Reaktion der KI-basierten Lidar-Objekterkennung zu untersuchen. Dies zeigte eine Untersuchung von Leistungsgrenzen von Wetterumstandenen bei der Objekterkennung in einem virtuellen Rahmen und mögliche Fehlereffekte, wie die Falschklassifikation von Personen mit Fahrzeugen durch extremes Driften von Lidar-Punkten.

Bei der Durchführung der Tests wurde die Testbarkeit von Störgrößen untersucht, darunter mit einer verschmutzten oder befeuchteten Glasplatte, einer Störlichtquelle. Kontrastreiche Tropfen oder Verschmutzungen sorgten für Falschklassifikationen. Dies zeigte die Potentiale eines Kamera-Prüfstandes in der Hardware-in-the-Loop Systemumgebung um virtuell nicht abbildbare Phänomene systematisch untersuchen zu können.

2.1.7.3 Aufbau des Versuchsträgers

Im Projektverlauf wurde seitens des FZI ein ins Projekt eingebrachter Versuchsträger mit Sensorik und weiteren Hardware-Komponenten aufgerüstet, um Testumfänge für das automatisierte Fahren im urbanen Raum abzubilden. Dies beinhaltet eine umfassende Konzeptionierung des Umbaus, insbesondere die Auswahl und Anordnung der Sensoren betreffend. Um den Anforderungen an den industriellen Standard und dem aktuellen Stand der Forschung gerecht zu werden, wurde ein anderweitig finanziertes, neues Fahrzeug als Basisplattform aufgebaut. Der Audi A6 Avant Plug-In Hybrid (siehe Abbildung 50) wurde als neue Basisplattform für die im Rahmen von VVM geplante Integration und Einbau von Sensorik und Rechenkomponenten eingesetzt.

Für die Erstellung eines ausführlichen Lastenhefts für den Umbauprozess wurden mehrere Workshops abgehalten. Ziel hierbei war es die Anforderungen an die Sensorik sowie an die Rechnerkomponenten in unterschiedlichsten Funktionsszenarien zusammenzutragen. Dabei wurde auch die Komplexität des Verkehrsszenarios variiert, um beispielsweise die Hardware an die Funktional Use Cases aus der Impulsgebung des Projekts, als auch an die schlussendlichen Bedingungen im realen Testbetrieb anzupassen. Nicht nur die reine Perzeption, sondern auch die daraus resultierende Umgebungsabbildung spielt für das automatisierte Fahren eine wichtige Rolle. Dementsprechend wurden auch die Rechnerkomponenten auf den zu erwartenden, notwendigen Detaillierungsgrad der Perzeption ausgelegt. Weiterhin wurde der Energiebedarf aller Komponenten feingranular erfasst und optimiert, um trotz der leistungsfähigen Hardware für den mobilen Einsatz geeignet zu sein. Insbesondere die Auswahl der Rechnerkomponenten war diesbezüglich getroffen worden. Für das Sensorsetup wurden Simulationen in Carla entwickelt und durchgeführt, um tote Winkel und Verdeckungen zu minimieren.

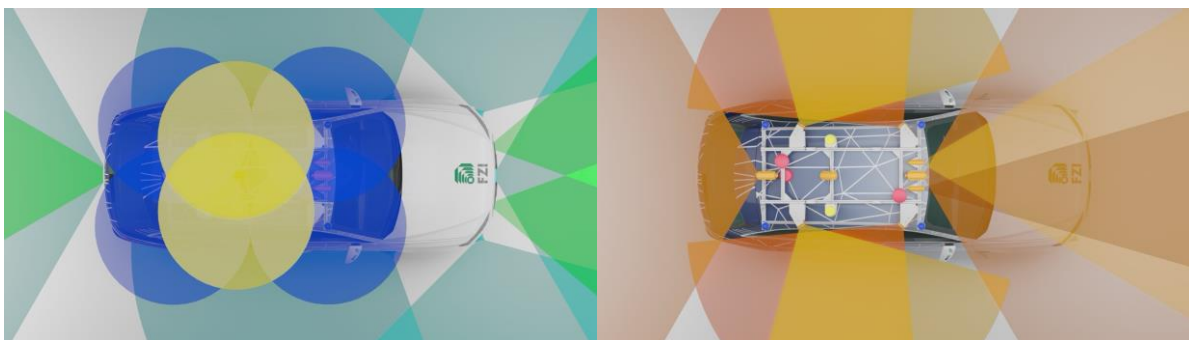


Abbildung 49: Lidar und Radar (links) sowie Kamera (rechts) FOV Versuchsträger (rechts)

Der Umbau wurde an einen Unterauftragnehmer vergeben. Während der Umbauphase wurde der Fortschritt sowohl mit Konferenzen als auch durch mehrere Vor-Ort-Besuche überwacht und abgenommen. Durch Lieferschwierigkeiten einiger Komponenten im Zuge der Pandemie wurde das Fahrzeug erst im letzten Projektjahr fertiggestellt.

Anschließend wurden umfassende Integrationsarbeiten durchgeführt, um die Hardware ansteuern zu können. Hierzu gehört neben der Kalibrierung und Synchronisierung der Sensoren auch die Konfiguration des GPS Inertialsystems sowie die Einrichtung des Hauptrechners.



Abbildung 50: FZI Versuchsträger CoCar NextGen auf dem VVM Abschlussevent

Die Integration der im Rahmen von VVM weiterentwickelten automatisierten Fahrfunktion ermöglichte die Darstellung von realistischen urbanen Szenarien, beispielsweise dem Abbiegen oder langsames Auffahren auf Objekte mit Verdeckung. Der fertige Versuchsträger konnte für reale Testfahrten auf dem Testfeld Autonomes Fahren (TAF) Baden-Württemberg verwendet werden. Abbildung 50 zeigt den finalen Aufbau des Versuchsträgers im Rahmen des Abschlussevents von VVM. Der Versuchsträger ist gleichzeitig auch ein Instrument des FZI, um Forschung rund um die Entwicklung und Absicherung von automatisierten Fahrfunktionen weiter voranzubringen sowie die Sichtbarkeit der Themen beispielsweise über Messen, Veranstaltungen oder Testfahrten zu erhöhen, siehe auch Abschnitt Verwertung. Eine realistische Sicht der Fahrzeug-Sensorik ist in Abbildung 51 dargestellt.

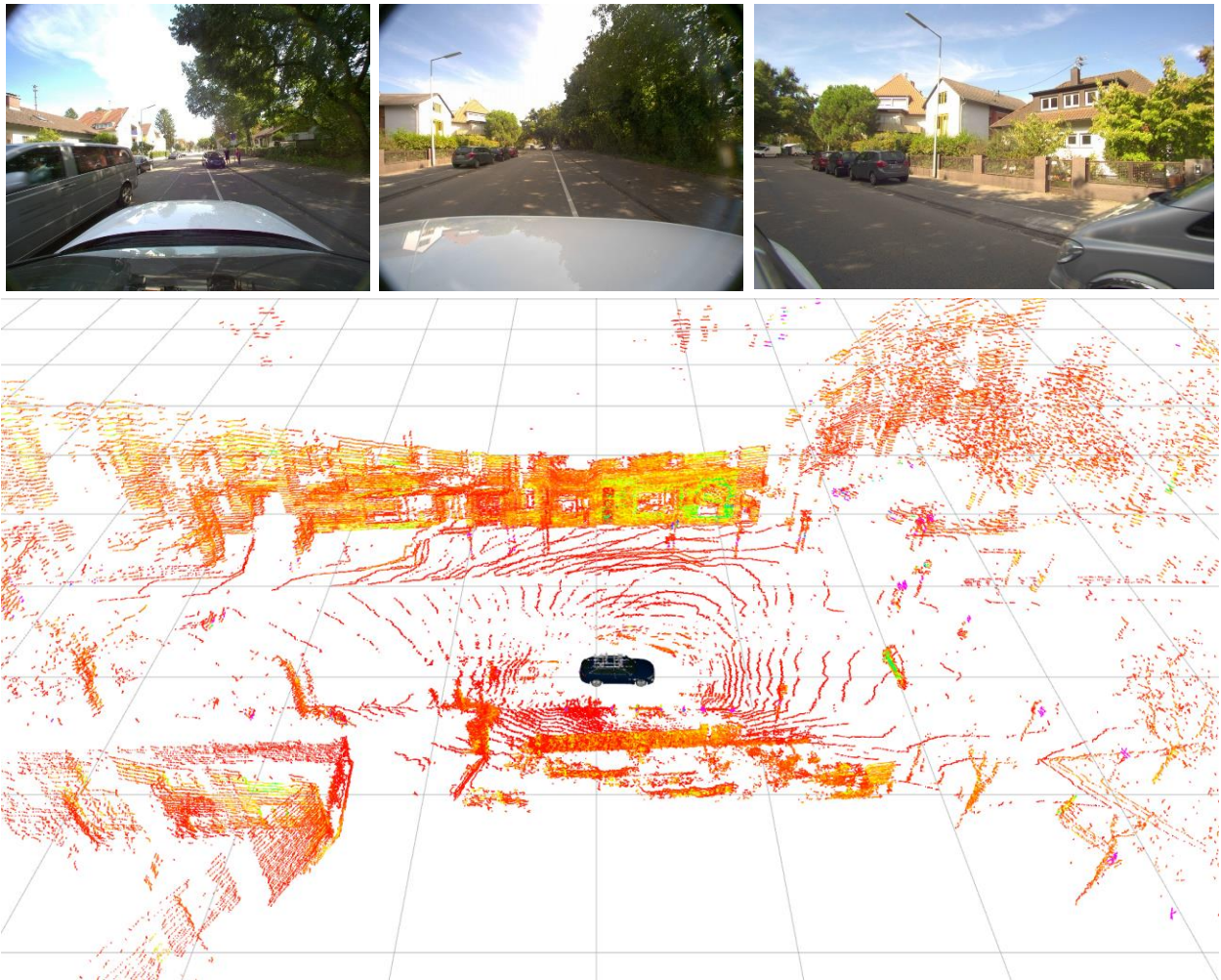


Abbildung 51: Lidarpunktwolke und exemplarische Kamerabilder aus dem Versuchsträger

2.1.7.4 Integration des Schwingungsprüfstand an das Hardware-in-the-Loop System

Der Hardware-in-the-Loop-Prüfstand inklusive Kameraboxen wurde in Zusammenarbeit mit dem Fraunhofer LBF am FZI erweitert, um nicht-funktionale Anforderungen in einem durchgängigen Anwendungsbeispiel aufzuzeigen. Hierbei war nach fachlicher Diskussion im Projekt abgestimmt worden, dass der HiL-Prüfstand inklusive Kamerabox des FZI einen passenden Anwendungsfall darstellte. Die Grundidee war es, nicht-funktionale Anforderungen bzw. Effekte wie Schwingungen in der Simulation zu testen, beispielsweise wie die Vibration des Fahrzeugs die physische Kamera bzw. die Perzeption beeinflusst. Bisher war hierfür nur eine Defektanalyse betrieben worden (z.B. wie viel Aufschlag eine Hardware aushält, bevor sie ausfällt). Die Vibrationen konnten zu klassischen Bildverzerrungseffekten und auch zu Störausfällen im Chip führen. Das Fraunhofer LBF als Projektpartner hatte hierfür einen Schwingungsprüfstand entwickelt, um diese Effekte aufzuzeigen.



Abbildung 52: Konzeptioneller Aufbau des Kamera-Schwingungsprüfstandes

Ziel war es, den Schwingungsprüfstand des Fraunhofer LBF in den HiL-Prüfstand des FZI inklusive Kamerabox zu integrieren. Der konzeptionelle Aufbau des Prüfstandes ist in Abbildung 52 dargestellt.

Im Kameraprüfstand des FZI sind die realen Kameras verbaut inkl. Kamera-Perzeption. Mittels Realdaten-Einspeisung von Schwingprofilen von Straßen, wie Kopfsteinpflastern lässt sich die reale Kamera mit Schwingungen innerhalb der Kamerabox anregen. Die durch Abfilmen des Kamerabox-Bildschirm und gerenderten Simulationsbild entstehenden Bilddaten werden in die Funktion auf Edge-Computing Hardware eingespeist. Dies ermöglichte eine technische Evaluation der Objektdetektorperformance, mit gegenübergestellter Intensität an Vibrationsamplitude. Somit können Möglichkeiten neuer HW/SW-Testmittel und methodischer Einbindung von nicht-funktionalen Anforderungen bzw. deren Auswirkungen auf das Teilfunktionen sowie das Gesamtsystem in der gesamten Wirkkette aufgezeigt werden, um etwaige Lücken bei der Durchführung von Testszenarien zu erschließen. Zudem ermöglicht der Aufbau, reale Schwingungsprofile auch in einer simulativen Umgebung einzubringen. Für das Abschlussevent wurde der Demonstrator aufgebaut und ein Dashboard für die Testergebnisse entwickelt. Die Demonstration zeigte einerseits als UseCase die Vorteile der prozeduralen Generierung und andererseits die automatisierte technische Evaluation mit inkrementeller Applikation der Schwingung auf die Kamera um nicht tolerables Verhalten der Funktion aufzudecken.

2.1.7.5 Aufbau der Simulation in CARLA für die hochautomatisierte Fahrfunktion

Um ein realistisches urbanen Szenario für die simulative Absicherung zur Verfügung zu haben und somit eine Durchgängigkeit von der Simulation bis hin zum Realtest zu erzielen, wurde die zentrale Kreuzung des Testfelds Testfeld Autonomes Fahren (TAF) BW für eine Abbildung in der Simulationsumgebung ausgewählt. Die Anordnung des digitalen Zwillings für die Kreuzung k729 aus dem TAF BW musste noch einmal angepasst werden, da es eine geringe Abweichung der

Positionen der Karte zu der Lanelet Karte gab. Zudem wurde festgestellt, dass die Spawnpoints und die Routen für Fahrzeuge fehlerhaft positioniert werden. Die stetige Weiterentwicklung der HAF macht außerdem eine ständige Harmonisierung der HAF mit der Simulation nötig. Dabei wurde unter anderem die Planungskomponente der HAF optimiert indem ein Routenfile für die Testfeldkreuzung erstellt wurde. Für das Abschlussevent wurde ein auf der k729 aufgezeichnetes Szenario ausgewählt, das die Interaktion verschiedener Verkehrsteilnehmer zeigt, und weiter aufbereitet. Für die Wiedergabe des Szenarios in der Simulation wurden die Fußgängerbewegungen beim Prozess der Imitation von Realdaten (Trajektoriendaten) in der Simulation realistischer gestaltet.



Abbildung 53: Szene aus dem resimulierten Szenario von der Testfeldkreuzung k729

2.1.7.6 Erweiterung einer hochautomatisierten Fahrfunktion (HAF) als System-under-Test

Im Rahmen des Projektes wurde eine hochautomatisierte Fahrfunktion als System-under-Test weiterentwickelt für die benötigten Use Cases im urbanen Raum. Die Fahrfunktion baut hierbei auf der Integration verschiedener Sensor-Technologien auf, um eine Abbildung des Stand der Technik darzustellen. Die Integration erfolgte in den eingebrachten Versuchsträger des FZI sowie in die simulativen Testinfrastrukturen.

Kameraperzeption

Das Umgebungsmodell wird aus Informationen zweier Quellen gefüllt. Eine Quelle stellt die Punktwolke von LiDAR Sensoren da. Die andere Informationsquelle ist ein Videostream einer Kamera. Sowohl in der Punktwolke als auch im Kamerabild sollen dynamische und statische Objekte erkannt werden um ein belastbares Umgebungsmodell zu erstellen. Für die Kameraperzeption wurde ein Konzept entwickelt, welches einen monokularen Objektdetektions-Ansatz verfolgt. Dieser Algorithmus soll Verkehrsteilnehmer in drei Dimensionen im Kamerakoordinatensystem detektieren. Dem aktuellen State-Of-The-Art folgend wurde ein Ankerboxfreies Convolutional Neural Network implementiert und verwendet. Trainiert wurde das maschinelle Lernverfahren mithilfe eines Datensatzes, welcher aus Bildern der Realität besteht. Es existiert ein Domänenshift zwischen Bildern aus einer Simulation und echten Bildern. Da der endgültige Anwendungszweck die Realität darstellt und gute Datensätze mit Echtbildern existieren wurden Echtbilder als Trainingsdaten verwendet. Erste Ergebnisse des neuronalen Netzes sind vielversprechend und zeigen, dass trotz des Trainings auf einer anderen Domäne die Detektion auch in der Simulation möglich ist. Die Detektionen müssen daraufhin mithilfe eines Trackingalgorithmus verfolgt und folgenden Gliedern der Wirkungskette zur Verfügung gestellt werden.

Lidarperzeption

Neben der Kameraperzeption werden Punktwolken von Lidar-Sensoren als eine weitere Komponente zur Perzeption des Umfelds genutzt. Diese liefern Informationen über das 3-dimensionale Umfeld des Fahrzeugs. Um die Daten der Lidar-Sensorik für Fahrfunktionen nutzen zu können, ist eine semantische Einordnung der resultierenden 3D-Strukturen notwendig. Dies geschieht in unserem Fall über eine semantische Segmentierung. Hierbei wird jeder 3D-Punkt einer Objektklasse (z.B. Auto, Straße, Vegetation, etc.) zugeordnet.

Für die Prädiktion von Objektklassen für 3D-Punktwolken wird eine eigens entwickelte Architektur von Sparse 3D-Convolutional Neural Networks verwendet. Während der Entwicklung lag der Fokus auf der Echtzeitfähigkeit des Ansatzes. Somit können die Punktwolken von hochauflösenden Lidar-Sensoren mit bis zu 1.3 Millionen Punkten pro Sekunde auf einer einzelnen handelsüblichen Grafikkarte verarbeitet werden. Dies entspricht einem modernen 128-Zeilen Lidar-Sensor mit einer Aufnahmezeit von 10 Hz.

Da die punktweise Annotation von Lidar-Punktwolken mit großem Aufwand verbunden ist, werden hierfür häufig öffentlich verfügbare Datensätze verwendet. Jedoch ist der Transfer von öffentlichen Datensätzen zu realen Sensordaten nicht trivial. Wie in Abbildung 54 (Lidar-Segmentierung aus der Vogelperspektive) zu erkennen, funktioniert der Transfer eines auf dem SemanticKITTI-Datensatz trainierten Modells mit mäßigem Erfolg auf Sensordaten des FZI-Forschungsfahrzeug CoCar NextGen mit einem ähnlichen Sensorsetup. Es sind jedoch auch deutliche Artefakte, besonders im Bereich der Boden-Klassifizierung (Straße, Vegetation und Bürgersteig) zu erkennen.

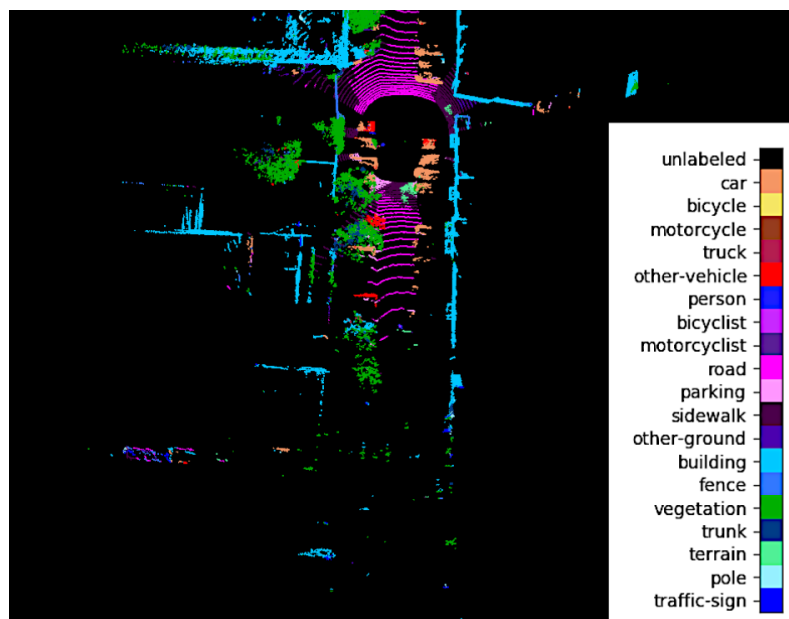


Abbildung 54: SemanticKITTI Dataset – Lidarsegmentierung

Fahrfunktion

Im Rahmen von VV Methoden wurde die hochautomatisierte Fahrfunktion, die seitens des FZI eingebracht und weiterentwickelt wurde, für die Use Cases im Projekt VVM angepasst und erweitert. Hierfür wurde der Funktionsumfang erweitert sowie Integrationshilfe beim Einsatz der Fahrfunktion durch verschiedene Partner im Projekt geleistet. Insbesondere wurde die Funktion dahingehend weiterentwickelt, dass sie statische Hindernisse überholen kann. Weiterhin wurde die bestehende Funktionalität in ihrer Robustheit verbessert. Außerdem wurde das Verhalten in komplexen

Kreuzungsszenarien überarbeitet. Dies geschah insbesondere durch eine fortschrittliche Entscheidungsfindung bezüglich Haltepunkte, zum Beispiel beim Linksabbiegen mit Gegenverkehr.

2.2 Notwendigkeit der Förderung

Die in VVM angestrebten technischen und wissenschaftlichen Ziele im Bereich der Absicherung von automatisierten Fahrfunktionen im urbanen Umfeld waren anspruchsvoll und von großen technologischen Herausforderungen begleitet. Dem potentiellen Technologiehub stand damit ein hohes Risiko gegenüber. Nur durch die Förderung wurde eine umfassende, technisch tiefe Zusammenarbeit des Konsortiums, in dem alle relevanten Rollen von Akteuren im Themenfeld vertreten waren (OEM, Zulieferer, Technologieentwickler, Wissenschaft und Forschung) ermöglicht. Hierbei ist die breite Allianz der deutschen OEMs in VVM zu erwähnen, um im vorwettbewerblichen Bereich geschlossen zu agieren. So konnten Fortschritte im Bereich der Test und die Absicherung automatisierter Straßenfahrzeuge erzielt werden, die eine breite Akzeptanz zu finden versprechen. Insbesondere im Bereich eines übergreifenden, strukturierenden Frameworks für die Sicherheitsargumentation, neuartigen Testmethoden sowie dem Ansatz des durchgängigen Testens konnte in Abstimmung mit allen Beteiligten ein gemeinsames Verständnis und gemeinschaftliche Ergebnisse erzielt werden. Die praktische Anwendbarkeit der Methoden konnte in gemeinsam umgesetzten Demonstrationen gezeigt werden, welche auch auf dem Abschlussevent und in gemeinsamen Publikationen und Deliverables öffentlich zur Verfügung gestellt werden. So hat der Ansatz, die akteurübergreifende Zusammenarbeit in einem Verbundprojekt zu fördern, wesentlich dazu beigetragen, im vorwettbewerblichen Bereich Übereinkünfte zu treffen, die zukünftig in wirtschaftliche Umsetzungen einfließen und fortschrittliche Technologien auf dem aktuellen Stand der Wissenschaft und Technik für alle Akteure verfügbar machen.

Das Vorhaben VVM leistete zudem einen Beitrag zur Sicherung der Wettbewerbsfähigkeit sowie Etablierung des Standort Deutschlands als Innovationsstandort. Die internationale Strahlkraft, insbesondere auch bzgl. dem deutschen Verständnis von Sicherheit und Qualität im Automobilbereich konnte nachhaltig verstärkt werden, da die Projektergebnisse aus VVM und das Projekt selber bzw. die gesamte Projektfamilie PEGASUS große internationale Aufmerksamkeit erhalten hat. Dies zeigte sich auch am Abschlussevent, wo der zweite Tag vollständig im Fokus des internationalen Austauschs stand. Zudem wurden alle wesentlichen Ergebnisse, Deliverables und Poster aus VVM in Englisch publiziert, um diese gemeinschaftliche Botschaft der deutschen Automobilindustrie in der Außendarstellung zu transportieren. Dies stärkt die internationale Wettbewerbsfähigkeit des Standortes Deutschland, bzgl. der Argumentation zur sicheren Einführung von automatisierten Fahrzeugfunktionen führend zu sein. Dem Ziel der Bundesregierung zur Stärkung der Innovationskraft im Bereich der Fahrzeug- und Systemtechnologien konnte auch durch VVM beigetragen werden. Das Forschungsvorhaben wäre ohne finanzielle Förderung durch das BMWK nicht durchführbar gewesen.

2.3 Bekannte Fortschritte auf dem Gebiet des Vorhabens bei anderen Stellen

Es sind keine relevanten Fortschritte anderer Stellen auf dem Gebiet des Vorhabens bekannt.

2.4 Erfolgte Veröffentlichungen

Im Rahmen des Projektes wurden folgende Veröffentlichungen und Vorträge hauptverantwortlich durch das FZI publiziert:

- “Evolutionary Behavior Tree Generation for Dynamic Scenario Creation in Testing of Automated Driving Systems” (International Conference on System Reliability and Safety, 22.-24.11.2023, Bologna)
- “Cyclist behavior: a survey on characteristics and trajectory modeling” (IEEE ITSC 2023, 24.-28.09.2023, Bilbao)
- “Utilizing Hybrid Trajectory Prediction Models to Recognize Highly Interactive Traffic Scenarios” (IEEE ITSC 2023, 24.-28.09.2023, Bilbao)
- “Paving the Ways for Vehicles-in-the-loop: Wireless Integration of Autonomous Vehicles into Mixed Reality Proving Grounds and Test Areas” (IEEE ITSC 2023, 24.-28.09.2023, Bilbao)
- “Fast Environment Generation Methods for Virtual Testing” (IEEE ITSC 2023, 24.-28.09.2023, Bilbao)
- “Traffic Scene Similarity: a Graph-based Contrastive Learning Approach” (IEEE SSCI 2023, 05.-08.12.2023, Mexiko Stadt)
- “From Model-Based to Data-Driven Simulation: Challenges and Trends in Autonomous Driving” (IEEE Computer Vision and Pattern Recognition Conference, 19.06.2023, Vancouver)
- „Data-driven development and testing for autonomous driving functions” (Vortrag auf der ADAS & Autonomous Vehicle Technology Expo, 15.06.2023, Stuttgart).
- „Inverse Universal Traffic Quality - a Criticality Metric for Crowded Urban Traffic Scenes“ (IEEE Intelligent Vehicles (IV), 04.06.2023, Anchorage)
- „A Comprehensive Review on Ontologies for Scenario-based Testing in the Context of Autonomous Driving“ (IEEE Intelligent Vehicles (IV), 04.06.2023, Anchorage)
- „1001 Ways of Scenario Generation for Testing of Self-driving Cars: A Survey“ (IEEE Intelligent Vehicles (IV), 04.06.2023, Anchorage)
- „Fingerprint of a Traffic Scene: an Approach for a Generic and Independent Scene Assessment“ (International Conference on Electrical, Computer, Communications and Mechatronics Engineering, 16.11.2022, Laguna)
- „Self Supervised Clustering of Traffic Scenes using Graph Representations“ (International Conference on Electrical, Computer, Communications and Mechatronics Engineering, 16.11.2022, Laguna)
- „Towards Traffic Scene Description: The Semantic Scene Graph“ (IEEE ITSC, 12.10.2022, Macau)
- „Maneuver-based Visualization of Similarities between Recorded Traffic Scenarios“ (DATA2022, 11.07.2022, Lissabon)
- „A Unified Description of Proving Grounds and Test Areas for Automated and Connected Vehicles“ (IEEE Intelligent Vehicles (IV), 04.-09.06.2022, Aachen)
- „Bewertungskonzept für eine entwicklungs-begleitende Szenario-basierte Absicherung“ (Vortrag auf der Fachtagung „Automatisiertes Fahren“, 29.-30.03.2022, München)

- „Projektvorstellung VVMethoden“ (Testfeld Autonomes Fahren BW Abschlussevent, 12./13.11.2021, Karlsruhe)
- „Data-Driven Development for Trust-Worthy Systems – Enabling Self-Learning and Autonomous Vehicles“ (Safetrans Industrial Day, 04.06.2021, Online)

3 Abbildungsverzeichnis

Abbildung 1: Die TP's aus VVM und deren Zusammenhang.....	10
Abbildung 2: Die drei Arbeitsphasen in VVMMethoden.	12
Abbildung 3: Testpyramide für die Absicherung	13
Abbildung 4: Verifikation und Validierung im Entwicklungsprozess	14
Abbildung 5: Zusammensetzung des VVM-Konsortiums.....	17
Abbildung 6: Nachahmung der Straßengeometrie mit Clothoiden (links), Straßengeometrie (rechts). Die Orientierung des Straßenabschnitts ist in der linken Abbildung gedreht.	19
Abbildung 7: Konvertierte Opendrive Karte aus Lanelet.....	19
Abbildung 8 Geometrische Abbildung der elementaren Straßenabschnitte (A,B,C,D;E;F), die durch die Fahrspurränder definiert werden (a). Der resultierende Straßengraph von der Straßengeometrie aus a). Mögliche Beziehungen (consecutive, adjacent, overlapping) werden durch Kanten zwischen den Knoten definiert (b).....	21
Abbildung 9: Das gleiche Straßennetz wie in Abbildung 8 mit sechs Projektionsidentitäten m von fünf Fahrzeugen (a). Der resultierende Szenegraph, in dem jeder Verkehrsteilnehmer durch einen Knoten und die Beziehungen zwischen seinen Projektionsidentitäten als Kanten dargestellt sind (b)	22
Abbildung 10: Zwei beispielhafte Verkehrsszenen bei unterschiedlichen Straßengeometrien (unten) und die entsprechenden Semantischen Szenegraphen (oben). Die Knoten wurden eingefärbt, um die Visualisierung von Gruppen zu verbessern	23
Abbildung 11: Embedding Pipeline für die Graphentriplets [16].....	25
Abbildung 12: Der PCA-reduzierte Embedding Space, eingefärbt nach der Durchschnittsgeschwindigkeit der in der Szene vorhandenen Fahrzeuge (links) und nach der Anzahl der Fahrzeuge (rechts) [16].....	27
Abbildung 13: Der Embedding Space S wurde mit Hilfe des UMAP-Algorithmus auf einen 2-dimensionalen Raum reduziert. Hierarchisches Clustering wurde eingesetzt, um die Datenpunkte 25 verschiedenen Clustern zuzuordnen, die durch unterschiedliche Farben dargestellt werden [16] .	28
Abbildung 14: Beispiel eines Testfalls für ein Notbremssystem.....	29
Abbildung 15: Abstraktionsebenen von Szenarien nach [39].....	30
Abbildung 16: Testprozess [2].....	32
Abbildung 17: Testinstanzen und Testitems angelehnt an [49].....	33
Abbildung 18: Klassifikation von Testumgebungen angelehnt an PEGASUS [39]	33
Abbildung 19: Raum für die Kategorisierung von Szenario-Ontologien auf der Grundlage des statischen und dynamischen Abstraktionsebene. Die Realität befindet sich im Ursprung.....	35
Abbildung 20: Einordnung der V&V Anteile in der Gesamtmethodik	37
Abbildung 21: Hierarchische Repräsentation der Kapitel der technischen Testspezifikation	40
Abbildung 22: Seamless Testing über verschiedene Testinstanzen	42
Abbildung 23: Beispielhafte Klassifizierung von Ansätzen zur Verifikation & Validierung [55]	44
Abbildung 24: Unterschiedliche Eigenschaften von Testplattformen	45

Abbildung 25: Vergleich verschiedener Testplattformen auf der Grundlage des Integrationsgrads	45
Abbildung 26: Gesamtarchitektur des Testsystems für die Integration mehrerer Fahrzeuge.	48
Abbildung 27: Blockdiagramm zur Veranschaulichung des Simulationsprozesses. Die Vorschläge für die Trajektorien werden aus einer gesehenen Szene berechnet, wobei verschiedene Verhaltensmodelle auf der Grundlage der aktuellen Situation berücksichtigt werden.	50
Abbildung 28: Die Abszisse ist die Kritikalität und die Ordinate die Dichte. Die blauen und roten Farben zeigen die Verteilung der Kritikalität für alle Child-simulationen an.	52
Abbildung 29: Das Diagramm zeigt die kumulative Dichte für Kritikalitätsmetriken für 2 Seed-Szenen (rot und blau)	53
Abbildung 30: Extrahierte und ersetzte Objekte in FUC 2.3	54
Abbildung 31: Digitaler Zwilling in Karlsruhe für FUC 2.3	54
Abbildung 32: Anforderungen und Konzeptbild für die hinreichende Reduktion von Szenarien	55
Abbildung 33: Schritte des Selektionsprozess zur Auswahl von repräsentativen konkreten Szenarien für ein logisches Szenario	55
Abbildung 34: Aufbau eines Methodenbausteins im Selektionsprozess	56
Abbildung 35: Konzept der digitalen Erprobung mit kontinuierlichem Bewertungskonzept	57
Abbildung 36: Anwendung im Entwicklungsprozess.....	58
Abbildung 37: Gemeinsame Datenbasis und Evaluationsmetriken über verschiedene Testinstanzen	59
Abbildung 38: Konzeptbild zur Austauschbarkeit von Artefakten im Testprozess	60
Abbildung 39: Vergleich zwischen Testgelände und Simulation	60
Abbildung 40: Auszug aus der aufgebauten Testing Ontologie in Protege	61
Abbildung 41: Konzept zum Testen von Fahrerassistenzsystemen in Fahrszenarien bestehend aus einer Bewertung der FAS, Validierung der Szenarien und der Bestimmung des Reifegrads	62
Abbildung 42: Screenshot der Detailansicht für ein logisches Szenario im FZI-Dashboard	64
Abbildung 43: Abgezielte Realfahrten-Konditionen die unter Labor-Bedingungen verlagert werden sollen	65
Abbildung 44: Konzeptioneller Aufbau und Methodik	65
Abbildung 45: Digitaler Zwilling der Testfeldkreuzung und des Versuchsträgers basierend auf CARLA.....	67
Abbildung 46: Resimuliertes Szenario (links) und reale Aufnahme (rechts).....	68
Abbildung 47: Links: Szene aus einem Szenario in CARLA. Rechts: RViz Ansicht mit Lidar Punktwolken, Lanelets und Planer.	69
Abbildung 48: Architektur der Hardware-in-the-Loop Testinstanz am FZI	70
Abbildung 49: Lidar und Radar (links) sowie Kamera (rechts) FOV Versuchsträger (rechts)	71
Abbildung 50: FZI Versuchsträger CoCar NextGen auf dem VVM Abschlussevent.....	72
Abbildung 51: Lidarpunktwolke und exemplarische Kamerabilder aus dem Versuchsträger	73
Abbildung 52: Konzeptioneller Aufbau des Kamera-Schwingungsprüfstandes.....	74

Abbildung 53: Szene aus dem resimulierten Szenario von der Testfeldkreuzung k729	75
Abbildung 54: SemanticKitti Dataset – Lidarsegmentierung	76

4 Tabellenverzeichnis

Tabelle 1: Regressionsleistung des Embedding Space auf handgefertigte Merkmale [16]	26
Tabelle 2: Auszug von abgeleiteten Anforderungen an Tests und Testinstanzen.....	35
Tabelle 3: Auszug einer exemplarischen technischen Testspezifikation für die Umfänge des FUC2.3 anhand eines AEB	40
Tabelle 4: Auszug an Kriterien für die Evaluation von Testinstanzen	46
Tabelle 5: Modelle die in der Simulation genutzt werden.....	51

5 Literaturverzeichnis

- [1] I. Industrieanlagen-Betriebsgesellschaft, „V-Modell 97 - komplett im Format Word (Teil 1-3),“ Ottobrunn, 1997.
- [2] E. Sax, *Automatisiertes Testen eingebetteter Systeme in der Automobilindustrie*, Hamburg: Carl Hanser Verlag, 2008.
- [3] Verband der Automobilindustrie (VDA), „Automotive SPICE Prozessassessment Version 2.3,“ 2007.
- [4] A. Albers, „X-in-the-Loop-Framework für Fahrzeuge, Steuergeräte und Kommunikationssysteme,“ *ATZ elektronik*, pp. 60-65, Mai 2010.
- [5] B. Schick, J. Henning, U. Wurster und B. Klein-Ridder, „Simulation methods to evaluate and verify functions, quality and safety of driver assistance systems in the continuous MiL, SiL and HiL process,“ in *3rd Conference Active Safety through Driver Assistance*, 2008.
- [6] S. Otten, J. Bach, C. Wohlfahrt, C. King, J. Lier, H. Schmid, S. Schmerler und E. Sax, „Automated Assessment and Evaluation of Digital Test Drives,“ in *Advanced Microsystems for Automotive Applications 2017*, Karlsruhe, Springer Verlag, 2017.
- [7] J. Bach, M. Holzaepfel, S. Otten und E. Sax, „Reactive-Replay approach for verification and validation of closed-loop control systems in early development,“ 2017.
- [8] M. Conrad, „A Systematic Approach to Testing Automotive Control Software,“ 2004.
- [9] W. Wachenfeld und H. Winner, *The Release of Autonomous Vehicles*, Springer, 2016.
- [10] J. Mazzega, F. Köster und K. Lemmer, „Absicherung hochautomatisierter Fahrfunktionen,“ *ATZ*, 2016.
- [11] J. Bach, *Methoden und Ansätze für die Entwicklung und den Test*, Karlsruhe, 2018.
- [12] J. Bock, „The inD Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections,“ 2019. [Online]. Available: <http://arxiv.org/abs/1911.07602>.
- [13] W. Zhan, L. Sun, H. Shi, A. Clausse, M. Naumann, J. Kummerle, H. Königshof, C. Stiller, A. de La Fortelle und M. Tomizuka, „INTERACTION Dataset: An INTERnational, Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps,“ 2019. [Online]. Available: <https://arxiv.org/abs/1910.03088>.
- [14] M. Zipfl, T. Fleck, M. R. Zofka und J. M. Zöllner, „From Traffic Sensor Data To Semantic Traffic Descriptions: The Test Area Autonomous Driving Baden-Württemberg Dataset (TAF-BW Dataset),“ in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2020.

- [15] M. Zipfl und J. M. Zöllner, „Towards Traffic Scene Description: The Semantic Scene Graph,“ in *IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, Macau, 2022.
- [16] M. Zipfl, M. Jarosch und J. M. Zöllner, „Self Supervised Clustering of Traffic Scenes using Graph Representations,“ in *International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Laguna, 2022.
- [17] M. Zipfl, M. Jarosch und J. M. Zöllner, „Traffic Scene Similarity: a Graph-based Contrastive Learning Approach,“ in *IEEE Symposium Series on Computational Intelligence (SSCI)*, Mexico City, 2023.
- [18] International Organization for Standardization (ISO) 2009., „ISO 29119 - Software and systems engineering - software testin,“ 2009.
- [19] W. Campero, „Testkonzept IEEE 829 - Testmanagement nach ISTQB Standard,“ Qytera, 3 November 2019.
- [20] VVM, „Glossar VVM,“ VVM.
- [21] IEEE , „829-2008 - IEEE Standard for Software and System Test Documentation,“ IEEE, 2009.
- [22] F. Witte, Testmanagement und Softwaretest, Wiesbaden: Springer Verlag, 2019.
- [23] International Organization for Standardization, *ISO 29119 Software Testing*.
- [24] N. Hartmann, Test Automatisierung im Labor, Automatisiertes Testen Eingebetteter System in der Automobilindustrie, München: Carl Hanser Verlag, 2008.
- [25] C. E. Tuncal, G. Fainekos, D. Prokhorov und H. I. a. J. Kapinski, „Requirements-driven Test Generation for Autonomous Vehicles with Machine Learning Components,“ 2019. [Online]. Available: <https://arxiv.org/pdf/1908.01094.pdf>.
- [26] International Organization for Standardization, „Road vehicles - Functional safety, ISO/IEC/IEEE 26262-4:2011(E),“ 2011.
- [27] D. Zhao, Y. Guo und Y. J. Jia, „TrafficNet: An Open Naturalistic Driving Scenario Library,“ in *IEEE 20th International Conference on Intelligent Transportation Systems*, 2017.
- [28] V. L. Neale, T. A. Dingus, S. G. Klauer, J. Sudweeks und M. Goodman, *An overview of the 100-car naturalistic study and findings*, National Highway Traffic Safety Administration, Paper, no. 05-0400, 2005.
- [29] R. Ervin, J. Sayer, D. LeBlanc, S. Bogard, M. Hefford, M. Hagan, Z. Bareket und C. Winkler, „Automotive collision avoidance system field operational test report: methodology and results,“ Tech. Rep., 2005.

- [30] D. LeBlanc, „Road departure crash warning system field operational test: methodology and results. volume 1: technical report,“ 2006.
- [31] T. Victor, J. Bärghmann, M. Hjalmdahl, K. Kircher, E. Svanberg, S. Hurtig, H. Gellermann und F. Moeschlin, „Sweden-michigan naturalistic field operational test (semifot) phase 1: Final report,“ SAFER Report, vol. 2, 2010.
- [32] F. Schuldt, F. Saust, B. Lichte, M. Maurer und S. Scholz, „Effiziente systematische Testgenerierung für Fahrassistenzsysteme in virtuellen Umgebungen,“ in *Automatisierungssysteme, Assistenzsysteme und Eingebettete Systeme Für Transportmittel*, 2013.
- [33] H. Schittenhelm, „Testing of level-3 Systems,“ 2017. [Online]. Available: [https://www.pegasusprojekt.de/files/tmpl/Symposium-Vortraege/12 Testing%20of%20level-3%20Systems Schittenhelm.pdf](https://www.pegasusprojekt.de/files/tmpl/Symposium-Vortraege/12%20Testing%20of%20level-3%20Systems%20Schittenhelm.pdf).
- [34] E. d. Gelder und J.-P. Paardekooper, „Assessment of Automated Driving Systems Using Real-Life Scenarios,“ in *2017 IEEE Intelligent Vehicles Symposium, IV*, 2017.
- [35] A. Pütz, A. Zlocki, J. Bock und L. Eckstein, „System validation of highly automated vehicles with a database of relevant traffic scenarios,“ in *12th ITS European Congress*, 2017.
- [36] T. Menzel, G. Bagschik und M. Maurer, „Scenarios for Development, Test and Validation of Automated Vehicles,“ in *IEEE Intelligent Vehicles Symposium*, 2018.
- [37] B. Gerrit, M. Till, R. Andreas und M. Markus, „Szenarien für Entwicklung, Absicherung und Test von automatisierten Fahrzeugen,“ in *11. Workshop Fahrerassistenzsysteme*, 2017.
- [38] J. Bach, S. Otten und E. Sax, „Model based scenario specification for development and test of automated driving functions,“ in *IEEE Intelligent Vehicles Symposium (IV)*, Gothenburg, 2016.
- [39] Pegasus Projekt, „Pegasus Projekt,“ 13 05 2019. [Online]. Available: Die freigabe des autonomen fahrens. [Zugriff am 04 09 2020].
- [40] J. Broy, *Modellbasierte Entwicklung und Optimierung flexibler zeitgesteuerter*, Karlsruhe: Diss. Karlsruhe: Karlsruher Institut für, 2010.
- [41] S. Shalev-Shwartz, S. Shammah und A. Shashua, „On a Formal Model of Safe and Scalable Self-driving Cars,“ *Mobileye*, 2017.
- [42] C. King, „A Taxonomy and Survey on Validation Approaches for Automated Driving Systems,“ 2020.
- [43] C. Wang und H. Winner, „Overcoming Challenges of Validation Automated Driving and Identification of Critical Scenarios,“ in *IEEE Intelligent Transportation Systems Conference, ITSC*, 2019.

- [44] C. Wohlfahrt et al., „Von Systematischer Absicherung zur "Digitalen Erprobungsfahrt",“ in 6. *Fachkonferenz AUTOTEST*, Stuttgart, 2016.
- [45] C. King, L. Ries, C. Kober, C. Wohlfahrt und E. Sax, „Automated Function Assessment in Driving Scenarios,“ in *IEEE International Conference of Software Testing, Verification and Validation*, 2019.
- [46] J. E. Stellet, M. Woehrle, T. Brade, A. Poddey und W. Branz, „Validation of automated driving - structured analysis and survey of approaches,“ 2019.
- [47] J. Nilsson, *Computational verification methods for automotive safety systems*, Chalmers University of Technology, 2014.
- [48] M. R. Zofka, R. Kohlhaas, T. Schamm und J. M. Zöllner, „Semivirtual simulations for the evaluation of vision-based adas,“ *IEEE Intelligent Vehicles Symposium*, pp. 121-126, 2014.
- [49] Aptiv, Audi, BMW, Netcom Science, Continental Teves, Daimler AG, FCA, HERE, Infineon, Intel, Volkswagen, „Safety first for automated driving,“ 2019. [Online]. Available: <https://group.mercedes-benz.com/documents/innovation/other/safety-first-for-automated-driving.pdf>. [Zugriff am 06 2024].
- [50] M. Steimle, T. Menzel und M. Maurer, „A Method for Classifying Test Bench Configurations in a Scenario-Based Test Approach for Automated Vehicles,“ 2019.
- [51] R. Matthaei und M. Maurer, „Autonomous driving – a top-down-approach,“ at - *Automatisierungstechnik*, Bd. 3, Nr. 63, 2015.
- [52] F. Schuldt, „Ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen,“ 2017.
- [53] J. Stellet, „Testing of Advanced Driver Assistance Towards Automated Driving: A Survey and Taxonomy on Existing Approaches and Open Questions,“ 2015.
- [54] M. Zipfl, N. Koch und J. M. Zöllner, „A Comprehensive Review on Ontologies for Scenario-based Testing in the Context of Autonomous Driving,“ in *IEEE Intelligent Vehicles Symposium (IV)*, Anchorage, 2023.
- [55] C. King, L. Ries, J. Langner und E. Sax, *A Taxonomy and Survey on Validation Approaches for Automated*, Karlsruhe, 2020.
- [56] M. R. Zofka, T. Fleck, P. Schörner, M. Heinrich und J. M. Zöllner, „Paving the Ways for Vehicles-in-the-loop: Wireless Integration of Autonomous Vehicles into Mixed Reality Proving Grounds and Test Areas,“ in *IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, 2023.
- [57] F. Mütsch, H. Gremmelmaier, N. Becker, D. Bogdoll, M. R. Zofka und J. M. Zöllner, „From Model-Based to Data-Driven Simulation: Challenges and Trends in Autonomous Driving,“ in *IEEE Computer Vision and Pattern Recognition Conference*, Vancouver, 2023.

- [58] H. Gremmelmaier, L. Hierzegger, M. Sohail und J. M. Zöllner, „Cyclist behavior: a survey on characteristics and trajectory modeling,“ in *IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, 2023.
- [59] M. Zipfl, B. Schütt und J. M. Zöllner, „Scene-Extrapolation: Generating Interactive Traffic Scenarios,“ in *IEEE Intelligent Vehicle Symposium (IV)*, Jeju, 2024.
- [60] M. Zipfl, S. Spickermann und J. M. Zöllner, „Utilizing Hybrid Trajectory Prediction Models to Recognize Highly Interactive Traffic Scenarios,“ in *IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, 2023.
- [61] P. Schulz, A. Viehl und O. Bringmann, „Fast Environment Generation Methods for Virtual Testing,“ in *IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, 2023.
- [62] B. Schütt, M. Zipfl, J. M. Zöllner und E. Sax, „Fingerprint of a Traffic Scene: an Approach for a Generic and Independent Scene Assessment,“ in *International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Laguna, 2022.
- [63] B. Schütt, J. Ransiek, T. Braun und E. Sax, „1001 Ways of Scenario Generation for Testing of Self-driving Cars: A Survey,“ in *IEEE Intelligent Vehicles Symposium*, Anchorage, 2023.
- [64] B. Schütt, M. Zipfl, J. M. Zöllner und E. Sax, „Inverse Universal Traffic Quality - a Criticality Metric for Crowded Urban Traffic Scenes,“ in *IEEE Intelligent Vehicles Symposium (IV)*, Anchorage, 2023.
- [65] L. Westhofen, C. Neurohr, T. Koopmann, M. Butz, B. Schütt, F. Utesch, B. Neurohr, C. Gutenkunst und E. Böde, „Criticality Metrics for Automated Driving: A Review and Suitability Analysis of the State of the Art,“ in *Archives of Computational Methods in Engineering*, Online, SpringerLink, 01 August 2022, p. 1–35.
- [66] ASAM, „OpenSimulationInterface (OSI),“ [Online]. Available: <https://www.asam.net/standards/detail/osi/>. [Zugriff am 06 2024].
- [67] ASAM, „OpenSCENARIO,“ [Online]. Available: <https://www.asam.net/standards/detail/openscenario/v200/>. [Zugriff am 06 2024].
- [68] ASAM, „OpenLabel,“ [Online]. Available: <https://www.asam.net/standards/detail/openlabel/>. [Zugriff am 06 2024].
- [69] A. Puetz, A. Zlocki, J. Bock und L. Eckstein, „System validation of highly automated vehicles with a database of relevant,“ in *12th ITS European Congress*, 2017.
- [70] B. Schütt, M. Zhang, C. Steinhauser und E. Sax, „Evolutionary Behavior Tree Generation for Dynamic Scenario Creation in Testing of Automated Driving Systems,“ in *7th International Conference on System Reliability and Safety (ICSRS)*, Bologna, 2023.

Berichtsblatt

1. ISBN oder ISSN	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht		
3. Titel VVMethoden - Verifikations- und Validierungsmethoden automatisierter Fahrzeuge Level 4 und 5			
4. Autor(en) [Name(n), Vorname(n)] Otten, Stefan; Schwab, Stefan; Fuchs, Julian; Braun, Thilo; Zipfl, Maximilian; Gremmelmaier, Helen; Reiter, Sebastian; Schütt, Barbara; Schulz, Patrick; Padusinski, Hubert		5. Abschlussdatum des Vorhabens 31.12.2023	
		6. Veröffentlichungsdatum 30.06.2024	
		7. Form der Publikation Schlussbericht	
8. Durchführende Institution(en) (Name, Adresse) FZI Forschungszentrum Informatik Haid-und-Neu-Str. 10-14 76131 Karlsruhe		9. Ber. Nr. Durchführende Institution 1	
		10. Förderkennzeichen 19A19002L	
		11. Seitenzahl 88	
12. Fördernde Institution (Name, Adresse) Bundesministerium für Wirtschaft und Klimaschutz (BMWK) 53107 Bonn		13. Literaturangaben 70	
		14. Tabellen 5	
		15. Abbildungen 54	
16. Zusätzliche Angaben -			
17. Vorgelegt bei (Titel, Ort, Datum) -			
18. Kurzfassung Das Gesamtziel des Projektes VVMethoden war die Entwicklung von Systematik und Methoden für den Sicherheitsnachweis von vollautomatisierten und fahrerlosen Fahrfunktionen und Fahrzeugen Level 4/5 zur Homologation im urbanen Umfeld. Der Beitrag des FZI Forschungszentrum Informatik im Projekt VVMethoden hatte die folgenden Schwerpunkte: - Ansätze zum durchgängigen Testen mit verschiedenen Testinstanzen, von Simulation zu Realtests - Ableitung und Aufbereitung von realistischen Szenarien und Szenen für die Simulation - Testkonzepte zur durchgängigen Bewertung von automatisierten Fahrfunktionen, Übergreifende Bestimmung des Reifegrads von Funktionen - Methoden zur Identifikation und Aussagekraft von Szenarien - Durchgängige Verknüpfung und Abbildung von Szenarienbeschreibungen in virtuelle Testumgebungen - Evaluation von Testinfrastrukturen, Konzeption und Methodik für eine nahtlose Verknüpfung von Testkonfigurationen und Testinfrastrukturen („Seamless Testing“) sowie einer ganzheitlichen System- und Funktionsbewertung - Aufbau und Umsetzung verschiedener Testinstanzen und Demonstration im Bereich Simulation, Hardware-in-the-Loop (HiL)-Prüfstand sowie Versuchsträger			
19. Schlagwörter Absicherung, Test, Verifikation & Validierung, Automatisiertes Fahren, Prüfstand, Simulation			
20. Verlag -		21. Preis -	

Document Control Sheet

1. ISBN or ISSN	2. type of document (e.g. report, publication) Final report		
3. title VVMethoden - Verifikations- und Validierungsmethoden automatisierter Fahrzeuge Level 4 und 5			
4. author(s) (family name, first name(s)) Otten, Stefan; Schwab, Stefan; Fuchs, Julian; Braun, Thilo; Zipfl, Maximilian; Gremmelmaier, Helen; Reiter, Sebastian; Schütt, Barbara; Schulz, Patrick; Padusinski, Hubert		5. end of project 31.12.2023	
		6. publication date 30.06.2024	
		7. form of publication Final report	
8. performing organization(s) (name, address) FZI Forschungszentrum Informatik Haid-und-Neu-Str. 10-14 76131 Karlsruhe		9. originator's report no. 1	
		10. reference no. 19A19002L	
		11. no. of pages 88	
12. sponsoring agency (name, address) Bundesministerium für Wirtschaft und Klimaschutz (BMWK) 53107 Bonn		13. no. of references 70	
		14. no. of tables 5	
		15. no. of figures 54	
16. supplementary notes -			
17. presented at (title, place, date) -			
18. abstract The overall objective of the project VVMethods was the development of concepts and methods for the verification and validation of fully automated and driverless driving functions SAE level 4/5 vehicles in the urban environment. The contribution of the FZI Research Center for Information Technology in VVMethods had the following focal points: - Approaches to seamless testing with different test instances, from simulation to real tests - Derivation and preparation of realistic scenarios and scenes for simulation - Test concepts for the consistent evaluation of automated driving functions, comprehensive determination of the maturity level of functions - Methods for the identification and informative value of scenarios - Consistent linking and derivation of scenario descriptions for virtual test environments - Evaluation of test infrastructures, design and methodology for seamless combining different test configurations and test infrastructures as well as a holistic system and functional evaluation - Development and implementation of various test instances and demonstration in the area of simulation, Hardware-in-the-loop (HiL) test bench and test vehicle			
19. keywords Verification, Validation, Test, Automated Driving, Test Infrastructure, Test Bench, Simulation			
20. publisher -		21. price -	