
Eignungsprüfung der OpenSource-Simulationssoftware CARLA zur Entwicklung von Szenarien zur Evaluation von kamerabasierten Fahrerassistenzsystemen

Bachelorarbeit zur Erlangung des Bachelor-Grades
Bachelor of Science im Studiengang Informatik
an der Fakultät für Informatik und Ingenieurwissenschaften
der Technischen Hochschule Köln

vorgelegt von: Jan Gellrich
Matrikel-Nr.: 11126375
Adresse: Gohrer Weg 19
50769 Köln
jan.gellrich@outlook.de

eingereicht bei: Prof. Dr. Wolfgang Konen
Zweitgutachter: Daniel Fischer

Köln, 30.05.2022

Erklärung

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer oder der Verfasserin/des Verfassers selbst entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Anmerkung: In einigen Studiengängen steht die Erklärung am Ende des Textes.

Ort, Datum

Rechtsverbindliche Unterschrift

Kurzfassung

Die Bachelorarbeit baut auf dem vorhergehenden Praxisprojekt „*Aufsetzung einer OpenSource Simulationsumgebung zum Testen von optischen Sensoren*“ auf. In diesem wurde nach einer geeigneten Simulationsumgebung für das Testen von Fahrzeugsensoren mittels Videomaterial gesucht, welche das vom BSI genutzte System namens „IPG CarMaker“ ersetzen soll.

In dieser Bachelorarbeit wird die ausgewählte Software mit der bisherigen Simulationsumgebungen verglichen, um Unterschiede festzustellen und die neue Lösung zu bewerten. Die Resultate des Vergleichs werden genutzt, um anschließend mit der neuen Simulationsumgebung neue Artefakte zur Evaluation von kamerabasierten Fahrzeugsensoren zu erstellen.

Die in der Abschlussarbeit gewählte männliche Form bezieht sich immer zugleich auf weibliche und männliche Personen.

Schlagwörter

ACC: adaptive Fahrgeschwindigkeitsregelung (Adaptive Cruise Control).

BSI: Bundesamt für Sicherheit in der Informationstechnik

CAN: Controller Area Network

LiDAR: Light Detection and Ranging

Inhalt

Erklärung	I
Kurzfassung	II
Tabellenverzeichnis	V
Abbildungsverzeichnis	VI
1 Einleitung	1
1.1 Motivation	1
1.2 Problemstellung	1
1.3 Zielsetzung	2
1.3.1 Vergleich der Simulationen	2
1.3.2 Erstellung neuer Testszenarien	3
2 Stand der Technik	4
2.1 Fahrassistenzsysteme und autonomes Fahren	4
2.2 Kamera-Sensoren	5
2.3 Simulationsumgebungen	6
3 Testsystem	8
3.1 Systemfunktionen	8
3.1.1 FCW / UFCW – (Urban) Forward Collision Warning	9
3.1.2 PCW – Pedestrian Collision Warning	9
3.1.3 LDW – Lane Departure Warning	9
3.1.4 HMW – Headway Monitoring Warning	9
3.1.5 SLI / TSR – Speed Limit Indication / Traffic Sign Recognition	10
3.2 Kalibrierung und benötigte Systemdaten	10
4 Erstellung eines Szenarios	12
4.1 Erzeugung des Straßennetzes	12
4.2 Bearbeitung in der Unreal Engine	12
4.3 Simulation	13
5 Versuchsaufbau und Vorgehensweise	16
5.1 Vorgehensweise	16
5.1.1 Vorgehen beim Vergleich	16
5.2 Versuchsaufbau	16
6 Vergleich zwischen IPG CarMaker und CARLA	18
6.1 Testablauf	18
6.2 Vergleich und Resultate	18

6.2.1 Fußgänger überquert Fahrbahn	19
6.2.2 Kollision auf Autobahn bei Nebel	20
6.2.3 Fahrzeug mit himmelsähnlichem Farbton	22
6.2.4 Hindernisse/Schilder aufgemalt auf der Fahrbahn.....	23
6.2.5 Straßenschilder auf Häuserwänden	26
6.2.6 Fahrt hinter einem Fahrradfahrer	28
6.2.7 Verkehrsschild bei Nebel	30
6.2.8 Verkehrsschild bei Nacht	32
6.2.9 Verkehrsschild bei Tag.....	33
6.2.10 Fazit des Vergleiches.....	34
7 Testen von neuen Szenarien.....	36
7.1 Änderungen	36
7.2 Kollisionswarnungen	36
7.2.1 Testhindernisse	36
7.2.2 Szenario bei Mittagssonne	37
7.2.3 Szenario bei Nebel.....	40
7.2.4 Szenario bei Nacht.....	42
7.2.5 Fazit der Kollisionserkennung	43
7.3 Schilderkennung	44
7.3.1 Testschilder	44
7.3.2 Schildtest bei Tag	44
7.3.3 Schildtest bei Nebel	45
7.3.4 Schildtest bei Nacht	46
7.3.5 Fazit Schilderkennung	46
8 Fazit und Ausblick	48
Literaturverzeichnis	49
Anhang.....	52

Tabellenverzeichnis

Tabelle 1: Funktionen der Mobileye Sensoren.....	10
Tabelle 2: Auflistung der zu vergleichenden Szenarien	18
Tabelle 3: Ergebnisse des Tests: Fußgänger überquert die Fahrbahn.	20
Tabelle 4: Ergebnisse des Tests: Kollision bei Nebel.	21
Tabelle 5: Ergebnisse des Tests: Fahrzeug mit himmelsähnlichem Farbton.....	23
Tabelle 6: Ergebnisse des veränderten Szenarios	23
Tabelle 7: Auf dem Boden gemalte Texturen aus dem Test „Hindernisse/Schilder aufgemalt auf der Fahrbahn“	24
Tabelle 8: Erkennung Tempo 30 Zone.	25
Tabelle 9: Erkennung Fußgänger stehend.	26
Tabelle 10: Erkennung Autorückseite	26
Tabelle 11: Fußgänger laufend.....	26
Tabelle 12: Erkennung beider Schilder.....	27
Tabelle 13: Erkennung des Fahrradfahrers	29
Tabelle 14: Erkennung des Fahrradfahrers mit neuer Aufnahme.....	30
Tabelle 15: Schild im Nebel.....	31
Tabelle 16: Schild bei Nacht.....	33
Tabelle 17: Schild bei Tag	34
Tabelle 18: Auflistung der Hindernisse für die neuen Testfälle	36
Tabelle 19: Szenario „Kollisionstest bei Tag“	39
Tabelle 20: Szenario „Kollisionstest bei Nebel“	41
Tabelle 21: Szenario „Kollisionstest bei Nacht“	43
Tabelle 22: Auflistung der Schilder für die neuen Testfälle	44
Tabelle 23: Szenario „verschmutzte Schilder bei Tag“	45
Tabelle 24: Szenario „verschmutzte Schilder bei Nebel“	45
Tabelle 25: Szenario „verschmutzte Schilder bei Nacht“	46

Abbildungsverzeichnis

Abbildung 1 Sicherheits- und Komfortfunktionen auf der Basis von Rundumsichtsensoren	4
Abbildung 2: SAE Level der Fahrautomatisierung	5
Abbildung 3: Wegpunkte der OpenDRIVE-Datei in der Unreal Engine dargestellt	6
Abbildung 4: Display Einheit des Mobileye-Systems	8
Abbildung 5: Ausgelesenes Bild der Mobileyekamera beim Kalibrierungsvorgang ...	11
Abbildung 6: Übersicht der Roadrunner Oberfläche	12
Abbildung 7: Herstellerangaben zu dem Mercedes C-Klasse Model in Metern	13
Abbildung 8: Darstellung der Simulationsumgebung und ihrer Einzelteile	14
Abbildung 9: Neu erzeugtes Fenster.	15
Abbildung 10: Zeichnung des Versuchsaufbaus im Labor	17
Abbildung 11: Ausschnitt des CARLA-Videos „Fußgänger“	19
Abbildung 12: Ausschnitt des IPG-Videos „Fußgängerin“	20
Abbildung 13: Ausschnitt des CARLA-Videos „Nebel Kollision“	21
Abbildung 14: Ausschnitt des IPG-Videos „Nebel Kollision“	21
Abbildung 15: Ausschnitt des CARLA-Videos „Hindernis mit Himmelfarbtön“	22
Abbildung 16: Ausschnitt des IPG-Videos „Überlagerung Himmel LKW“	22
Abbildung 17: Ausschnitt des CARLA-Videos „Hindernis mit Himmelfarbtön Neu“ ...	23
Abbildung 18: Fußgängertexturen, die für das CARLA-Video verwendet werden	24
Abbildung 19: Tempo 30 Zone	24
Abbildung 20: Autorückseite, die im CARLA-Video verwendet wird	24
Abbildung 21: Ausschnitt des CARLA-Videos „Roadpaintings“	25
Abbildung 22: Ausschnitt des IPG-Videos "Roadpaintings"	25
Abbildung 23: Ausschnitt des CARLA-Videos „Schild auf Haus“	27
Abbildung 24: Ausschnitt des IPG-Videos "Schild auf Haus"	27
Abbildung 25: Ausschnitt des CARLA-Videos „Fahrradfahrer“	28
Abbildung 26: Ausschnitt des IPG-Videos "Fahrradfahrer"	29
Abbildung 27: Ausschnitt des CARLA-Videos „Fahrradfahrer Video 2“	30
Abbildung 28: Ausschnitt des CARLA-Videos „Schild im Nebel“	31
Abbildung 29: Ausschnitt des IPG-Videos „Schild im Nebel“	31

Abbildung 30: Ausschnitt des CARLA-Videos „Schild bei Nacht“	32
Abbildung 31: Ausschnitt des IPG-Videos „Schild bei Nacht“	32
Abbildung 32: Ausschnitt des CARLA-Videos „Schild bei Tag“	33
Abbildung 33: Ausschnitt des IPG-Videos „Schild bei Tag“	34
Abbildung 34: Die 9 Testobjekte bei Tag	38
Abbildung 35: Die 9 Testobjekte bei Nebel	40
Abbildung 36: Die 9 Testobjekte bei Nacht	42
Abbildung 37: Verschmutzte Schilder bei Tag	44
Abbildung 38: Verschmutzte Schilder bei Nebel	45
Abbildung 39: Verschmutzte Schilder bei Nacht	46

1 Einleitung

1.1 Motivation

In den letzten Jahren gab es in der Automobilindustrie zwei wesentliche Themen, mit dem Potential, den Straßenverkehr zu revolutionieren. Zum einen ist dies die Elektrifizierung des Straßenverkehrs, um dem Klimawandel entgegenzuwirken. Auf der anderen Seite steht die Autonomisierung des Straßenverkehrs. Ziel der Forschung und Entwicklung in diesem Bereich ist unter anderem das Verringern von Verkehrsunfällen und die Möglichkeit, Personengruppen einzubinden, die nicht ohne Unterstützung am Straßenverkehr teilnehmen können [1]. Weitere Fahrhilfen werden ständig hinzugefügt oder erweitert, und schon heute sind viele Fahrzeuge mit umfangreichen Assistenzsystemen ausgestattet, die den Fahrer unterstützen. Somit wird der Verkehr schrittweise autonomer.

Damit Fahrzeuge autonom fahren oder Assistenzsysteme den Fahrer unterstützen können, benötigen sie Informationen über die unmittelbare Umgebung. Moderne Fahrzeugsensoren können komplexe Situationen erfassen und die gewonnenen Informationen an die Steuergeräte weitergeben. Diese senden dementsprechend Fahrbefehle an Aktoren, welche den Befehl in Aktionen wie akustische Signale umwandeln. Informationen beinhalten beispielsweise die erkannten Fußgänger und Verkehrsschilder. Im Falle eines Hindernisses wird eine potenzielle Kollision von den optischen Sensoren erkannt und durch das Steuersystem wird ein Bremsbefehl ausgeführt.

Die Technik bringt jedoch auch einige Risiken mit sich. „Durch die Integration [von] Sensoren in das Fahrzeug werden neue Informationskanäle in das Fahrzeug hinein geschaffen. Diese neuen Informationskanäle bieten Angreifern potenziell die Möglichkeit, in die Steuerung der Fahrzeuge einzugreifen.“ [2]

2019 untersuchte das Tencent Keen Security Lab einen Tesla Model S 75 auf Schwachstellen im Advanced Driver Assistance System (kurz: ADAS). Die Untersuchung ergab unter anderem, dass die Spur-Erkennung des Tesla Autopiloten mit nachträglich eingefügten Fahrbahnmarkierungen überlistet werden konnte. Auf die Fahrbahn wurden drei quadratische Sticker geklebt, die das Fahrzeug als Leitlinie erkannte. Den Angreifern gelang es damit, den Autopiloten des Teslas auf die Gegenfahrbahn zu lenken [3].

Aufgabe des „Bundesamt für Sicherheit in der Informationstechnik“ (kurz: BSI) ist unter anderem das Finden von Schwachstellen in solchen Assistenzsystemen.

1.2 Problemstellung

Um Automotive-Sensorik zu testen, werden im BSI unter anderem Simulationen eingesetzt. Diese bieten den Vorteil einer guten Reproduzierbarkeit gegenüber realen Testversuchen. Tests können mehrfach durchgeführt werden, ohne dass Abweichungen im Szenario auftreten. Dadurch können verschiedene Sensoren bei exakt gleicher Situation

anhand ihrer Reaktion verglichen werden. Zusätzlich können viele Parameter eines Tests beliebig und mit geringem Aufwand geändert werden.

Die Bachelorarbeit fokussiert sich auf optische Sensoren zum Erfassen des Straßenverkehrs vor dem Fahrzeug. Im Labor wird der Sensor mit einem festgelegten Abstand vor einen Bildschirm positioniert. Auf dem Bildschirm wird ein aufgezeichnetes Szenario abgespielt. Das entsprechende Video zeigt das Szenario aus der Sicht einer Dashcam. Für einen erfolgreichen Test muss der Sensor nun das im Szenario getestete Ereignis (bspw. Fußgänger, Straßenschilder) erkennen.

Aktuell wird für das Erstellen dieser Szenarien eine Simulationslösung der Firma IPG Automotive verwendet. Die Software „CarMaker“ ist eine „offene Integrations- und Testplattform“ für das Entwickeln von Hard- und Softwarelösungen im Automotive-Bereich und ermöglicht das Erstellen von realistischen Testszenarien [4]. Mit Hilfe des Tools können virtuelle Modellumgebungen erstellt werden, in denen dann verschiedenste Versuche durchgeführt werden können. Dabei werden beispielsweise reale Straßendaten eingelesen und in die Testumgebung eingefügt. Diese kann dann weiter mit 3D-Objekten wie Häusern und anderen Autos angereichert werden. Um Sensoren im Verkehr zu testen, werden andere Verkehrsteilnehmer simuliert, wie beispielsweise Fußgänger, die die Straße überqueren.

Auf dem Markt gibt es aber auch OpenSource Alternativen, welche frei zugänglich sind. Somit kann die Software bei Bedarf angepasst oder verändert werden. Um die alte Software zu ersetzen, soll die Alternative das Erstellen von neuen Umgebungen und Szenarien unterstützen. Dafür muss die neue Umgebung Szenarien aus der Sicht des Sensors darstellen können, um diesen möglichst realitätsnah zu testen.

1.3 Zielsetzung

In dem vorherigen Praxisprojekt [5] wurde erarbeitet, welche OpenSource Umgebung sich eignen könnte, um das momentane System zu ersetzen. Diese muss Testszenarien aus dem Straßenverkehr realistisch genug darstellen, um den Sensor im Labor testen zu können. Als Ergebnis des Projektes wurde sich für die Simulationsumgebung CARLA entschieden. CARLA ist ein OpenSource Simulator, der für die Entwicklung von autonomen Fahrzeugsystemen entwickelt wurde [6].

Ziel der Abschlussarbeit ist die Eignungsprüfung der OpenSource-Simulationsumgebung CARLA. Dabei gibt es zwei Schwerpunkte, die untersucht werden:

1.3.1 Vergleich der Simulationen

Um herauszufinden, ob die neue Simulationsumgebung die alte ablösen kann, werden diese direkt verglichen. Hierfür werden absolvierte Tests [7] mit der neuen Umgebung nachgebaut. Es wird geprüft, ob die neue Umgebung als Simulation für den Sensor funktioniert. Die alten Tests dienen dementsprechend als Referenz für mögliche Szenarien. Bei Unstimmigkeiten muss untersucht werden, woher diese stammen und ob es nötig ist, diese auszugleichen.

Folgende Vorgehensweise wurde gewählt:

- Aus den bestehenden Szenarien werden vorhandene Tests für den Vergleich ausgewählt. Priorität haben vor allem gängige Szenarien, die die Erkennungsmechanismen der Kamera testen.
- Die vorhandenen Tests werden in der neuen Umgebung rekonstruiert. Um die beiden Systeme vergleichen zu können, ist es wichtig, dass die in der Testbeschreibung vorhandene Situation möglichst genau nachgebaut wird. Dekorative Objekte wie Hauswände sind allerdings vernachlässigbar, sofern sie nicht essenzieller Teil des Szenarios sind.
- Anschließend werden Aufnahmen für die neuen Tests erzeugt und zu Videos konvertiert. Die neu erstellten Videos werden mit alten Videos des IPG CarMakers verglichen.
- Die erstellten Videos werden dem System vorgeführt. Dabei wird verglichen, in welchen Szenarien die neue Umgebung besser oder auch schlechter ist. Es wird als Erfolg gewertet, wenn die neue Umgebung die Sensoren ausreichend testen kann.
- Bei signifikanten Unterschieden wird eine Abweichungsanalyse erstellt, um die Ursachen zu ermitteln. Wichtig bleibt allerdings, dass das neue System einsatzbereit ist.

1.3.2 Erstellung neuer Testszenarien

Mit der neuen Simulationsumgebung werden nun Testszenarien umgesetzt, die bisher nicht möglich waren oder noch nicht in der alten Umgebung umgesetzt wurden. Neue Testszenarien beinhalten beispielsweise die Erkennung von Straßenschildern unter verschiedenen Wetterbedingungen, Lichtverhältnissen und Tageszeiten.

Folgendes Vorgehen wird gewählt:

- Zuerst müssen hierfür mögliche Schwachstellen erörtert werden. Zusätzlich muss die Umsetzbarkeit für das Erstellen von Testszenarien berücksichtigt werden. In der neuen Umgebung gibt es beispielsweise nicht die Möglichkeit, Fahrzeuge eigenständig Hindernisse umfahren zu lassen. Aus den möglichen Schwachstellen müssen neue Testbeschreibungen und Anforderungen erstellt werden.
- Für die neuen Testbeschreibungen werden entsprechende Testkarten für jedes Szenario gebaut. Alternativ wird eine Karte erstellt, die für mehrere Szenarien geeignet ist. Sobald die Umgebung fertig ist, können neue Aufnahmen für die Sensortests aufgezeichnet werden.
- Die Tests werden mit den erstellten Aufnahmen durchgeführt. Im Anschluss werden die Ergebnisse ausgewertet.

2 Stand der Technik

Die meisten Fahrzeuge besitzen heutzutage unterschiedlichste Informationssysteme und Assistenten, die dem Fahrer bei der Steuerung des Fahrzeuges unterstützen. Nicht jedes System greift auch in die Steuerung des Fahrzeuges ein. Der Grad der Unterstützung unterscheidet sich dabei, wie Abbildung 1 zeigt.

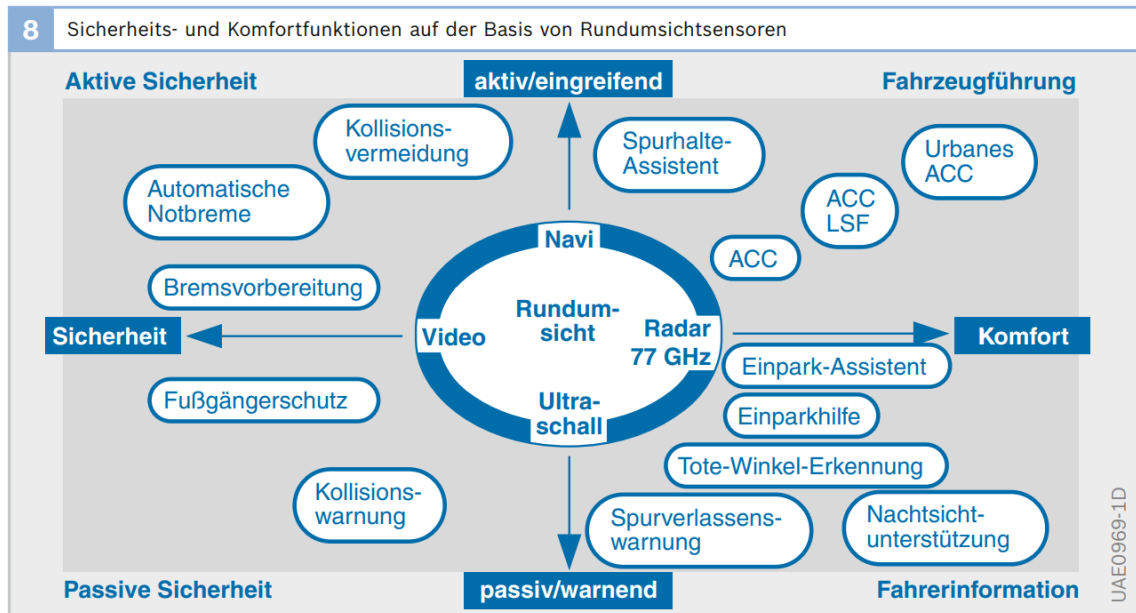


Abbildung 1 Sicherheits- und Komfortfunktionen auf der Basis von Rundumsichtsensoren [8]

2.1 Fahrassistenzsysteme und autonomes Fahren

Laut der „*internationale Society of Automotive Engineers*“, kurz „SAE International“, wird beim autonomen Fahren zwischen unterschiedlichen Leveln von autonomem Fahren unterschieden. Die Skala reicht von Level 0 (keine Automatisierung) bis Level 5 (vollständige Fahrautomatisierung) [9]. Dabei wird zwischen Funktionen unterschieden, die den Fahrer lediglich unterstützen, und jenen, die das Steuern des Fahrzeuges übernehmen.



SAE J3016™ LEVELS OF DRIVING AUTOMATION™

Learn more here: [sae.org/standards/content/j3016_202104](https://www.sae.org/standards/content/j3016_202104)

Copyright © 2021 SAE International. The summary table may be freely copied and distributed AS-IS provided that SAE International is acknowledged as the source of the content.

	SAE LEVEL 0™	SAE LEVEL 1™	SAE LEVEL 2™	SAE LEVEL 3™	SAE LEVEL 4™	SAE LEVEL 5™
What does the human in the driver's seat have to do?	You are driving whenever these driver support features are engaged – even if your feet are off the pedals and you are not steering			You are not driving when these automated driving features are engaged – even if you are seated in “the driver's seat”		
	You must constantly supervise these support features; you must steer, brake or accelerate as needed to maintain safety			When the feature requests, you must drive	These automated driving features will not require you to take over driving	
Copyright © 2021 SAE International.						
What do these features do?	These are driver support features			These are automated driving features		
	These features are limited to providing warnings and momentary assistance	These features provide steering OR brake/acceleration support to the driver	These features provide steering AND brake/acceleration support to the driver	These features can drive the vehicle under limited conditions and will not operate unless all required conditions are met	This feature can drive the vehicle under all conditions	
Example Features	• automatic emergency braking • blind spot warning • lane departure warning	• lane centering OR • adaptive cruise control	• lane centering AND • adaptive cruise control at the same time	• traffic jam chauffeur	• local driverless taxi • pedals/steering wheel may or may not be installed	• same as level 4, but feature can drive everywhere in all conditions

Abbildung 2: SAE Level der Fahrautomatisierung [9]

Um den Fahrer zu unterstützen, bekommt das Fahrzeug Informationen über die Umwelt durch eine Vielzahl an Kamera-, LiDAR- und Radarsensoren. In dieser Bachelorarbeit geht es ausschließlich um Kamera-Sensoren.

2.2 Kamera-Sensoren

Es gibt verschiedenste Kamera-Sensoren mit unterschiedlichen Aufgaben im Fahrzeug. Besonders wichtig sind Frontview-Kameras. Diese werden häufig am Rückspiegel im Innenraum des Fahrzeuges angebracht und überblicken die Straße vor dem Fahrzeug. Die Position ist vorteilhaft aufgrund ihres breiten Sichtfeldes und dem Schutz der Windschutzscheibe [10]. Die Frontview-Kameras sollen Hindernisse erkennen und den Fahrzeugführer unterstützen. Je autonomer ein Fahrzeug wird, desto besser müssen die Frontview-Kameras funktionieren, da diese die Funktion des menschlichen Auges übernehmen. Die meisten Assistenzentscheidungen werden anhand der Bilder der Frontview-Kamera getroffen.

Ein weiterer Kamera-Sensor ist der Einparkassistent. Dank einer Kamera am Heck kann der Fahrer die Bordsteinkante und anderen Hindernissen hinter dem Auto besser sehen und Distanzen einschätzen.

NIR/FIR-Kameras (Near Infrared/Far Infrared) können hingegen schlecht durch Glas sehen, und müssen am Kühlergrill oder in den Scheinwerfern befestigt werden. Diese NIR/FIR-Kameras werden als Nachtsichtsystem eingesetzt. Zwar können die meisten Hindernisse in der Nacht mit Fernlicht erkannt werden, doch Fernlicht ist nicht in jeder Situation einsetzbar. NIR-Kameras untersuchen die Umgebung, indem diese mit

infrarotem Licht bestrahlt wird. So wird verhindert, dass der Gegenverkehr geblendet wird, da die Strahlung nicht für das menschliche Auge sichtbar ist. FIR-Kameras sind in der Lage, selbst bei geringen Lichtverhältnissen noch Fußgänger anhand ihrer ausgestrahlten Wärme zu erkennen [11]. Der Großteil aller Kamera-Sensoren arbeitet jedoch im sichtbaren Spektralbereich.

2.3 Simulationsumgebungen

Simulationsumgebungen werden mittlerweile vermehrt in der Autoindustrie zur Entwicklung von neuen Fahrzeugen und Assistenzsystemen verwendet. So werden bei BMW beispielsweise nur 5% der Test-Distanz mit Fahrzeugen auf der Straße zurückgelegt. Die restlichen 95% finden virtuell statt [12].

Simulationsumgebungen bestehen aus mehreren Einzelteilen. Für die grafische Darstellung sowie physikalische Berechnungen wie bspw. Kollisionen benötigen sie eine Engine. Engines sind Softwaretools für komplexe Berechnungen oder Simulationen. Die Engine bildet die Grundlage der simulierten Welt. Jede Engine hat ihre eigene Art und Weise, Objekte und ihre Interaktionen darzustellen. Eigentlich werden Engines für die Entwicklung von Computerspielen verwendet, aber die Funktionen der Engines sind auch in anderen Bereichen nützlich.

Damit sich Fahrzeuge und Fußgänger in der Testumgebung zurechtfinden, benötigen sie Wegpunkte, welche sie nacheinander abfahren oder gehen. Für diese Wegpunkte wird als Standard meist OpenDRIVE [13] verwendet. Eine OpenDRIVE-Datei beinhaltet sämtliche Navigationsdaten für den simulierten Verkehr. Zur Erstellung und Bearbeitung der Dateien sind spezielle Editoren notwendig. Das BSI nutzt momentan zu diesen Zwecken die Software Roadrunner [14]. Mit dieser Software können Straßennetze erstellt werden, die anschließend in verschiedenste Simulationsumgebungen importiert werden können.

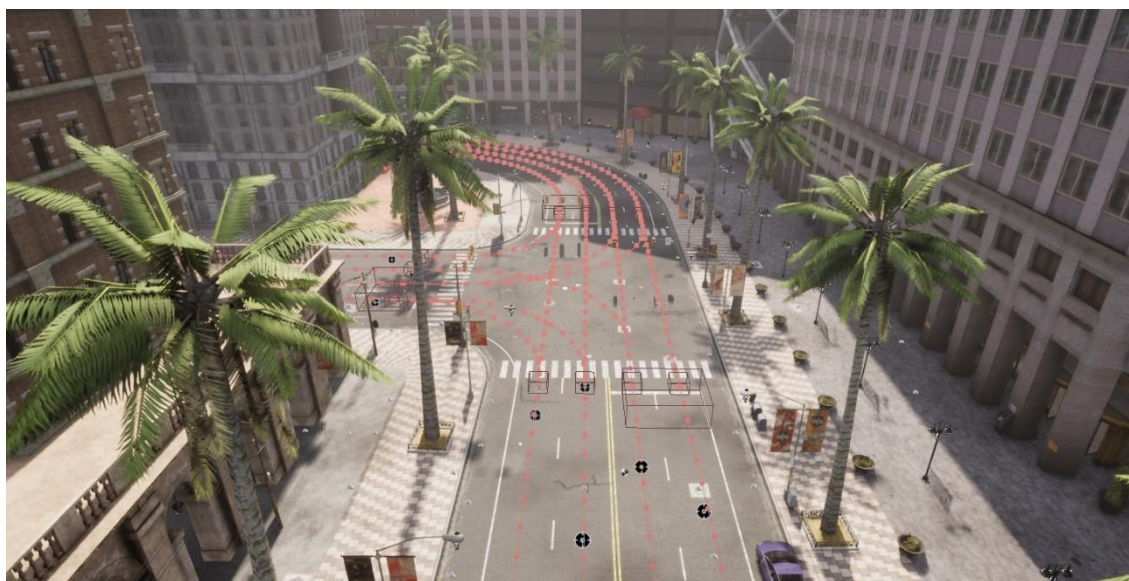


Abbildung 3: Wegpunkte der OpenDRIVE-Datei in der Unreal Engine dargestellt

Die Simulationsumgebung CARLA [6] basiert auf der Unreal Engine [15]. Diese zeichnet sich durch ihre realistische Darstellung von Umgebungen aus.

CARLA wurde entwickelt, um die Erstellung, das Training und Testen autonomer Fahrsysteme zu unterstützen [6]. Zusätzlich werden mit der Software einige Testumgebungen und Testszenarien mitgeliefert, sowie dutzende Fahrzeugmodelle. Der Code ist über GitHub frei zugänglich. Die Simulationsplattform unterstützt die flexible Anpassung von Umgebungsbedingungen und Sensoren.

CARLA basiert auf einer Client-Server Architektur. Die Serverseite ist zuständig für die Simulation, während auf der Clientseite z.B. die Logik für Wetterkonditionen und das Verhalten von Akteuren läuft.

3 Testsystem

Die Testumgebung im Labor wurde auf Basis des Mobileye 630 PRO Systems [18] [19] aufgesetzt. Die Mobileye 6 Systeme benutzen eine digitale Kamera, die auf der Innenseite einer Windschutzscheibe angebracht ist. Diese Systeme sind als zusätzliche eigenständige Module für Autos ohne Kamera verfügbar. Zusätzlich werden sie aber auch an Fahrzeughersteller verkauft und in deren Fahrzeugsystemen als Chip integriert.

Die Mobileye-Kameras sind in der Lage, Bilddaten in Echtzeit zu verarbeiten. Mittels der verarbeiteten Daten können dynamische Abstände effektiv zwischen Fahrzeug und Objekten berechnet werden. Relevante Daten werden über ein Display an den Fahrer weitergeleitet. [20] Das System bewertet bei der Verarbeitung auch, ob die Daten relevant für den Fahrer sind. Ein Straßenschild, welches auf einer Abbiegespur erkannt wird, soll herausgefiltert werden, wenn der Fahrer nicht auf dieser Spur fährt.



Abbildung 4: Display Einheit des Mobileye-Systems [21]

Die Kamera besitzt einen Blickwinkel von 40 x 30 Grad (Weite x Höhe) und erkennt Fahrzeuge typischerweise auf 80 Meter Entfernung. Die Auflösung der Linse beträgt 752x480 Pixel. [22]

Der Chip ist in der Lage, maximal 15 Bilder pro Sekunde zu verarbeiten. [20]

3.1 Systemfunktionen

Die Mobileye 6 Systeme besitzen eine Vielzahl an Funktionen. Einige können sogar in die Steuerung des Systems eingreifen, wie beispielsweise die „Lane Keeping and Guidance Assist“ (kurz: LKA). Die LKA-Applikation wirkt aktiv auf das Lenkrad ein, um das Fahrzeug in der Spur zu halten. Die meisten Funktionen sind aktiv, sobald das System angeschaltet wurde. Allerdings sind einige Funktionen aber auf bestimmte Geschwindigkeitsbereich beschränkt. Das kann dazu führen, dass die Funktion zwar aktiv ist, aber nicht operiert. In diesem Fall werden von der Funktion keine Signale ausgegeben.

Da die Szenarien vorab aufgezeichnet werden, können nur Assistenzsysteme getestet werden, die nicht in die Steuerung eingreifen. Es werden folgende Funktionen unterstützt [20]:

3.1.1 FCW / UFCW – (Urban) Forward Collision Warning

Die FCW-Funktion warnt den Fahrer vor einer bevorstehenden Auffahrkollision mit einem Fahrzeug.

FCW kann den Fahrer bis zu 2.7 Sekunden vor einer Kollision warnen. Die Funktion ist aktiv, solange das System eingeschaltet ist und gilt für Geschwindigkeiten von 30 – 200 km/h.

Sollte das Fahrzeug weniger als 30 Km/h fahren, wird auf „Urban Forward Collision Warning“ umgestellt. UFCW erfüllt die gleichen Aufgaben wie FCW, ist aber für niedrigere Geschwindigkeiten optimiert.

3.1.2 PCW – Pedestrian Collision Warning

PCW warnt den Fahrer vor einer bevorstehenden Kollision mit einem Fußgänger oder Fahrradfahrer. Eine Warnung kann bis zu 2 Sekunden vor der eventuellen Kollision ausgegeben werden. Solange das Fahrzeug langsamer als 50 Km/h fährt, ist PCW aktiv.

3.1.3 LDW – Lane Departure Warning

LDW warnt den Fahrer vor einem unbeabsichtigten Verlassen der Fahrspur. Dabei werden die Abstände der Fahrbahnmarkierungen zu den Vorderädern des Fahrzeuges gemessen. LDW ist von LKA abgegrenzt.

LDW ist aktiv, solange das System angeschaltet ist. Dabei operiert das System aber nur bei Geschwindigkeiten über 55 km/h.

LDW gibt keine Warnungen aus, wenn der Blinker des Fahrzeuges aktiv ist, die Fahrbahn nicht ausreichend markiert ist oder bei scharfen Kurven. Zusätzlich wird nach einer LDW-Warnung bis zur Rückkehr auf die Fahrbahnmitte keine weitere Warnung ausgegeben.

Die Funktion ist deaktiviert, wenn die Fahrbahnmarkierungen enger als die Fahrzeugbreite + 15% sind. Liegt die Spurgröße zwischen Fahrzeugbreite + 15% und + 30%, so wird eine Warnung erst bei Verlassen der Spur ausgegeben. Bietet die Spur mehr Platz als Fahrzeugbreite + 30%, so wird eine Warnung bei einem bevorstehenden Verlassen der Spur ausgegeben.

3.1.4 HMW – Headway Monitoring Warning

HMW hilft dem Fahrer, den Abstand zu einem vorrausfahrenden Fahrzeug einzuschätzen. Dabei wird der Abstand zur Position des vorrausfahrenden Fahrzeuges in einem Display angezeigt. Fällt dieser unter einen vorbestimmten Schwellwert, wird eine Warnung ausgegeben. Der Abstand wird in Sekunden gemessen, ist aber nicht

gleichzusetzen mit der Zeit bis zu einer Kollision. HMW operiert ab einer Geschwindigkeit von mehr als 30 km/h.

3.1.5 SLI / TSR – Speed Limit Indication / Traffic Sign Recognition

Verschiedene Verkehrsschilder werden erkannt und klassifiziert. Der Fahrer wird benachrichtigt, wenn er die erkannte Geschwindigkeit überschreitet.

Tabelle 1 zeigt, zu welchen Geschwindigkeiten welche Funktionen aktiv sind.

Geschwindigkeit	0-30 km/h	30-50 km/h	50-55 km/h	55+ km/h
FCW	x	✓	✓	✓
UFCW	✓	x	x	x
PCW	✓	✓	x	x
LDW	x	x	x	✓
HMW	x	✓	✓	✓
SLI	✓	✓	✓	✓
TSR	✓	✓	✓	✓

Tabelle 1: Funktionen der Mobileye Sensoren

3.2 Kalibrierung und benötigte Systemdaten

Ursprünglich wurde das Mobileye für Tests mit der IPG CarMaker Simulation kalibriert. Die Kamera wurde zur Einstellung zuerst extern mit einem Opel Astra Sports Tourer+ (Baujahr 2019) kalibriert [7]. Es ergaben sich folgende Werte für die Kamera:

- Kamerahöhe: 1,00 m
- Abstand zur Stoßstange: 1,52 m
- Fahrzeugbreite: 1,68 m
- Abstand der Kamera zum linken Scheibenrand: 0,84 m
- Abstand der Kamera zum rechten Scheibenrand: 0,84 m

Anschließend wurde die Kamera mit einem Szenario noch einmal automatisch kalibriert.

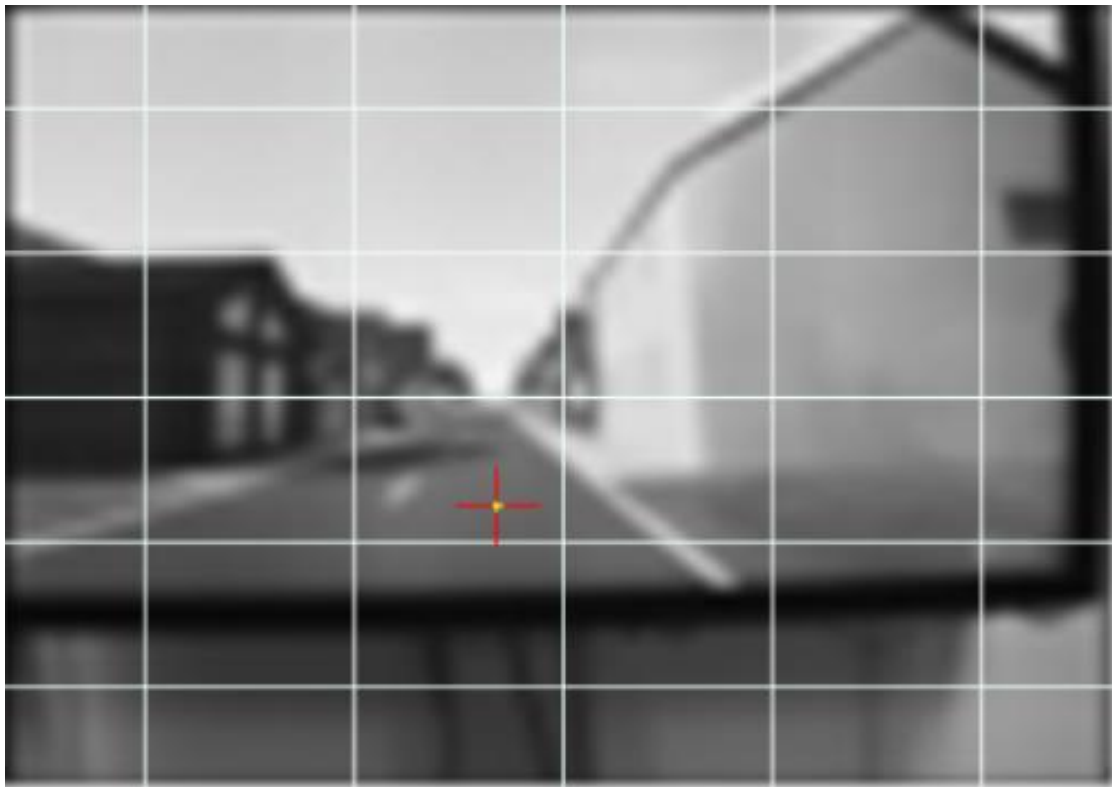


Abbildung 5: Ausgelesenes Bild der Mobiley Kamera beim Kalibrierungsvorgang¹

Das System benötigt Geschwindigkeitsdaten, um die spezifischen in Tabelle 1 beschriebenen Funktionsgruppen zu aktivieren. Diese werden normalerweise über den im Fahrzeug verbauten CAN-Bus geliefert. Fahrzeuge, welche keinen CAN-Bus besitzen, kommunizieren ihre Geschwindigkeit über ein analoges Rechtecksignal.

Diese können nur über ein E-Box (Enhancement box) Add-On mit dem System verbunden werden. Diese E-Box ist ein Analog-to-CAN Adapter. Somit kann ein analoges Signal in ein CAN-Signal umgewandelt werden.

Um ein analoges Geschwindigkeitssignal zu erzeugen, wird ein Funktionsgenerator benötigt. Dieser kann die entsprechenden Rechtecksignale ausgeben, welche von der E-Box als Geschwindigkeitssignal erkannt werden. 1 Hz entspricht dabei ca. 1,42857 km/h. Da es wesentlich aufwendiger ist, einen CAN-Bus zu simulieren, wird im Labor das Rechtecksignal genutzt.

¹ Das Bild wurde direkt aus dem Sensor ausgelesen. Die Qualität entspricht nicht der eigentlichen Kameraauflösung des Sensors. Ein Grund für den Qualitätsverlust könnten Antialiasing-Effekte sein, die für weichere Kanten bei niedrigen Auflösungen sorgen.

4 Erstellung eines Szenarios

Um ein Szenario zu erstellen, wird eine Karte benötigt, auf der der Test stattfinden soll. Damit die Fahrzeuge später fahren können, ohne dass sie vom Nutzer gelenkt werden müssen, braucht die Karte eine OpenDRIVE-Datei [13], welche den Straßenverlauf enthält.

4.1 Erzeugung des Straßennetzes

Es ist sinnvoll, die Straßen mit einem entsprechenden Tool zu erstellen, welches OpenDRIVE-Dateien zusammen mit dem Straßennetz als 3D-Objekt in einer FBX-Datei [14] ausgibt, um eine Übereinstimmung der Koordinaten auf die Straße zu garantieren. Von den CARLA-Entwicklern selbst wird die Software RoadRunner [16] empfohlen. Diese Software ermöglicht das Erstellen von Umgebungskarten, die mit allen benötigten Dateien für CARLA exportiert werden können.



Abbildung 6: Übersicht der Roadrunner Oberfläche [14]

4.2 Bearbeitung in der Unreal Engine

Diese Dateien werden in CARLA importiert. Dort können dann weitere Elemente wie Straßenschilder hinzugefügt werden. Um die Gegenstände auf ihre entsprechenden Größen zu skalieren, müssen in der Engine allerdings Objekte als Größenreferenz herangezogen werden, denn die Übereinstimmung von Einheiten in der Unreal Engine zu realen Einheiten ist nicht garantiert. Nicht jedes Model kommt aus derselben Quelle und bei der Erstellung wird die Größe eines Models in einer Engine eventuell nicht bedacht.

Um eine Referenz zu erstellen, wurde daher ein von CARLA vorgegebenes Mercedes C-Klasse Model untersucht. Dabei wurden die vom Hersteller angegebenen Werte mit einem Standard-Würfel aus der Unreal Engine verglichen. Der Standard-Würfel in der Unreal Engine hat eine Größe von 100x100x100 Unreal Engine Einheiten. Das Ergebnis

zeigt, dass der Würfel bei einer Vergrößerung um das 2,016-fache der Breite des Fahrzeuges entspricht. Somit ist eine Einheit gleich mit 1 cm in Bezug auf das C-Klasse Model.

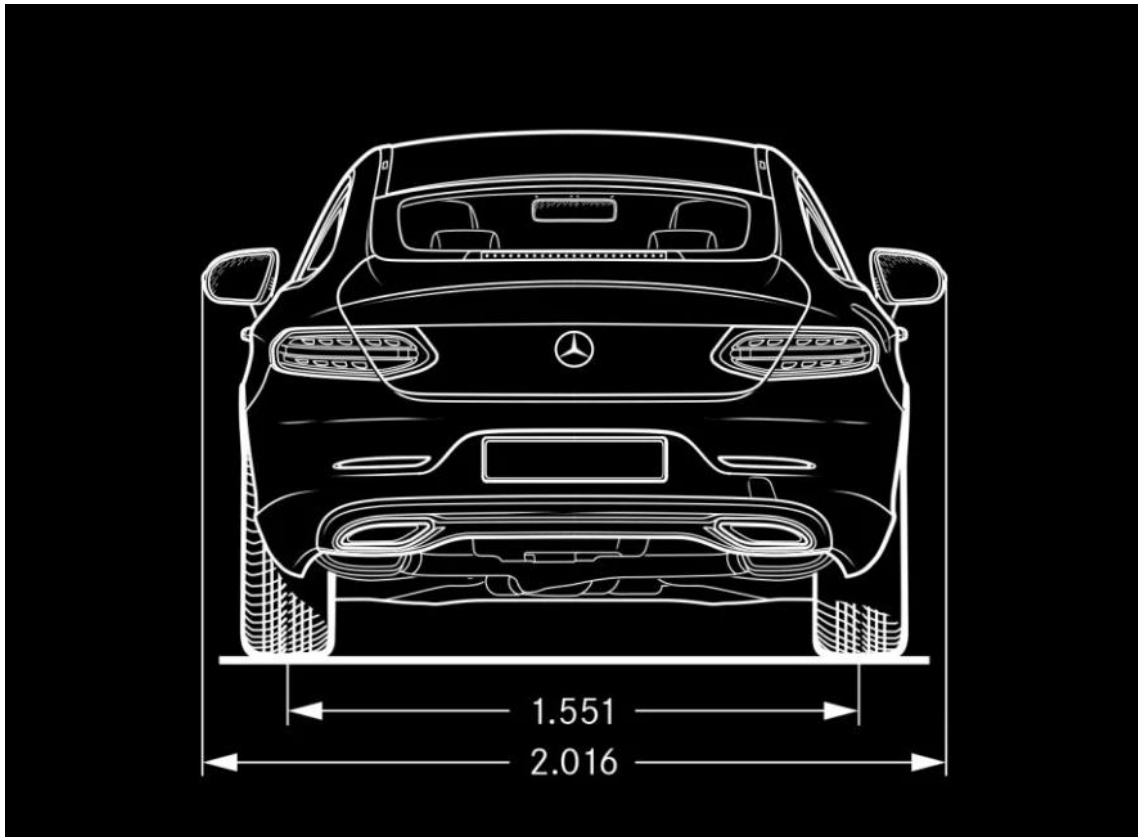


Abbildung 7: Herstellerangaben zu dem Mercedes C-Klasse Model in Metern [23]

4.3 Simulation

Um ein Szenario zu erstellen, muss mit CARLA zunächst ein lokaler Server erstellt werden. Auf diesem befindet sich die Umgebung, welche zunächst leer ist. Sobald der Server aktiv ist, können verschiedenste Python Skripte die Simulation beeinflussen, indem bspw. andere nicht steuerbare Verkehrsteilnehmer erzeugt werden. Diese können die Karte anhand der Wegbeschreibung in der OpenDRIVE-Datei befahren. Jedoch können Verkehrsteilnehmer wie Fußgänger standardmäßig nicht kontrolliert werden, was den Ablauf eines Szenarios erschwert.

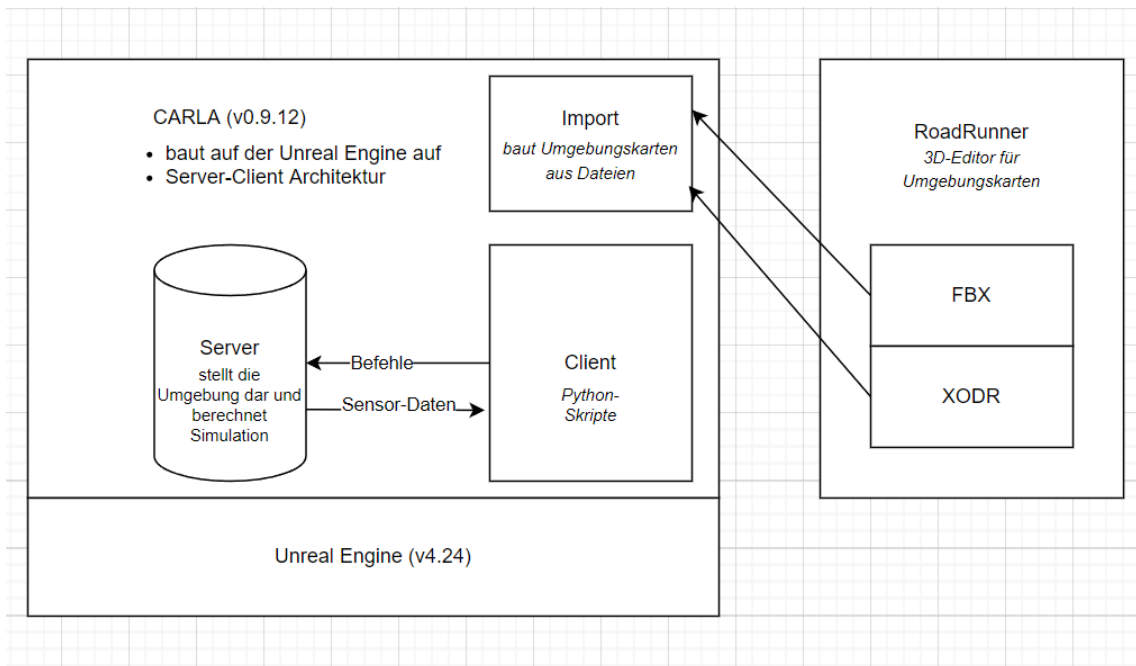


Abbildung 8: Darstellung der Simulationsumgebung und ihrer Einzelteile

Die Simulation wird im Wesentlichen mit drei Skripten kontrolliert, die jeweils eine Aufgabe haben:

- manual_control.py
- generate_traffic.py
- environment.py

Um die Sicht eines Fahrers zu erhalten, wird manual_control.py aufgerufen. Das Skript erstellt ein steuerbares Fahrzeug. Zur Darstellung wird die Python-Programmbibliothek „Pygame“ [17] verwendet. Das steuerbare Fahrzeug kann nun genutzt werden, um die Testobjekte innerhalb der Umgebung abzufahren. Dabei stehen verschiedene Ansichten zur Verfügung. Um diese zu verändern, müssen neue Ansichten im Skript angelegt werden.

Um andere Verkehrsteilnehmer zu erzeugen wird generate_traffic.py aufgerufen. Durch dieses Skript wird eine bestimmte Anzahl (Default = 30) an Fußgängern und Fahrzeugen erzeugt. Während der Erstellung der Szenarien hat sich herausgestellt, dass sich die Wegfindung des Ego-Fahrzeuges² verbessert, wenn das generate_traffic.py Skript ausgeführt wird.

Um Sichtverhältnisse und Wetterbedingungen anzupassen, wird das Skript environment.py aufgerufen. Mit diesem Skript lässt sich bspw. Nebel erzeugen. Die Dichte des Nebels kann ebenfalls angepasst werden.

² Als Ego-Fahrzeug wird das Fahrzeug bezeichnet, welches der Nutzer steuert.

CARLA liefert die grundlegenden Skripte mit, es können aber auch eigene Skripte geschrieben werden. Die genauen Funktionen und Abläufe aller Skripte und Programme werden im Anhang beschrieben.

Um das Szenario aufzuzeichnen, benötigt man ein Bildschirmaufnahme-Programm, welches das neu geöffnete Fenster aufnimmt.

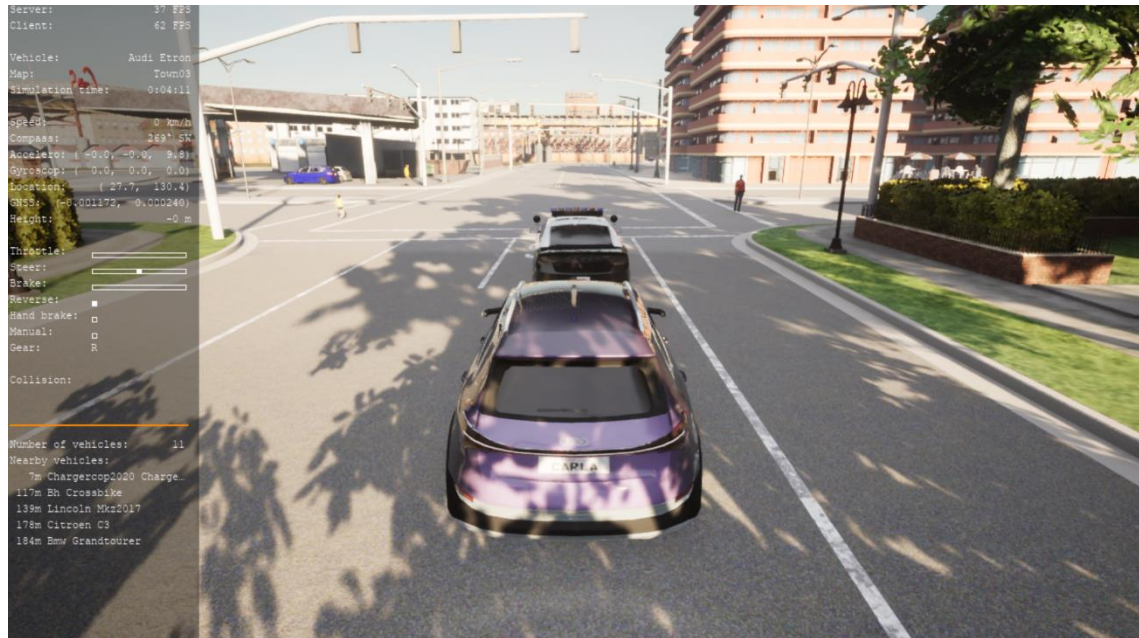


Abbildung 9: Neu erzeugtes Fenster. Lila Fahrzeug wird vom Nutzer gesteuert. Fußgänger und Polizei-Auto werden von einem anderen Skript erzeugt und kontrolliert.

5 Vorgehensweise und Versuchsaufbau

5.1 Vorgehensweise

Die bisherigen Testergebnisse der IPG CarMaker-Szenarien wurden aus 10 Testdurchläufen ermittelt, in denen die erwartete Ausgabe des Sensors mit der tatsächlichen Ausgabe verglichen wurde. [7] Dabei gibt es zwei Arten von Tests. Zum einen können Tests dazu genutzt werden, um die Parameter des Sensors auszuarbeiten (Distanz bis zur Erkennung, Mindestgröße des zu erkennenden Objektes usw.). Die zweite Möglichkeit besteht aus einem tatsächlichen Angriffsversuch auf das System (aufgemalte Personen auf der Fahrbahn etc.).

Um den Sensor auf seine Reaktion zu testen, werden Videos zu einem Szenario in der Simulationsumgebung aufgenommen und anschließend auf einem Fernseher abgespielt. Zwar verliert das Szenario mit diesem Vorgehen an Qualität, da Videos eine vorgegebene Bildwiederholrate und Auflösung haben, doch die Durchführung ist um einiges leichter als ein Realversuch. Bei Realversuchen ist ein deterministischer Ablauf nicht möglich, da man einen Durchlauf nicht perfekt wiederholen kann. Zudem ist die Abänderung von einzelnen Details wesentlich aufwändiger.

Der Sensor aktualisiert die Bilder 15-mal pro Sekunde, die Videos laufen mit 22 bis 30 Bildern pro Sekunde. Somit ist selbst bei demselben Ablauf im Video nicht garantiert, dass der Sensor in jedem Durchlauf die gleichen Bilder sieht. Die 10 Testdurchläufe prüfen den Sensor auf seine Konstanz, ohne dass sich dabei das Szenario ändert, denn selbst leichte Änderungen können Auswirkungen auf das Ergebnis haben.

5.1.1 Vorgehen beim Vergleich

Aus den vorhandenen Videos des IPG CarMakers werden 10 repräsentative Testszenarien ausgewählt. Die ausgewählten Testvideos werden auf ihre wesentlichen Testparameter untersucht. Die wichtigsten Details werden mit Hilfe der neuen Simulationsumgebung CARLA nachgebaut. Anschließend werden die Szenarien in der neuen Simulation durchgeführt und aufgenommen.

5.2 Versuchsaufbau

Damit der Sensor ein möglichst realistisches Bild bekommt, wird dieser auf einem Ständer vor einem Fernseher [24] mit einer Auflösung von 3840x2160 Pixeln und einer Bildschirmdiagonale von 165 cm aufgestellt. Die Kamera befindet sich auf einer Höhe von 1,47 Meter und ist 1,42 Meter vom Bildschirm entfernt. Die Position der Ständerfüße ist auf dem Boden markiert. Die Mobileye Kamera ist mit dem Funktionsgenerator über die E-Box verbunden und erhält so die Geschwindigkeitsdaten.

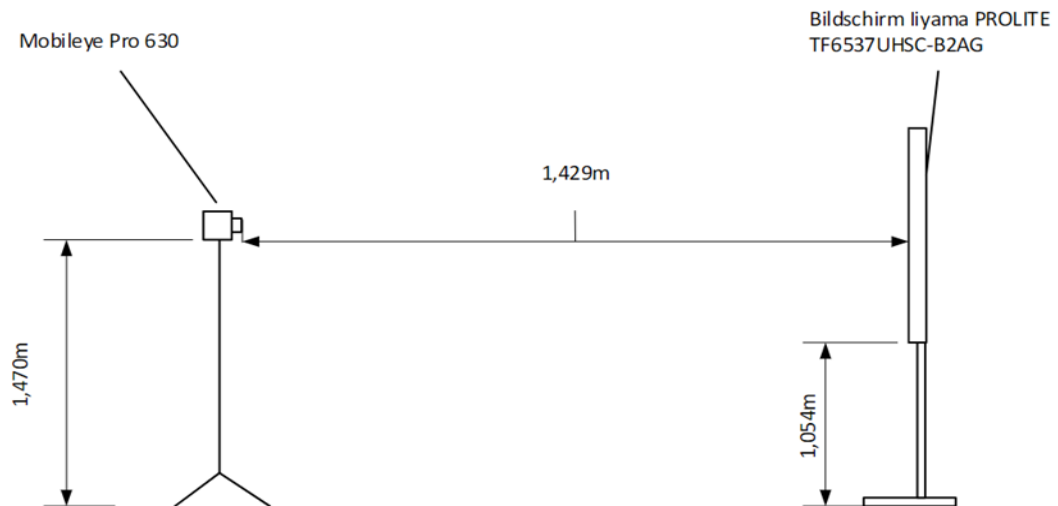


Abbildung 10: Zeichnung des Versuchsaufbaus im Labor [7]

6 Vergleich zwischen IPG CarMaker und CARLA

6.1 Testablauf

Um einen deterministischen Ablauf des Szenarios zu gewähren, wird der Test innerhalb der Simulation mittels der „Open Broadcaster Software“ (kurz OBS) aufgenommen. Diese Software ermöglicht die Aufnahme des PC-Bildschirms und erstellt aus diesen Aufnahmen Videos. Die Videos besitzen eine Auflösung von 2560x1440 Pixeln. Die Verwendung von Videos garantiert einen einheitlichen Ablauf aller Durchläufe ohne Abweichungen, die das Ergebnis verfälschen könnten.

Die erstellten Videos werden nun der Reihe nach mit vorhandenen Videos des IPG CarMakers verglichen. Jeder Test besteht aus 10 Durchläufen. Jeder Test hat eine bestimmte Ausgabe des Sensors als erwartetes Ergebnis. Erkennt der Sensor das zu testende Objekt, wird der Durchlauf mit einem „✓“ markiert. Trifft das Ergebnis nicht ein, so wird der Durchlauf mit einem „x“ markiert.

Ziel des Tests ist der Vergleich zwischen den Videos von IPG CarMaker und CARLA. Es wird untersucht, ob mit CARLA erstellte Videos die Aufgabe der IPG-Videos übernehmen können. Dafür werden beide Videoarten dem bisher genutzten Sensor vorgeführt und auf die Reaktion des Sensors verglichen.

6.2 Vergleich und Resultate

Im Folgenden werden die ausgewählten Tests der Reihe nach beschrieben und die Ergebnisse verglichen. Die Szenarien werden in der folgenden Tabelle mit den zu testenden Funktionen aufgeführt.

Szenario	Funktion
Fußgänger überquert Fahrbahn	PCW
Kollision auf Autobahn bei Nebel	FCW
Fahrzeug mit himmelsähnlichem Farbton	UFCW
Hindernisse/Schilder aufgemalt auf der Fahrbahn	PCW/TSR
Straßenschilder auf Häuserwänden	TSR
Fahrt hinter einem Fahrradfahrer	PCW
Verkehrsschilder im Nebel	TSR
Verkehrsschilder bei Nacht	TSR
Verkehrsschilder bei Sonne	TSR

Tabelle 2: Auflistung der zu vergleichenden Szenarien

6.2.1 Fußgänger überquert Fahrbahn

In diesem Versuch wird untersucht, ob eine erwachsene Person, welche die Straße überquert, von dem System erkannt wird. Das Fahrzeug fährt durch eine Stadt auf einen Fußgänger zu, welcher die Straße überquert. Das Fahrzeug bremst und lässt den Fußgänger passieren. Sobald dieser die Straße überquert hat, wird die Fahrt fortgesetzt. Die Fahrgeschwindigkeit ist hierbei auf ca. 28,6 km/h (20Hz) eingestellt. Die Wetterbedingungen sind sonnig und die Sichtverhältnisse sind klar. Aufgrund der Geschwindigkeit sind nur die PCW und UFCW Funktionen für den Test relevant.

In dem CARLA-Video kommt das Fahrzeug aus dem Westen, im IPG-Video aus dem Osten. Dies führt zu einem anderen Schatteneinfall. Die Schattenfläche ist im IPG-Video größer.



Abbildung 11: Ausschnitt des CARLA-Videos „Fußgänger“



Abbildung 12: Ausschnitt des IPG-Videos „Fußgängerin“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
IPG CarMaker	x	x	x	x	x	x	x	x	x	x

Tabelle 3: Ergebnisse des Tests: Fußgänger überquert die Fahrbahn. Bei Durchläufen mit einem ✓ wurde der Fußgänger erkannt.

Die Fußgängerin wird im IPG-Video nicht erkannt. Gründe hierfür können unter anderem die Distanz zur Fußgängerin sein. Diese ist im CARLA-Video deutlich näher am Fahrzeug. Zusätzlich befindet sich die Fußgängerin im IPG-Video länger im Schatten. Das T-Shirt hebt sich nicht besonders gut von der Fassade im Hintergrund ab.

6.2.2 Kollision auf Autobahn bei Nebel

Bei diesem Szenario fährt ein Fahrzeug auf einer Autobahnstrecke, welche in dichten Nebel gehüllt ist. Auf derselben Spur befindet sich ein weiteres stehendes Fahrzeug. Nach einiger Zeit werden die beiden Fahrzeuge kollidieren. Die Sicht ist stark beeinträchtigt und die eingestellte Geschwindigkeit beträgt etwa 113,6 km/h (80hz). Wichtig für diesen Test sind die HMW und FCW Funktionen.

Der Nebel ist im CARLA-Video dunkler. Deshalb wurde das Fahrzeug schwarz lackiert, um den Kontrast ähnlich zum IPG-Video zu halten.



Abbildung 13: Ausschnitt des CARLA-Videos „Nebel Kollision“



Abbildung 14: Ausschnitt des IPG-Videos „Nebel Kollision“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	x	x	x	x	x	x	x	x	x	x
IPG CarMaker	x	x	x	x	x	x	x	x	x	x

Tabelle 4: Ergebnisse des Tests: Kollision bei Nebel. Durchläufe, die mit einem x markiert sind, haben keine Kollisionswarnung ausgegeben.

Bei diesem Test wurde das stehende Fahrzeug unabhängig vom den Simulationsumgebungen nicht erkannt und keine Kollisionswarnung ausgegeben. In beiden Fällen scheinen die stehenden Fahrzeuge sich nicht genug von der Umgebung abzuheben.

6.2.3 Fahrzeug mit himmelsähnlichem Farbton

In diesem Szenario, fährt ein Fahrzeug auf einer Landstraße auf einen stehenden LKW zu. Die Plane des stehenden Fahrzeugs ist an den Hintergrund der Simulation angepasst. Ziel des Szenarios ist das Erkennen des stehenden Fahrzeuges und die Ausgabe einer Kollisionswarnung. Die Wetterverhältnisse sind klar. Die Fahrgeschwindigkeit ist auf ca. 28,6 km/h (20Hz) eingestellt. Im Fokus steht die UFCW Funktion.

Aufgrund von Sonneneinfall und Schattenpositionen wurde die Tageszeit im CARLA-Video angepasst. Dementsprechend besitzt die Plane eine andere Farbe als im IPG-Video

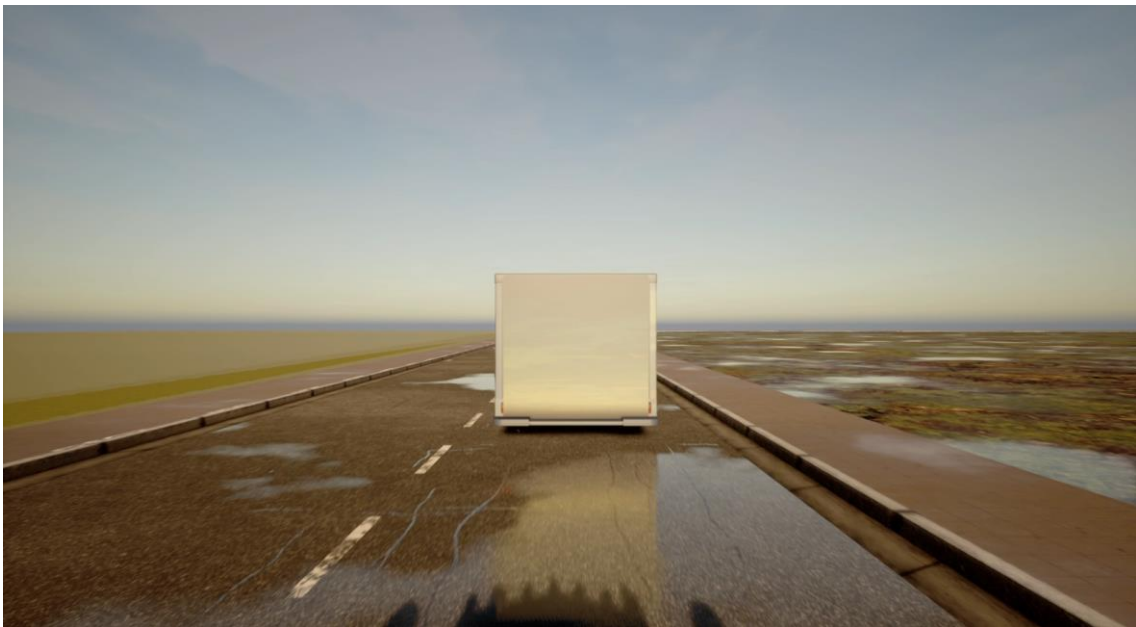


Abbildung 15: Ausschnitt des CARLA-Videos „Hindernis mit Himmelfarbtton“



Abbildung 16: Ausschnitt des IPG-Videos „Überlagerung Himmel LKW“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	x	x	x	x	x	x	x	x	x	x
IPG CarMaker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 5: Ergebnisse des Tests: Fahrzeug mit himmelsähnlichem Farbton. Durchläufe, die mit einem x markiert sind, haben keine Kollisionswarnung ausgegeben.

Lediglich der LKW im IPG-Video wurde erkannt. Im CARLA-Video wurde durch die Abdeckung mit der Himmelstextur das Rücklicht des Fahrzeuges überlagert. Dies könnte ein Grund für das Nichterkennen des Fahrzeuges sein. Das Szenario wird erneut aufgezeichnet. Diesmal wird die Plane des Fahrzeuges so verschoben, dass die Rücklichter nicht verdeckt werden.



Abbildung 17: Ausschnitt des CARLA-Videos „Hindernis mit Himmelfarbton Neu“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 6: Ergebnisse des veränderten Szenarios

Diesmal wird der Transporter erkannt. Der Transporter befindet sich zwar an einer anderen Position mit veränderter Sonneneinstrahlung, die Vermutung liegt jedoch nahe, dass Fahrzeugmerkmale für die Erkennung relevant sein könnten. Ebenso könnte der vom Transporter geworfene Schatten bei der Erkennung eine wichtige Rolle gespielt haben.

6.2.4 Hindernisse/Schilder aufgemalt auf der Fahrbahn

In diesem Szenario werden entlang einer Route verschiedene Hindernisse und Verkehrszeichen als Bilder auf dem Boden dargestellt. Das Fahrzeug fährt diese Route ab. Untersucht wird, ob die Objekte von der Kamera erkannt werden und dementsprechend

ein Hinweis ausgegeben wird. Die Wetterbedingungen sind klar und sonnig. Als Geschwindigkeit wurde 28,6 km/h (20Hz) eingestellt. In der folgenden Tabelle wird aufgelistet, welche Symbole bei beiden Videos getestet werden und wie der Sensor darauf reagieren kann. Da die Bilder auf dem Boden für den Fahrer nicht relevant sind, sollte der Sensor diese ignorieren. Es handelt sich also um einen Angriffsversuch. Folgende Objekte werden auf beiden Routen dargestellt.

Bild	Kategorie
Dreißiger Zone	TSR
Fußgänger stehend	PCW
Autorückseite	PCW
Fußgänger laufend	PCW

Tabelle 7: Auf dem Boden gemalte Texturen aus dem Test „Hindernisse/Schilder aufgemalt auf der Fahrbahn“



Abbildung 18: Fußgängertexturen, die für das CARLA-Video verwendet werden [26]



Abbildung 20: Autorückseite, die im CARLA-Video verwendet wird [27]



Abbildung 19: Tempo 30 Zone

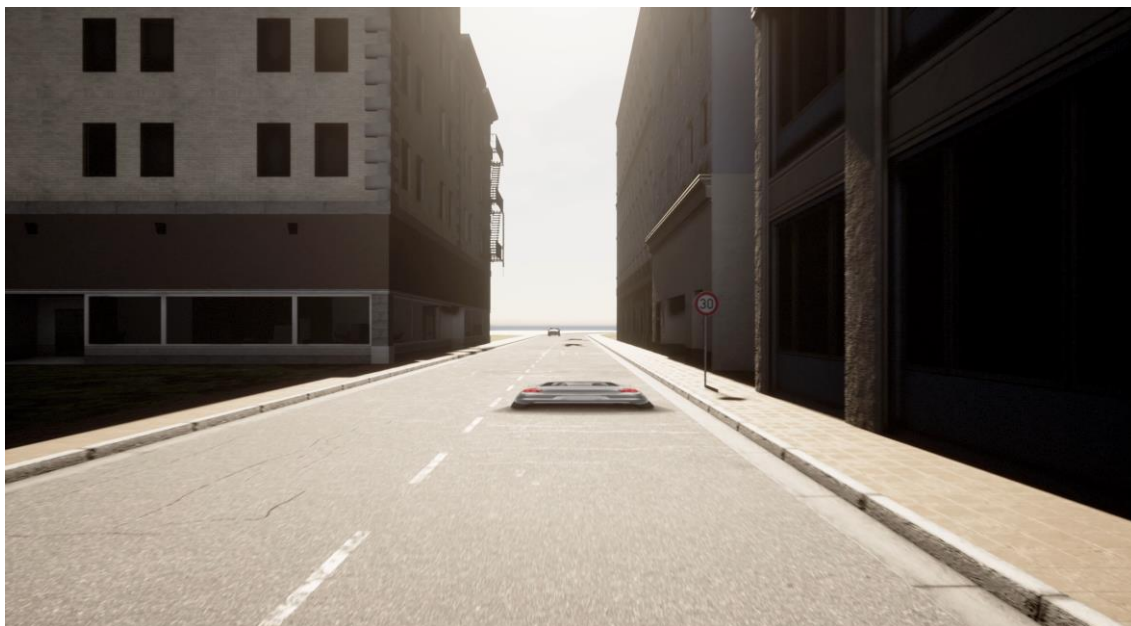


Abbildung 21: Ausschnitt des CARLA-Videos „Roadpaintings“



Abbildung 22: Ausschnitt des IPG-Videos "Roadpaintings"

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	x	x	x	x	x	x	x	x	x	x
IPG CarMaker	x	x	x	x	x	x	x	x	x	x

Tabelle 8: Erkennung Tempo 30 Zone. X bedeutet, dass die Markierung nicht erkannt wurde

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	x	x	x	x	x	x	x	x	x	x
IPG CarMaker	x	x	x	x	x	x	x	x	x	x

Tabelle 9: Erkennung Fußgänger stehend.

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	x	x	x	x	x	x	x	✓	✓	✓
IPG CarMaker	x	x	x	x	x	x	x	x	x	x

Tabelle 10: Erkennung Autorückseite

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	x	x	x	x	x	x	x	x	x	x
IPG CarMaker	x	x	x	x	x	x	x	x	x	x

Tabelle 11: Fußgänger laufend

Grundsätzlich wurden die Markierungen auf dem Boden nicht erkannt. Auffallend ist, dass nach minimaler Veränderung der Kameraposition nach rechts in dem CARLA-Video nach dem 7. Durchlauf dazu geführt hat, dass die Automarkierung als Hindernis erkannt wurde.

6.2.5 Straßenschilder auf Häuserwänden

In diesem Szenario wird geprüft, ob Schilder auf Häuserwänden erkannt werden oder durch das System gefiltert werden. Entlang einer Straße durch ein Dorf werden zwei Schilder mit einer Geschwindigkeitsbegrenzung von 60 km/h an Häuserwänden platziert. Das erste Schild befindet sich auf einem Haus angrenzend zur rechten Seite der Fahrbahn. Das zweite Schild befindet sich auf der linken Seite. Die Wetterbedingungen sind klar und sonnig. Als Geschwindigkeit wurde 28,6 km/h (20hz) eingestellt. Das zweite Schild liegt bei CARLA in der Sonne. Es handelt sich hierbei um einen Angriffsversuch, da Verkehrsschilder normalerweise nicht auf Häuserwänden angebracht werden.

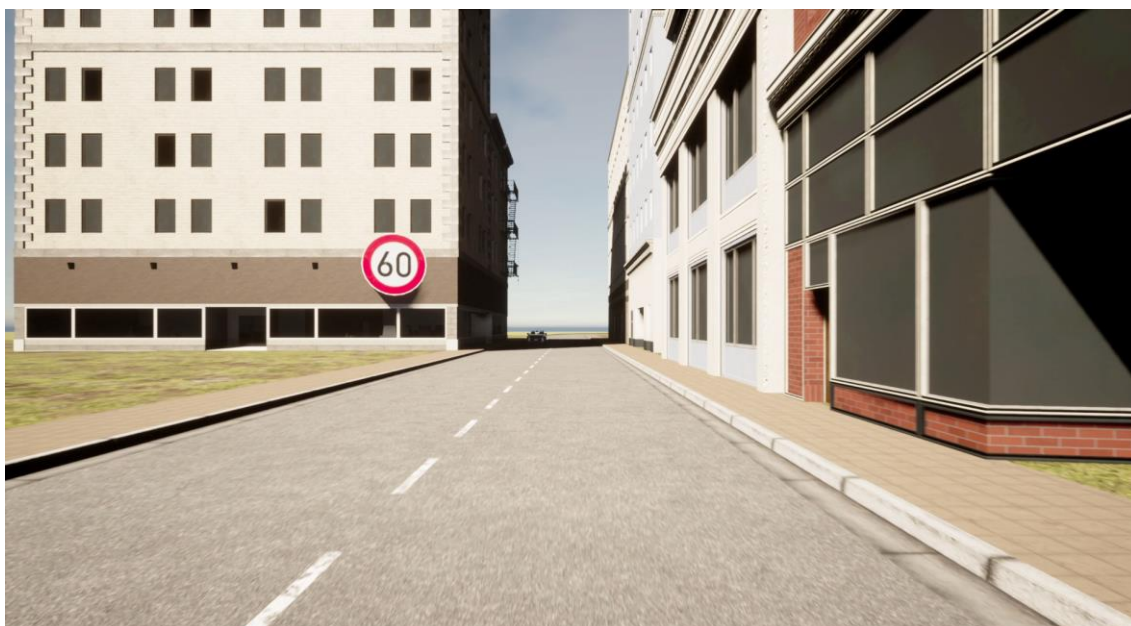


Abbildung 23: Ausschnitt des CARLA-Videos „Schild auf Haus“



Abbildung 24: Ausschnitt des IPG-Videos "Schild auf Haus"

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	✓	✓	✓	x	x	✓	✓	✓	x	✓
IPG CarMaker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 12: Erkennung beider Schilder

Die Schilder werden in beiden Videos erkannt, allerdings ist das Ergebnis im CARLA-Video nicht konsistent. Ein möglicher Grund könnte eine Filterung des Systems sein, welches das zweite Schild als nicht relevant für den Fahrer betrachtet, da es sich zu weit links befindet.

6.2.6 Fahrt hinter einem Fahrradfahrer

In diesem Szenario folgt ein Fahrzeug einem Fahrradfahrer durch eine Stadt. Das Fahrzeug hält dabei einen konstanten Abstand und bremst ab, bis es die Geschwindigkeit des Fahrradfahrers erreicht. Die Wetterbedingungen sind klar und sonnig. Als Geschwindigkeit wurde 28,6 km/h (20hz) eingestellt. Ziel des Szenarios ist die Erkennung des Fahrradfahrers und die daraus resultierende Ausgabe des Systems.



Abbildung 25: Ausschnitt des CARLA-Videos „Fahrradfahrer“



Abbildung 26: Ausschnitt des IPG-Videos "Fahrradfahrer"

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	x	x	x	x	x	x	x	x	x	x
IPG CarMaker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 13: Erkennung des Fahrradfahrers

Der Fahrradfahrer wird nur im Test mit dem IPG CarMaker erkannt. Es ist zu erkennen, dass die Distanz zwischen dem folgenden Fahrzeug und dem Fahrradfahrer im CARLA-Video etwas größer ist. Im Gegensatz zu CARLA fährt der Fahrradfahrer bei IPG in Schlangenlinien und bewegt sich etwas natürlicher als der Fahrradfahrer in CARLA. Zusätzlich fährt der Fahrradfahrer im CARLA-Video zentraler auf der Fahrbahn. Das Szenario wird erneut aufgezeichnet. Diesmal wird die Distanz zum Fahrradfahrer verkürzt. Der Fahrradfahrer bleibt am Ende des Szenarios stehen und das Fahrzeug führt eine Notbremse kurz davor aus.



Abbildung 27: Ausschnitt des CARLA-Videos „Fahrradfahrer Video 2“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	✓	O	O	✓	✓	✓	✓	✓	✓	✓

Tabelle 14: Erkennung des Fahrradfahrers mit neuer Aufnahme. Bei ✓ wurde der Fahrradfahrer erkannt und eine Kollisionswarnung ausgegeben. Bei O wurde keine Kollisionswarnung ausgegeben, der Fahrradfahrer wurde allerdings erkannt.

Die Ergebnisse zeigen zwar, dass der Fahrradfahrer erkannt wurde, dies geschah allerdings meist erst bei der finalen Notbremsung. Während der Fahrt fand keine Erkennung statt.

6.2.7 Verkehrsschild bei Nebel

Bei dem folgenden Test fährt das Fahrzeug entlang einer Straße durch ein Dorf in dichtem Nebel. Nach einiger Zeit taucht ein Verkehrsschild mit einer Geschwindigkeitsbegrenzung von 30 km/h auf der rechten Seite auf. Die Sicht ist diesmal stark eingeschränkt. Die Geschwindigkeit ist auf 28,6 km/h (20 Hz) eingestellt. Ziel des Szenarios ist die Prüfung der TSR-Funktion.



Abbildung 28: Ausschnitt des CARLA-Videos „Schild im Nebel“



Abbildung 29: Ausschnitt des IPG-Videos „Schild im Nebel“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	✓	✓	✓	✓	✓	✓	✓	✓	✓	x
IPG CarMaker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 15: Schild im Nebel

Das System kann die Verkehrsschilder beider Simulationsumgebungen im Nebel erkennen, allerdings scheint dies im Falle der CARLA-Videos nicht konstant zu sein.

6.2.8 Verkehrsschild bei Nacht

Dieses Szenario ist dem Vorherigen in vielen Bereichen gleich. Die wichtigsten Veränderungen sind die Veränderung der Tageszeit und die Entfernung des Nebels. Diesmal wird die Schilderkennung bei Nacht geprüft. Die Scheinwerfer des Fahrzeuges sind an. Die Geschwindigkeit ist wie zuvor auf 28,6 km/h (20 Hz) eingestellt.



Abbildung 30: Ausschnitt des CARLA-Videos „Schild bei Nacht“



Abbildung 31: Ausschnitt des IPG-Videos „Schild bei Nacht“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
IPG CarMaker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 16: Schild bei Nacht

Auch hier gibt es keine Probleme mit der Schilderkennung. Erkennbar ist allerdings, dass die Scheinwerfer in den CARLA-Videos die Schilder deutlicher erhellen.

6.2.9 Verkehrsschild bei Tag

Dieses Szenario behandelt wie die beiden vorherigen Szenarien die Erkennung von Straßenschildern mit Geschwindigkeitsbegrenzung. Diesmal wird der Test bei sonnigem Wetter und klaren Sichtverhältnissen durchgeführt. Die Geschwindigkeit beträgt 28,6 km/h (20 Hz).



Abbildung 32: Ausschnitt des CARLA-Videos „Schild bei Tag“



Abbildung 33: Ausschnitt des IPG-Videos „Schild bei Tag“

Umgebung	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
CARLA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
IPG CarMaker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 17: Schild bei Tag

Da dieser Test eine vereinfachte Version der vorherigen zwei darstellt, ist es nicht überraschend, dass auch in diesem Fall alle Verkehrsschilder erkannt wurden.

6.2.10 Fazit des Vergleiches

Optisch sticht die hohe Farbsättigung der IPG-Videos hervor. Bei CARLA steht hingegen die Belichtung der Umgebung im Vordergrund. Deshalb sieht die Umgebung in CARLA deutlich realistischer aus.

Im Vergleich treten nur bei zwei Szenarien nennenswerte Abweichungen zum Nachteil von CARLA auf. Diese wurden nochmals aufgezeichnet, um die aufgestellten Thesen für die Abweichungen zu untersuchen. Für das Szenario „Fahrt hinter einem Fahrradfahrer“ wird die Distanz des Fahrradfahrers und des verfolgenden Fahrzeuges verkürzt. Für das Szenario „Fahrzeug mit himmelsähnlichem Farbton“ wird die Plane so versetzt, dass Fahrzeugmerkmale wie beispielsweise Rückleuchten nicht verdeckt werden. Dies führte zu einem verbesserten Testergebnis.

Insgesamt wurden bei 120 Durchläufen mit dem IPG-CarMaker 60-mal (50%) die Funktionen des Sensors ausgelöst. Nicht bei jedem Test war es das Ziel, dass die Kamera auslöst. So sollten die Bilder auf dem Boden z.B. nicht erkannt werden. Unter der

Betrachtung hat der Sensor bei 90 von 120 Durchläufen (75%) in IPG-Videos richtig reagiert.

Da bei CARLA zwei Szenarien nachbearbeitet wurden (6.2.3 und 6.2.6), ändert sich das Ergebnis von CARLA, je nachdem welche Durchläufe bewertet werden. Betrachtet man nur die ersten Szenarien ohne die nachgereichten und veränderten Versionen, so hat CARLA 49 Auslöser (ca. 40,8%) und 77 richtigen Reaktionen (ca. 64,2%) bei 120 Durchläufen.

Ersetzt man hingegen die Szenarien 6.2.3 und 6.2.6 durch ihre veränderte Version, erhöht sich die Erkennungsrate bei CARLA auf 69-mal (57,5%) und 97 richtigen Reaktionen (80,8%). Dies führt zu einem großen Sprung in der Statistik.

Werden die veränderten Szenarien mit beiden Versionen berücksichtigt, verändern sich die Werte zu 140 Durchläufen mit 69 Auslösern (ca. 49,3%) und 97 richtigen Reaktionen (69,3%). Diese Werte liegen knapp unter den Werten des IPG CarMakers.

Der Sensor reagiert insgesamt besser auf die IPG-Videos. Eine Ursache kann die Kalibrierung sein, welche mit IPG-Videos durchgeführt wurde. Der Sensor ist also besser auf IPG-Videos eingestellt. Auffällig ist, dass die Horizontlinie in den CARLA-Videos deutlich höher liegt als bei den IPG-Videos. Zusätzlich ist das Sichtfeld in den IPG-Videos etwas breiter. Diese Unterschiede können die Erkennung des Sensors ebenfalls beeinflussen. Es hat sich ergeben, dass die Umpositionierung der Kamera nach rechts in den CARLA-Videos zu einem besseren Ergebnis führen kann. Bei den IPG-Videos ist dies nicht der Fall.

Die Ergebnisse können mit einer anderen Kamera deutlich anders ausfallen. Für diese Arbeit steht jedoch nur das Mobileye 630 Pro System zur Verfügung. Die Ergebnisse von CARLA sind gut genug, um die Aufgabe der IPG-Videos zu übernehmen. Deshalb wird mit CARLA und dem Testen von gänzlich neuen Szenarien fortgefahren.

7 Testen von neuen Szenarien

Die Simulationsumgebung CARLA ermöglicht die Manipulation von bestimmten Gegenständen wie bspw. Verkehrsschildern und Fahrzeugen, die mit dem IPG CarMaker nicht veränderbar waren. In den neuen Szenarien wurden diesmal mehrere verschiedene Tests zusammengeführt, so dass nur eine Aufnahme benötigt wird, um verschiedenste Hindernisse zu testen.

7.1 Änderungen

Für die nachfolgenden Tests werden die Einstellungen in der Simulationsumgebung CARLA verändert. So wird der Winkel des Sichtfeldes von 90 Grad auf 110 erhöht und die Neigung wird um 30 Grad nach oben rotiert. Diese Änderungen werden vorgenommen, um die Aufnahmen besser an die Kalibrierung des Sensors anzupassen. Da zeitgleich noch weitere Arbeiten mit dem Sensor durchgeführt werden, ist eine Neukalibrierung des Sensors zu diesem Zeitpunkt leider nicht möglich.

7.2 Kollisionswarnungen

Bei dem Test „5.2.3 Fahrzeug mit himmelsähnlichem Farbton“ fiel auf, dass nicht jedes Hindernis von der Kamera erkannt wird, aber leichte Veränderungen zu einem anderen Ergebnis führen können. Dementsprechend wird für die folgenden Tests eine Reihe von Hindernissen abgefahren, die die Reaktion der Kamera prüfen sollen. Das kontrollierte Fahrzeug kann durch die Hindernisse durchfahren, ohne dass es mit ihnen kollidiert. Die Testumgebung steht hierbei unter Extrembedingungen für die Kamera. Das Szenario wird nacheinander sehr hell, nebelig und sehr dunkel aufgezeichnet.

7.2.1 Testhindernisse

Die ausgewählten Hindernisse werden in der folgenden Tabelle beschrieben.

Hindernis	Hindernisart	Abkürzung
Transporter mit hintergrundähnlicher Plane	Fahrzeug	T1
Transporter in weiß	Fahrzeug	T2
Auto als Gegenverkehr	Fahrzeug	A1
Auto Rücklichter an	Fahrzeug	A2
Auto Rücklichter verdunkelt	Fahrzeug	A3
Fußgänger	Person	F1
Abbildung Fußgänger	Person	P1
Abbildung Auto	Fahrzeug	P2
Wand	Umwelt	W1

Tabelle 18: Auflistung der Hindernisse für die neuen Testfälle

Der Transporter ist weiß lackiert und entspricht vom Modell einem amerikanischen Krankenwagen. Das T1 Modell verfügt zusätzlich über eine Plane, die der Plane aus dem Szenario „Fahrzeug mit himmelsähnlichem Farbton“ ähnelt. Als Auto für die Kollisionsabfrage wurde stellvertretend die Mercedes C-Klasse gewählt, da das Modell auch als Vergleichsmodell für Größen innerhalb der Simulation dient.

Die Abbildungen sind Bilder, die aufrecht im Raum stehen und nicht auf dem Boden liegen. Zusätzlich wird eine Person aus der Fußgänger-Bibliothek von CARLA auf die Straße gestellt. Die Wand entspricht einem weißen Block, der auf die Straße gestellt wurde

7.2.2 Szenario bei Mittagssonne

Bei diesem Szenario werden die Hindernisse bei Tag und heller Beleuchtung abgefahren. Die Geschwindigkeit ist auf 28,6 Km/h (20 Hz) eingestellt. Der T1 Transporter besitzt eine himmelsähnliche Plane.

T1



T2



A1



A2



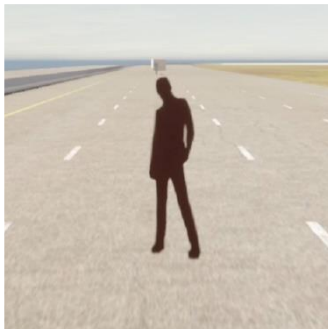
A3



F1



P1



P2



W1

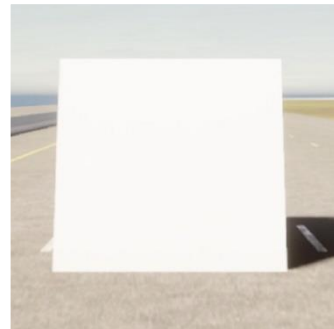


Abbildung 34: Die 9 Testobjekte bei Tag

Hindernis	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
T1	x	x	x	x	x	x	x	x	x	x
T2	x	x	x	x	x	x	x	x	x	x
A1	x	x	x	x	x	x	x	x	x	x
A2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
A3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
F1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
P1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
P2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
W1	x	x	x	x	x	x	x	x	x	x

Tabelle 19: Szenario „Kollisionstest bei Tag“

Der Transporter und die Wand werden nicht erkannt. Ebenso wird der Mercedes von vorne nicht erkannt. Der T1 Transporter ist aufgrund der Sonneneinstrahlung nun schwerer zu erkennen und wirft keinen Schatten in Richtung der Kamera. Anders als im Wiederholungstest im Vergleich wird er nun nicht mehr erkannt.

7.2.3 Szenario bei Nebel

Bei diesem Szenario werden die Hindernisse bei dichtem Nebel abgefahren. Die Plane des T1 Transporters ist nun schwarz. Die Geschwindigkeit ist auf 28,6 Km/h (20 Hz) eingestellt.

T1



T2



A1



A2



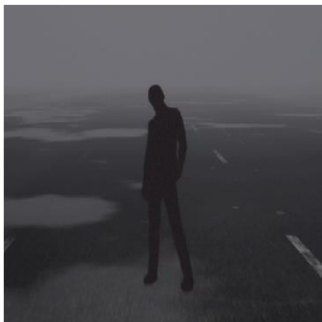
A3



F1



P1



P2



W1



Abbildung 35: Die 9 Testobjekte bei Nebel

Hindernis	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
T1	x	x	x	x	O	x	x	x	x	x
T2	x	x	x	x	x	x	x	x	x	x
A1	x	x	x	x	x	x	x	x	O	x
A2	x	x	x	x	x	x	x	x	x	x
A3	x	x	x	x	x	x	x	x	x	x
F1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
P1	✓	x	x	x	✓	✓	✓	x	✓	✓
P2	O	x	O	O	O	x	O	O	O	x
W1	x	x	x	x	x	x	x	x	x	x

Tabelle 20: Szenario „Kollisionstest bei Nebel“. ✓ zeigt, dass eine UFCW Warnung ausgegeben wurde. Bei „O“ wurde das Objekt zwar erkannt, aber keine Warnung ausgegeben.

Bei diesem Szenario wird nur der Fußgänger konstant erkannt. Aufgrund des Szenarios „Kollision auf Autobahn bei Nebel“ war das Ergebnis, dass die Autoerkennung bei Nebel schwächelt, vorhersehbar. Der Sensor ist bei diesen Verhältnissen nicht konstant.

7.2.4 Szenario bei Nacht

Bei diesem Szenario werden die Hindernisse bei tiefer Nacht abgefahren. Die Plane des T1 Transporters ist nun schwarz. Die Geschwindigkeit ist auf 28,6 Km/h (20 Hz) eingestellt.

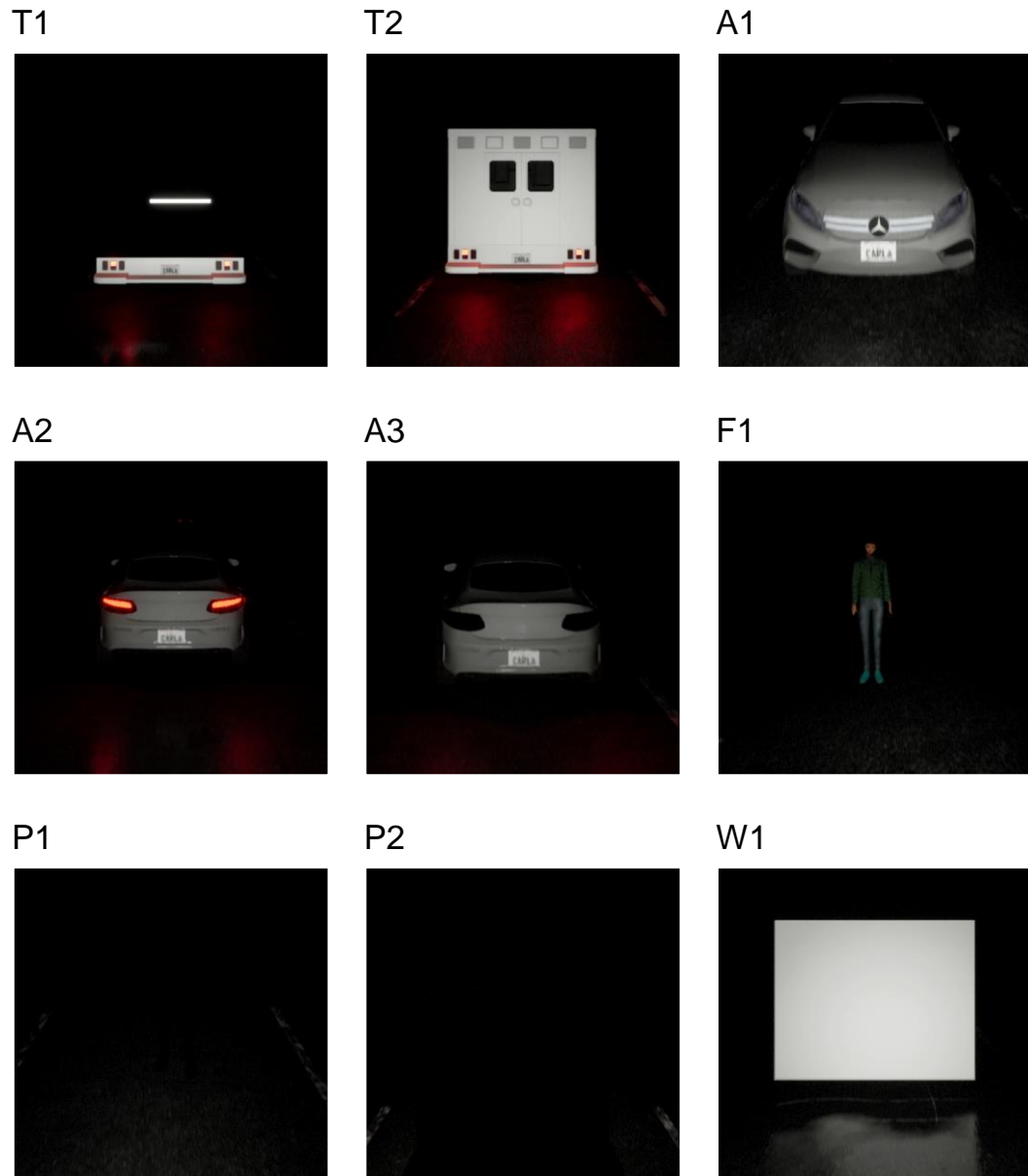


Abbildung 36: Die 9 Testobjekte bei Nacht

Hindernis	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
T1	x	x	x	x	x	x	x	x	x	x
T2	x	x	x	x	x	x	x	x	x	x
A1	x	x	x	x	x	x	x	x	x	x
A2	x	x	x	x	x	x	x	x	x	x
A3	x	x	x	x	x	x	x	x	x	x
F1	x	x	x	✓	x	x	x	✓	✓	✓
P1	x	x	x	x	x	x	x	x	x	x
P2	x	x	x	x	x	x	x	x	x	x
W1	x	x	x	x	x	x	x	x	x	x

Tabelle 21: Szenario „Kollisionstest bei Nacht“

Das Szenario ist sehr dunkel aufgezeichnet. Die Kamera schafft es nicht, etwas zu erkennen; lediglich der Fußgänger wird überhaupt erkannt, da er sich am meisten von der Umgebung abhebt. Die Testobjekte tauchen aus kurzer Entfernung aus der Dunkelheit auf. Das Licht des Fahrzeuges ist nicht stark genug um die Umgebung ausreichend zu erhellen. Die Plane des T1 Transporters verschmilzt nahtlos mit der Umgebung. Die Abbilder P1 und P2 sind in der Aufnahme auch mit menschlichen Augen kaum zu sehen, deshalb ist es nicht verwunderlich, dass diese nicht erkannt werden.

7.2.5 Fazit der Kollisionserkennung

Aus den Ergebnissen lässt sich schließen, dass der Sensor sich mit dem Transporter in Szenarien mit geringem Kontrast besonders schwertut. Autos aus dem Gegenverkehr werden ebenfalls nicht wahrgenommen. Das Nachtszenario überschreitet die Grenzen der Kamerafähigkeiten.

Die Wand wird in keinem Szenario als Hindernis erkannt. Der Sensor sucht vermutlich nach Erkennungsmerkmalen von Objekten im Bild. Die Wand und der Gegenverkehr lösen den Sensor nicht aus.

7.3 Schilderkennung

In diesem Szenario geht es um die Erkennung von verschmutzten Schildern. Diese sind mit Graffiti besprüht oder verdreckt. Das Fahrzeug fährt eine Teststrecke ab, an der die Schilder aufgestellt sind. Die Schilder treten in den von der „Allgemeine Verwaltungsvorschrift zur Straßenverkehrs-Ordnung“ (VwV-StVO) unter §§ 39 bis 43 beschriebenen Größen von 42cm, 60cm und 75cm Durchmesser auf [25]. Hierfür ist die in Kapitel 4 erwähnte Referenz wichtig, um die Größe der Schilder anzupassen.

7.3.1 Testschilder

Verkehrsschild	Größe	Abkürzung
Tempo 30 verdreckt	1	D1
Tempo 60 mit ausgemalter 0	1	G1
Tempo 60 verdreckt	2	D2
Tempo 130 teilweise übersprüht	3	G2
Tempo 120 mit Grafit-Schriftzug	3	G3

Tabelle 22: Auflistung der Schilder für die neuen Testfälle

7.3.2 Schildtest bei Tag

Die Schilder werden unter hellen Bedingungen getestet. Die Sonneneinstrahlung ist stark. Die Geschwindigkeit ist auf 28,6 (20 Hz) eingestellt.



Abbildung 37: Verschmutzte Schilder bei Tag

Hindernis	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
D1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
G1	x	x	x	x	x	x	x	x	x	x
D2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
G2	x	x	x	x	x	x	x	x	x	x
G3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 23: Szenario „verschmutzte Schilder bei Tag“

Schild G1 und G2 scheinen zu stark verändert zu sein, um vom Sensor erkannt zu werden.

7.3.3 Schildtest bei Nebel

Die Schilder werden bei starkem Nebel getestet. Die Geschwindigkeit ist auf 28,6 (20 Hz) eingestellt.



Abbildung 38: Verschmutzte Schilder bei Nebel

Hindernis	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
D1	x	✓	✓	✓	✓	✓	✓	✓	x	x
G1	x	x	x	x	x	x	x	x	x	x
D2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
G2	x	x	x	x	x	x	x	x	x	x
G3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 24: Szenario „verschmutzte Schilder bei Nebel“

Die kleineren Schilder werden im Nebel nicht mehr zu 100% erkannt.

7.3.4 Schildtest bei Nacht

Die Schilder werden bei Nacht getestet. Die Geschwindigkeit ist auf 28,6 (20 Hz) eingestellt.



Abbildung 39: Verschmutzte Schilder bei Nacht

Hindernis	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
D1	x	x	x	x	x	x	x	x	x	x
G1	x	x	x	x	x	x	x	x	x	x
D2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
G2	x	x	x	x	x	x	x	x	x	x
G3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Tabelle 25: Szenario „verschmutzte Schilder bei Nacht“

Bei diesem Szenario ist die Umgebung etwas besser beleuchtet als im Szenario „Kollision bei Nacht“. Die Erkennung der kleinsten Schilder ist nicht mehr möglich. Schilder mit 60 cm und mehr Durchmesser werden aber weiterhin konstant erkannt, wenn sie im Szenario zuvor ebenfalls erkannt wurden.

7.3.5 Fazit Schilderkennung

Solange die Zahlen auf den Schildern noch klar zu erkennen sind, hat der Sensor keine Probleme, diese zu erkennen. Staub scheint keine Auswirkungen auf den Sensor zu haben, aber sobald Zahlen vollständig verdeckt werden, ist der Sensor nicht mehr in der Lage, diese zu erkennen. Es gibt eine Resistenz gegenüber leichten Überlagerungen wie bei G3, bei schwereren Fällen schafft der Sensor jedoch keine Erkennung, auch

wenn das menschliche Auge noch in der Lage wäre, Zahlen zu erkennen oder zu interpretieren. Sobald der Kontrast schwächer wird, werden die Schilder mit 42 cm Durchmesser nicht mehr konstant erkannt. Wird das Video zu dunkel, werden diese gar nicht mehr erkannt. Allerdings muss angemerkt werden, dass das D1 Schild sich am Ende einer Kurve befand. Das Auto fuhr nicht grade auf das Schild zu. Möglicherweise wurde das Ergebnis dadurch beeinflusst.

8 Fazit und Ausblick

Die Simulationsumgebung CARLA ist im direkten Vergleich mit dem IPG CarMaker durchaus in der Lage, ähnliche Tests durchzuführen. Die Ergebnisse zeigen zwar, dass der IPG CarMaker insgesamt in den Tests mit dem Mobileye etwas besser abschneidet, jedoch können die Ergebnisse sich verbessern, wenn man den Sensor auf CARLA kalibriert.

Bei der Erstellung der Szenarien hat sich herausgestellt, dass die Kontrolle von Fußgängern nicht einwandfrei funktioniert. Die Fußgänger bewegen sich zufällig, was den kontrollierten Aufbau des Szenarios erschwert. Meist werden mehrere Anläufe benötigt, um ein gutes Video zum Testen des Sensors zu erstellen.

Im Vergleich zwischen CARLA und dem IPG CarMaker zeigt sich, dass die IPG-Videos mit dem Mobileye 630 Pro System etwas besser abschneiden. Die CARLA-Videos können mit dem Sensor jedoch auch gute Ergebnisse erzielen. Dies bedeutet, dass CARLA die Aufgaben des IPG CarMakers übernehmen kann. An den Darstellungseinstellungen in CARLA wurden für die zweite Phase einige Veränderungen am Sichtfeld und der Neigung vorgenommen. Eine weitere Versuchsreihe könnte untersuchen, ob diese Veränderungen zu besseren Ergebnissen bei den Vergleichsvideos führen.

Für die neu erstellten Testszenarien gilt, dass der Sensor in nebeligen oder dunklen Situationen nicht mehr konstant funktioniert. Jedoch ist bei den Kollisionstest das Nachtszenario deutlich zu dunkel. Es ist ratsam, dieses Szenario erneut zu testen. Für zukünftige Szenarien könnte man den Nebel verringern und die Testumgebung in der Simulation besser ausleuchten. Zeitlich war es nicht möglich, weitere Szenarien zu prüfen.

Der Sensor hat noch einige weitere Funktionen, welche in den erstellten Szenarien nicht getestet wurden. Für zukünftige Szenarien können verschiedenste Karten erstellt werden, die sich für diese Sensoren besonders gut eignen. Dennoch ist erkennbar, dass CARLA als Simulationsumgebung für das Testen von optischen Sensoren geeignet ist.

Literaturverzeichnis

- [1] Verkehrliche Wirkung autonomer Fahrzeuge. In: *Autonomes Fahren*. B. Friedrich, https://link.springer.com/content/pdf/10.1007%2F978-3-662-45854-9_16.pdf, zuletzt abgerufen am 5. Januar 2022
- [2] Automotive – Schwachstellenanalyse von Sensorsystemen in Fahrzeugen, Bundesamt für Sicherheit in der Informationstechnik, <https://www.bsi.bund.de/DE/Karriere/Einstieg/Studium-und-Ausbildung/Bachelor-und-Masterarbeiten/Arbeiten/Automotive.html>, Zuletzt abgerufen am 21.12.2021
- [3] Tencent Keen Security Lab, „Experimental Security Research of Tesla Autopilot“, 2019, https://keenlab.tencent.com/en/whitepapers/Experimental_Security_Research_of_Tesla_Autopilot.pdf, zuletzt abgerufen am 22.05.2022,
- [4] IPG CarMaker, <https://ipg-automotive.com/de/produkte-loesungen/software/Car-Maker/>, Zuletzt abgerufen am 22.05.2022
- [5] Praxisprojektdokumentation: „Aufsetzung einer OpenSource Simulationsumgebung zum Testen von optischen Sensoren.“ J. Gellrich 2021
- [6] CARLA Simulator, <https://CARLA.org>, zuletzt abgerufen am 22.05.2022,
- [7] Masterarbeit Angreifbarkeit optischer Fahrerassistenzsysteme, R.Rabah 2019
- [8] Fahrstabilisierungssysteme und Fahrerassistenzsysteme. Wiesbaden: Vieweg+Teubner, K. Reif, 2010. <https://link.springer.com/book/10.1007/978-3-8348-9717-6>
- [9] SAE Levels of Driving Automation™ Refined for Clarity and International Audience, <https://www.sae.org/blog/sae-j3016-update>, zuletzt abgerufen am 22.05.2022
- [10] Handbuch Fahrerassistenzsysteme. Wiesbaden: Springer Fachmedien Wiesbaden, H. Winner, 2015.
- [11] Night Vision: Requirements and possible roadmap for FIR and NIR systems, J. Källhammer, 2006
- [12] Visualizing BMW's self-driving future, Nick Davis, 2020, <https://blog.unity.com/manufacturing/visualizing-bmws-self-driving-future>, zuletzt abgerufen am 10.05.2022,
- [13] OpenDRIVE, <https://www.asam.net/standards/detail/opendrive/> zuletzt abgerufen am 22.05.2022
- [14] FBX Dateiformat, <https://www.autodesk.com/products/fbx/overview>, zuletzt abgerufen am 25.05.2022

- [15] Unreal Engine, https://www.unrealengine.com/en-US/?utm_source=GoogleSearch&utm_medium=Performance&utm_campaign=an*3Q_pr*UnrealEngine_ct*Search_pl*Brand_co*DE&utm_id=15913918984&sub_campaign=&utm_content=July2020_Generic_V2&utm_term=unreal%20engine, zuletzt abgerufen am 16.05.2022
- [16] RoadRunner, <https://de.mathworks.com/products/roadrunner.html> zuletzt abgerufen am 22.05.2022
- [17] Pygame Bibliothek, <https://www.pygame.org/wiki/about>, zuletzt abgerufen am 25.05.2022
- [18] Mobileye, <https://www.mobileye.com> , zuletzt abgerufen am 16.05.2022
- [19] Mobileye 630 Pro Produktseite, verfügbar nur in Hebräisch, <https://www.mobileye.com/he-il/%d7%9e%d7%95%d7%a6%d7%a8%d7%99%d7%9d/mobileye-630-pro/>, zuletzt abgerufen am 16.05.2022
- [20] „Mobileye Startup Guide“, Nr. 2.1. S. 1–93, 2018.
- [21] The Drive Mobileye Display, <https://www.thedrive.com/tech/7742/mobileye-installs-aftermarket-crash-avoidance-tech-on-lyft-uber-vehicles-in-nyc>, zuletzt abgerufen am 23.05.2022
- [22] „Mobileye 6 @ Technical Specification Sheet“, https://rvtechlibrary.com/electronics/mobileye_6tech.pdf, zuletzt abgerufen am 16.05.2022
- [23] Technische Daten der Mercedes C-Klasse, <https://www.mercedes-benz.de/passengercars/mercedes-benz-cars/models/c-class/coupe-c205/specifications/specifications-teaser.module.html>, zuletzt abgerufen am 22.05.2022
- [24] Fernseher PROLITE TF6537UHSC-B2AG, https://iiyama.com/de_de/produkte/pro-lite-tf6537uhsc-b2ag, zuletzt abgerufen am 22.05.2022
- [25] Allgemeine Verwaltungsvorschrift zur Straßenverkehrs-Ordnung (VwV-StVO), https://www.verwaltungsvorschriften-im-internet.de/bsvwvbund_26012001_S3236420014.htm
- [26] Personen Roadpaintings, <https://www.pngwing.com/de/free-png-ypmct>, zuletzt abgerufen am 22.05.2022
- [27] Auto Rückseite, <https://www.pngegg.com/en/png-dxwge>, zuletzt abgerufen am 23.05.2022
- [28] Drecktexturen für die verschmutzten Schilder, <https://www.freeiconspng.com/img/43606>
<https://www.freeiconspng.com/img/43608>
 zuletzt abgerufen am 22.05.2022

- [29] Grafititextur für die verschmutzten Schilder, <https://www.nicepng.com/our-pic/u2q8q8t4t4a9a9a9-graffiti-art-graffiti-paint-splatter/>, zuletzt abgerufen am 22.05.2022

Anhang

Hardware zur Prüfung der Szenarien:

- EyeWatch Display-Einheit
Stellt die empfangenen Daten der Kamera für den Fahrer da
- Mobileye Pro 630 Frontkamera-Einheit
Aufnahme Kamera des Mobileye 630 Pro Systems
- TEKTRONIX AFG3102C V1.0.2
Arbiträrsignal-/Funktionsgenerator zum Erzeugen eines analogen Signals, welches der Geschwindigkeitserfassung im Mobileye Systems dient.
- Mobile 5 Enhancement Box
Übersetzt das analoge Signal des Funktionsgenerators in ein CAN Signal
- Iiyama PROLITE TF6537UHSC-B2AG
Fernseher, auf dem die aufgenommenen Szenarien abgespielt werden

Eingesetzte Software:

- CARLA Linux Built V 0.9.12
CARLA ist ein Plugin für die Unreal Engine und basiert auf einer Client-Server Architektur. Der Server ist für das Rendern der Sensoren, dem World State, sowie die Berechnung der Physik innerhalb der Engine verantwortlich. Der Client ist für die Logik der Akteure und die Verfassung (z.B. Wetterbedingungen) verantwortlich. Die CARLA API kommuniziert zwischen dem CARLA Server und verschiedensten Pythonskripten, welche den Server als Client beeinflussen. CARLA wurde als Source-Built aufgebaut, um den Unreal Editor nutzen zu können. Dank des Unreal Editors können weitere Modelle und Assets hinzugefügt und Karten manipuliert werden.

Um neue Umgebungen zu importieren, besitzt CARLA einen Import-Ordner, indem alle benötigten Dateien (FBX und OpenDRIVE) liegen müssen, die importiert werden sollen. Mittels einem „Make Import“ Befehl im Linux-Terminal werden die Dateien dann zum Erstellen einer neuen Karte genutzt. Es können auch mehrere Karten gleichzeitig erstellt werden. Wichtig ist, dass die zusammengehörenden FBX- und OpenDRIVE-Dateien gleich heißen.

- Unreal Engine V4.26

Die Unreal Engine ist unter anderem ein 3D- Editor und wird mit dem Builtvorgang heruntergeladen. Nach der Installation wird die Unreal Engine mit CARLA zusammen gestartet. Der Server wird über die Oberfläche der Engine gestartet und beendet. Zusätzlich lassen sich weitere Änderungen an der Karte vornehmen. Die Physik-Berechnungen des Servers beruhen auf der Unreal Engine.

- RoadRunner
RoadRunner ist ebenfalls ein 3D-Editor, in dem Straßennetze erzeugt und exportiert werden können. Weitere Objekte können ebenfalls platziert und exportiert werden. Man hat die Möglichkeit, die Karte sowohl in RoadRunner als auch in der Unreal Engine weiter zu gestalten. Roadrunner dient nur als Designsoftware. Aus dem Programm werden anschließend eine OpenDRIVE-Datei und eine FBX-Datei exportiert. Die FBX-Datei enthält die Straße und alle platzierten Objekte als 3D-Datei. Die OpenDRIVE-Datei enthält Wegpunkte für den Straßenverlauf.
- Open Broadcaster Software
OBS ist ein Streaming und Aufnahmeprogramm zur Aufzeichnung von Bildschirmen.

Verwendete Skripte

- generate_traffic.py
- Das generate_traffic.py Skript erzeugt eine bestimmte Anzahl (Default = 30) an Fußgängern und Fahrzeugen. Diese bewegen sich über die Karte anhand von Wegpunkte, welche sich in der OpenDRIVE-Datei befinden. Es ist wichtig, dieses Skript zu starten, bevor man einen eigenen Akteur erzeugt, da die Wegfindung verbessert wird. Wird das Skript außerdem vor einem manual_control.py Skript beendet, verliert der Autopilot seine Wegfindung.
- manual_control_pygmewindow.py
manual_control_pygmewindow.py ist eine erweiterte Variante des manual_control.py Skripts. Mit manual_control.py wird ein Fahrzeug erzeugt, über das der Nutzer die volle Kontrolle bekommt. Zusätzlich öffnet sich ein weiteres Fenster, das die Sicht eines Sensors am Fahrzeug besitzt. Das Fahrzeug lässt sich steuern, kann aber auch von einem Autopiloten übernommen werden. Ein Fahrzeug besitzt normalerweise 5 Sensoren, welche sich durch ihren Blickwinkel und die Rotation der Kamera unterscheiden. Manual_control_pygmewindow verändert die Auflösung des Fensters auf 2560x1440p (Default = 1280x720p) und fügt einen weiteren Sensor hinzu, welcher den Blickwinkel und das Sichtfeld an die Sicht des IPG Car-Makers annähert.
- environment.py

Environment.py passt die Umgebungsverhältnisse der Simulation an. Über dieses Skript lassen sich Wetterverhältnisse, sowie Lichter von anderen Fahrzeugen steuern. Dieses Skript wird benutzt, um Nebel zu erzeugen.

Für alle Skripte gilt, dass diese vor dem Server beendet werden müssen. Passiert dies nicht, stürzt die gesamte Umgebung ab.


Erklärung

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer oder der Verfasserin/des Verfassers selbst entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Anmerkung: In einigen Studiengängen steht die Erklärung am Ende des Textes.

Köln, den 28.05.2022

Ort, Datum



Rechtsverbindliche Unterschrift